بسم الله الرحمن الرحيم

جامعة السّودان المفتوحة

برنامج الحاسوب

تطبيقات قواعد البيانات

رمز المقرر ورقمه: **4043**

إعداد المادة العلمية: د. طارق محمد حسن

تصمـــيم تعلـــيمي : د. عبد الباسط محمد شريف محمد

التدقيق اللغوي: أ. الهدي عبد الله محمد محمد أحمد

التصميم الفني : منى عثمان أحمد النقّة

منشورات جامعة السودان المفتوحة، الطبعة الأولى 2009

جميع الحقوق محفوظة لجامعة السودان المفتوحة، لا يجوز إعادة إنتاج أيّ جزء من هذا الكتاب، وبأيّ وجه من الوجوه، إلاّ بعد الموافقة المكتوبة من الجامعة.

مقدمة المقرر

الحمد لله فالق الإصباح وجاعل الليل سرمدا والشمس والقمر حسبانا، والذي جعل الشمس والقمر والنجوم مسخرات بأمره ألا له الخلق والأمر تبارك الله رب العالمين. والصلاة والسلام على البشير النذير رحمة الله للعالمين وشفيع الخلق أجمعين وإمام المتقين وقائد الغر المحجلين والمبعوث رحمة للعالمين. صلوات ربي و صلاته وتسليماته عليه وعلى من سار على دربه واهتدى بهديه إلى يوم الدين، وبعد.

عزيزي الدارس ،

تتطور نظم قواعد البيانات بسرعة كبيرة للغاية، ففي الأعوام الأخيرة ظهرت العديد من التكنولوجيا الجديدة ، كما اختفت الكثير من التكنولوجيا السابقة في مجال قواعد البيانات.

الهدف من هذا الكتاب "تطبيقات قواعد البيانات" هو توفير مرجع باللغة العربية لجميع المبرمجين الذين يرغبون بالتوجه نحو عملية بناء التطبيقات .

يشرح هذا الكتاب في وحدتين قواعد البيانات، تاريخ تطورها، قواعد البيانات ذات العلاقات، والتطبيق بواسطة لغة SQL.

ويشمل هذا الكتاب الوحدات التالية :

الوحدة الأولى : " أساسيات قواعد البيانات العلائقية " فيها تعريف قواعد البيانات ونظم إدارة قواعد البيانات، والمنظور التاريخي في نظم إدارة قواعد البيانات ومزايا نظم إدارة قواعد البيانات ، وتقدم الوحدة مقارنة بين نظام الملفات ونظم إدارة قواعد البيانات.

وتبحث الوحدة في النموذج العلائقي ، وتتناول لغات النموذج العلائقي من حيث لغة الاستعلام العلائقية ولغة الاستعلام الهيكلية ، وفي التصميم المنطقي لقاعدة البيانات يتم شرح خطوات تحويل مخطط ER بواسطة لغة الـ SQL إلى النموذج العلائقي"مجموعة من الجداول".

الوحدة الثانية وعنوانها: " التطبيق بواسطة لغة SQL " يشرح الكتاب في هذه الوحدة بأدق التفاصيل الاستعلام ومعالجة البيانات بوساطة لغة SQL. والاستعلامات

عبارة عن أوامر بلغة SQL مخصصة للقيام بعملية ما على البيانات ، وهي أيضاً قادرة على إجراء التعديلات على البيانات في قاعدة البيانات نفسها، وكذلك هي قادرة أيضاً على بناء جداول أخرى في قاعدة البيانات .

فهذا الكتاب "تطبيقات قواعد البيانات" يحتوي على كل ما يحتاجه الطالب المبتدئ والمتقدم لتعلم مفاهيم ومهارات تطبيقات قواعد البيانات، حيث يقدم شرحاً وافياً مصحوباً بالأمثلة للدروس الأساسية والمتقدمة في تطبيقات قواعد البيانات.

كل وحدة من هذه الوحدات تحمل عنواناً جامعاً يعبر عن موضوعاتها ، ولكل منها مقدمة وخلاصة ، كما زودت كل وحدة بعدد من أسئلة التقويم الذاتي ، و بعدد من التدريبات ، وكذلك تشمل كل وحدة بمسرداً للمصطلحات ، كما ذيلنا وحدات هذا المقرر بقائمة للمصادر والمراجع المتنوعة التي تم الاستعانة بها في بناء الوحدات ليلجأ إليها الدارس إذا ما احتاج تفاصيل أكثر.

عزيزي الدارس، لدى قراءك هذا الكتاب أرجو أن لا يغيب عن ذهنك أن مادة هذا الكتاب هي الحد الأدنى من المعرفة في موضوعه ، الذي يسمح بإثرائه على الدوام بالرجوع إلى القراءات المساعدة والاطلاع عليها .

عزيزي الدارس، وختاماً نتمنى أن نكون قد وفقنا في تقديم إضافة جديدة في مجال تطبيقات قواعد البيانات لم ولأمتنا الإسلامية والعربية والله أسأل التوفيق والسداد وإليه الملجأ والمآب، إنه ولي ذلك والقادر عليه و صلى الله على سيدنا محمد أولاً ،آخرا.

ونحن نرحب بكل نقد بناء يسهم في تطوير هذا العمل إلى الصورة الأفضل.

الأهداف العامة للمقرر



عزيزي الدارس، بعد فراغك من دراسة هذا المقرر يؤمل أن تكون لك القدرة على:

- **تعرّف** قواعد البيانات ونظم إدارة قواعد البيانات.
- **تقدم** نبذة عن تاريخ تطور قواعد البيانات.
- **تشرح** مكونات قاعدة البيانات العلائقية.
- تعداد جمل لغة SQL .
- استخدام جملة SELECT لاسترجاع البيانات المخزنة في قاعدة البيانات.
- استعراض بعض من دوال السجل.
- شرح كيفية التعامل مع عدة جداول.
- وصف عمل دوال المجموعات.
- توضيح كيفية التعامل مع البيانات من حيث:الإدخال – التعديل – الحذف بوساطة جمل الـ SQL .
- شرح كيفية إنشاء الجداول.
- استخدام قيود السلامة على مستوى الجداول للتحقق من سلامة العمليات التي تجرى على الجداول .

محتويات المقرر

الوحدة	اسم الوحدة	الصفحة
1	أساسيات قواعد البيانات العلائقية	1
2	التطبيق بواسطة لغة SQL	47


الوحدة الأولى
أساسيات قواعد البيانات العلائقية
Relational Database Concepts

محتويات الوحدة

المحتوى	رقم الصفحة
المقدمة	**3**
تمهيد	3
أهداف الوحدة	4
1. تعريف قواعد البيانات و نظم إدارة قواعد البيانات	**5**
1.1 تعريف قواعد البيانات	5
2.1 تعريف نظم إدارة قواعد البيانات	5
2. أنواع قواعد البيانات	**6**
3. المنظور التاريخي	**7**
4. مزايا نظام إدارة قواعد البيانات	**8**
5. مقارنة بين نظام الملفات و نظم إدارة قواعد البيانات	**11**
6. المخططات التجريدية	**13**
7. نموذج الكيانات العلائقي	**15**
1.7 نموذج الكينونات العلائقية	16
8. النموذج العلائقي	**26**
1.8 لغة الاستفسار (الاستعلام) العلائقية	27
2.8 لغة الاستعلام الهيكلية	**27**
3.8 التصميم المنطقي لقاعدة البيانات	33
الخلاصة	**40**
لمحة مسبقة عن الوحدة الدراسية التالية	**43**
حلول التدريبات	**44**
مسرد المصطلحات	**45**

المقدمة

تمهيد

عزيزي الدارس،

نرحب بك إلى الوحدة الأولى من مقرر " تطبيقات قواعد البيانات" والتي تأتي بعنوان " أساسيات قواعد البيانات العلائقية " تحتوي الوحدة على ثمانية أقسام يتناول القسم الأول تعريف قواعد البيانات ونظم إدارة قواعد البيانات ، القسم الثاني يتناول أنواع قواعد البيانات ، أما القسم الثالث فيبحث في المنظور التاريخي في نظم إدارة قواعد البيانات ، يتناول القسم الرابع مزايا نظم إدارة قواعد البيانات ، يتناول القسم الخامس مقارنة بين نظام الملفات ونظم إدارة قواعد البيانات ، والقسم السادس من الوحدة يبحث في المخططات التجريدية . أما القسم السابع فيبحث في نموذج الكيانات العلائقي ويشتمل على تعريف الكيان و تعريف لمجموعة الكيانات والمفتاح . كما يبحث القسم في كيفية استخدام العلاقة **Relationship** للربط بين كيانين أو أكثر، وكيفية استخدام قيد المفتاح لتوضيح نوع العلاقة بين الكيانات . ويتناول القسم أيضاً كيفية استخدام هرمية **ISA** **(ISA Hierarchy)** . القسم الثامن والأخير يبحث في النموذج العلائقي ويتناول لغات النموذج العلائقي، حيث يعرف القسم لغة الاستعلام العلائقية ولغة الاستعلام الهيكلية . كما يبحث القسم أيضاً في التصميم المنطقي لقاعدة البيانات، حيث يتم شرح خطوات **تحويل مخطط ER بواسطة لغة الـــSQL إلى النموذج العلائقي "مجموعة من الجداول"** .

أهداف الوحدة



عزيزي الدارس، بعد فراغك من دراسة هذه الوحدة ينبغي أن تكون قادراً على أن:

- **تعرِّف** قواعد البيانات ونظم إدارة قواعد البيانات.
- **تعدد** أنواع قواعد البيانات.
- **تبين** مزايا نظم إدارة قواعد البيانات.
- **تقارن** بين نظام الملفات ونظم إدارة قواعد البيانات .
- **تستعرض** المخططات التجريدية .
- **تذكر** الأشخاص الذين يتعاملون مع **DBMS** .
- **تمثل** أي مخطط **Schema** بواسطة **ER Diagrams** .
- **تحول** أي نقطة أو أي جزء من **ER Diagrams** إلـــي النمـــوذج العلائقي **Relational Schema** باستخدام جمل **SQL** .
- **تعرِّف** الكيان الضعيف .
- **توضح** المقصود بقيد التغطية **Covering Constraint** .
- **تفصل** مكونات قاعدة البيانات العلائقية .
- **تعرِّف** لغة الاستعلام الهيكلية **Structured Query Language (SQL)**.
- **توضح** المقصود بالاستعلام .
- **تعرِّف** الجدول .
- **تشرح** خطوات عملية تحويل مخطط **ER** بواسطة لغة **SQL** إلى النموذج العلائقي .

توطئة

عزيزي الدارس ،

تعتبر قواعد البيانات (**Database**) من المكونات الأساسية في المجتمعات الحديثة ويتم التعامل معها بصورة يومية، حيث إن النشاطات اليومية قد تعتمد عليها بشكل ملحوظ. فعلي سبيل المثال عندما نتعامل مع البنك من أجل عملية إيداع أو سحب مبالغ مالية نجد أن البنك يضع كل هذه البيانات المختلفة بأرصدة العملاء في قاعدة بيانات، وهي تعتبر وعاء لتلك البيانات. ومثال آخر على ذلك إجراء حجز في فندق أو على إحدي خطوط الطيران فإن كل هذه العمليات مرتبطة وبشكل مباشر مع قواعد البيانات . ولا يقتصر العمل بقواعد البيانات في الشركات ومؤسسات الأعمال الضخمة بل نجدها في الأعمال الصغيرة مثل المتاجر ومحلات قطع الغيار وما شابه ذلك.

1. تعريف قواعد البيانات و نظم إدارة قواعد البيانات

1.1 تعريف قواعد البيانات Database Definition

قاعدة البيانات (**Database**) عبارة عن وعاء لمجموعة من البيانات تعكس نشاط منظمة أو شركة أو هيئة ذات نشاط محدد. فعلى سبيل المثال نجد أن قاعدة بيانات الجامعة تشتمل على كل البيانات المتعلقة بالطلاب و الأساتذة و المقررات الدراسية.

2.1 تعريف نظم إدارة قواعد البيانات

Database Management Systems (DBMS)

تعتبر نظم إدارة قواعد البيانات (**DBMS**) مجموعة من الحزم البرمجية (**Software Packages**) مهمتها إدارة و تخزين قواعد البيانات.و مثال لهذه النظم **Oracle , MS-Access , SQL Server …**

أسئلة تقويم ذاتي

1/ عرِّف كلاً من قواعد البيانات ونظم إدارة قواعد البيانات.

2. أنواع قواعد البيانات Database Types

عزيزي الدارس ، مع نمو الطلب على تطبيقات قواعد البيانات (**Database Applications**) ظهرت تطبيقات جديدة متعلقة بقواعد البيانات حتي تناسب طبيعة البيانات التي يمكن أن تستوعبها و تقوم بعمل المعالجات عليها، نذكر منها الآتي:

أ- قواعد البيانات للوسائط المتعددة (Multimedia database)

يتيح هذا النوع من قواعد البيانات تخزين البيانات المتمثلة في الصورة ومقاطع الفيديو والرسائل الصوتية، و توفر آلية للتعامل مع هذا النوع من البيانات لكي يتسنى التعامل معها.

ب- أنظمة المعلومات الجغرافية (Geographic information syste)

وهي معنية بتخزين وتحليل الخرائط الجغرافية والصور المختلفة من الأقمار الصناعية .

ت- مخازن البيانات (DATAWARE HOUSE)

وهي مستخدمة من قبل الشركات الكبيرة وتحتوي على بيانات ضخمة تتيح عملية استخلاص وتحليل البيانات للمساعدة في عملية اتخاذ القرار .

أسئلة تقويم ذاتي

1/ عدد أنواع قواعد البيانات.

3. المنظور التاريخي (Historical perspective)

عزيزي الدارس،

نجد أن الحواسيب وتطبيقاتها كانت معنية وبشكل مباشر في تخزين ومعالجة البيانات . وأول نظام للأغراض الخاصة تم لنظم إدارة قواعد البيانات عن طريق **Charles Bachman** من شركة **General Electric** في العام **1960**، وكان يطلق على النظام **Integrated Data Store** وهذا النظام كان يستخدم كنموذج تشبيكي للبيانات **Network data Model**. في العام **1960** قامت شركة **IBM** بتطوير نظام لإدارة المعلومات **Information Management DBMS (IMS)**، وتم بناء هذا النظام باستخدام النموذج الهرمي للبيانات **Hierarchical data Model**. وفي العام **1970** في احد معامل **IBM** تم اقتراح إطار جديد لتمثيل البيانات، وسمي هذا التمثيل بنموذج البيانات العلائقي **Relational Data Model** وقد ساهم هذا النموذج العلائقي في تطوير عدد كبير من نظم إدارة قواعد البيانات **(DBMS)** وفي عام **1980** انتشر النموذج العلائقي بشكل كبير وأصبح يمثل النموذج الرئيسي لنظم إدارة قواعد البيانات .

تم تطوير لغة الاستعلام الهيكلية(**Structured Query Language: SQL**) لغة للتعامل مع قواعد البيانات العلائقية، وفي أواخر العام **1980** أصبحت لغة **SQL**، لغة قياسية . و تم عمل إصدار من هذه اللغة و سميت **SQL – 92** بواسطة معهد القياسات الوطني الأمريكي **(American National Standards Institutes ANSI)** ومنظمة القياسات العالمية **(International Standards Organization)**.

في أواخر الثمانينات وأثناء فترة التسعينيات تم التقدم في نظم إدارة قواعد البيانات بشكل ملحوظ من خلال الأبحاث التي دفعت بقوة استخدام **SQL** وكذلك نماذج قواعد البيانات **(Database Model)** مما جعل تلك النظم تتعامل مع التحليل المعقد للبيانات ولمختلف الأعمال . وظهرت عدد من الشركات التي تعتبر في يومنا هذا من عمالقة التقنية في مجال نظم إدارة البيانات، ونذكر من تلك الأنظمة آي بي ام **(IBM)**

وأوراكل (**Oracle**) , **DB2** , **Informix** وتتمتع هذه الأنظمة بقدرات فائقة في تخزين البيانات و إدارتها.

ولاشك أن دخول نظم إدارة قواعد البيانات **(DBMS)** عالم الانترنت(**Internet World** لتخزين المواقع المختلفة بياناتها علي قواعد البيانات تعتبر خطوة مهمة لظهور تطبيقات جديدة حولت إمكانية عمل الاستعلامات المختلفة من خلال مستعرض الإنترنت **(Internet Explore)**.

واستمرت نظم إدارة قواعد البيانات تزداد أهميتها مع ظهور التقنيات المتطورة في مجال الشبكات (**Network Technologies**)، والتي أتاحت خدمة تعامل الخط المباشر (**On–Line**) مع قواعد البيانات و التي تسمح في التعامل مع البيانات في الزمن الحقيقي **(Real Time)**، ومثال لفائدة هذه الخدمة الصرافات الآلية(**ATMs)** التي سهلت لعملاء البنوك التعامل مع حساباتهم على مدار الساعة و من مواقع مختلفة. وكذلك عملية الحجوزات على خطوط الطيران من خلال شبكة الإنترنت.

أسئلة تقويم ذاتي



1/ في أواخر الثمانينات وأثناء فترة التسعينيات ظهرت عدد من الشركات التي تعتبر في يومنا هذا من عمالقة التقنية في مجال نظم إدارة البيانات، اذكر بعضاً من تلك الأنظمة.

4. مزايا نظام إدارة قواعد البيانات

Advantages of Database Management System

عزيزي الدارس، هنالك الكثير من مزايا نظم إدارة قواعد البيانات نذكر منها على سبيل المثال:

أ- التحكم في التكرار Controlling Redundancy

تكرار البيانات يعني أن البيانات تكون مخزنة بصورة متكررة في قاعدة البيانات، وهذا التكرار يمكن أن يكون جذوراً لمشاكل متعددة ونذكر منها :

- إدخال عدد من البيانات بشكل متكرر يقود إلي زيادة الوقت اللازم لإدخال تلك البيانات، وهذا بالتالي يؤثر على مستوى الأداء **Bad Performance** .
- يتم استخدام مساحات إضافية لتخزين البيانات المكررة **Storage consumption**.
- إذا تم تعديل أي نسخة من البيانات لابد من تعديل كل النسخ، وإذا لم يحدث ذلك فان قاعدة البيانات تكون في حالة غير منسجمة **State Inconsistent** .

من خلال ما ذكر نجد أن منع تكرار البيانات يزيد من مستوى الأداء ويوفر مساحات تخزين و يحافظ على سلامة البيانات (**Data Integrity**). وعن طريق التصميم الجيد لقاعدة البيانات وعمل قيود السلامة (**Integrity Constraints**) تمنع نظم إدارة قواعد البيانات دخول البيانات بشكل متكرر .

ب- حماية البيانات Data Protection

تمنع نظم إدارة قواعد البيانات الأشخاص غير المصرح لهم من الوصول إلي البيانات (**Data Accessing**). ونستخدم في ذلك آلية التحكم في الوصول للبيانات **Access Control) Mechanism)** . و من خلال منح الصلاحيات (**Privileges**) للمستخدمين (**Users**) تستطيع **DMBS** فرض السياسات الأمنية (**Security Policies**) بصورة مرنة بما يتناسب مع احتياجات المنظمة، وذلك لأن تلك السياسات يمكن أن تختلف من جهة إلى أخرى.

ت- كفاءة الوصول للبيانات Efficient Data Access

ونعني بها أن هنالك كفاءة كبيرة في عملية تخزين واسترجاع البيانات، وتعتبر هذه الخاصية مهمة جدا إذا كانت البيانات مخزنة في وحدات خارجية. و نجد أن عملية التخزين تتم وفق سياسات تضمن تقليل زمن استرجاع تلك البيانات و هنالك خاصية تسمى **Pre-Fetching** و هي أن **DBMS** تستطيع أن تتنبأ بالبيانات المطلوبة قبل أن يتم طلبها. و تعتبر هذه الخاصية عاملا مهما لزيادة الأداء **(Performance)**.

ث- استقلالية البيانات Data Independence

وتعنى أن برنامج التطبيق**(Application Program)** يكون مستقل تماما من تفاصيل تمثيل وتخزين البيانات و بالتالي أي تعديل يتم على مستوي التطبيق لا تتأثر به البيانات.

ج- إدارة البيانات Uniform Data Administration

هي عملية إدارة البيانات مركزيا. ويتشارك عدد من المستخدمين في هذه البيانات مما يساهم في التطوير وفهم هذه البيانات و تتيح هذه الخاصية لمدير قاعدة البيانات**(DBA Database Administrator)** التحكم و ضبط قاعدة البيانات **(Controlling and Tuning Database)**.

أ- تزامن الوصول ومعالجة الأخطاء Concurrent Access and Crash Recovery

نسبة لأن **DBMS** يتيح لعدد من المستخدمين التعامل معه في نفس الوقت فإنه يستطيع أن يستقبل مجموعة من المعاملات**(Transactions)** في نفس الوقت وذلك عن طريق عمل جدولة**(Scheduling)** لأحدث المعاملات. ويشعر المستخدم الذي يعمل على قاعدة البيانات أنة هو الوحيد الذي يعمل عليها. يمتلك **DBMS** آلية لإحتواء أي فشل **(Crash)** قد يحدث بصورة غير متوقعة بحيث تعيد هذه الآلية قاعدة البيانات إلى وضع الانسجام **(Consistent State)**.

أسئلة تقويم ذاتي



1/ اذكر مزايا نظم إدارة قواعد البيانات.
2/ أحد مزايا نظم إدارة قواعد البيانات استقلالية البيانات **Data Independence**، ماذا تعني هذه الميزة؟

5. مقارنة بين نظام الملفات و نظم إدارة قواعد البيانـــات File System VS DBMS

أ- نجد أن نظام الملفات لا يدعم عملية وصول عدة مستخدمين للبيانات في نفس الوقت، بينما **DBMS** يدعم هذه الخاصية (يمكن لعدد من المستخدمين إستخدام هذه البيانات في نفس الوقت بدون أي تضاربات).

ب- نظام **DBMS** يقوم بتقسيم البيانات حسب الحاجة في (**Random Access Memory RAM**)، بمعنى أن البيانات التي يتم معالجتها هي فقط تحمل في الـ**RAM** بعكس نظام الملفات الذي يقوم بتحميل كل بيانات الملف في **RAM** لكي يتمكن من معالجتها، و في هذه الحالة يجب أن يكون حجم الملف محدوداً بينما نجد أن بيانات قاعدة البيانات يمكن أن تتمدد على القرص .

ت- في حالة نظام الملفات فإننا نحتاج إلى برنامج خارجي يستطيع أن يجيب عن أسئلة المستخدم. ويعتبر هذا البرنامج معقداً نظرا لأنه يتعامل مع كم هائل من البيانات، بينما **DBMS** يتيح أدوات يمكن من خلالها تسهيل هذه المهمة.

ث- يجب أن نتأ كد من أن البيانات تم إعادة تخزينها واحتوائها بنجاح إذا ما حدث دمار للبيانات أثناء التعامل معها (هذه الميزة متوفرة في **DBMS**).

ج- نظام الملفات يعتمد على كلمة السر في عملية الحماية في نظام الملفات وهذه الطريقة لا تتناسب مع عدة مستخدمين، أي يجب أن تكون هنالك مستويات حماية متعددة، وذلك لأن سياسات الأمن و الحماية قد تختلف من مؤسسة إلى أخرى.

ومن ذلك نخلص إلى أن نظام إدارة قواعد البيانات يستطيع أن يتعامل مع كم هائل من البيانات بقوة وكفاءة عالية وإمكانيات تستطيع أن تستوعب نمو البيانات و هنالك خصائص عديدة لا يمكن أن نجدها في نظام الملفات.

❑ نموذج البيانات Data Model

هي عبارة عن مستويات عليا تصف هياكل البيانات وتخفي كثيراً من التفاصيل، ونجد أن نظم إدارة قواعد البيانات تستخدم نوعاً من هذه النماذج يسمى نموذج البيانات العلائقي **Relational Data Model**.

❑ المخطط Schema

عبارة عن وصف خاص لهيكل بيانات معينة، أو لتجمع من البيانات يستخدم فيه نموذج البيانات. و يشمل اسم المخطط و خصائص البيانات **(Attributes)**.

❑ نموذج البيانات العلائقي Relational Model of Data

هو نوع من أنواع نماذج البيانات و يعتبر الأكثر انتشارا اليوم، ويعتمد على مفهوم رئيسي يسمى **Relation** يتكون من عدد من الصفوف **(Rows)** وعدد من الأعمدة **(Columns)** . ونجد**Schema** لكل **Relation** ، تقوم بوصف البيانات التي سوف يحتويها **Relation**.

مثال:

لتمثيل بيانات كل طالب يمكننا أن نقول لكل سجل في علاقة الطالب **Student** **Relation** و نصف أي طالب بواسطة 4 حقول هي :

Student (ID: *Integer* , Name: *string*, Age: Integer, GPA: *Real*)

ولكلٍ وصف خاص به في هذه **Relation**.

أسئلة تقويم ذاتي



1/ قارن بين نظام الملفات و نظم إدارة قواعد البيانات .

6. المخططات التجريدية Levels of Abstraction

عزيزي الدارس، تستخدم هذه المستويات لوصف قاعدة البيانات في عدة مستويات و تشمل المخطط التصوري **Conceptual Schema** و المخطط الفعلي **Physical Schema**الخارجي **External Schema** و في ما يلي نستعرض هذه المخططات:

ب- المخطط التصوري Conceptual Schema

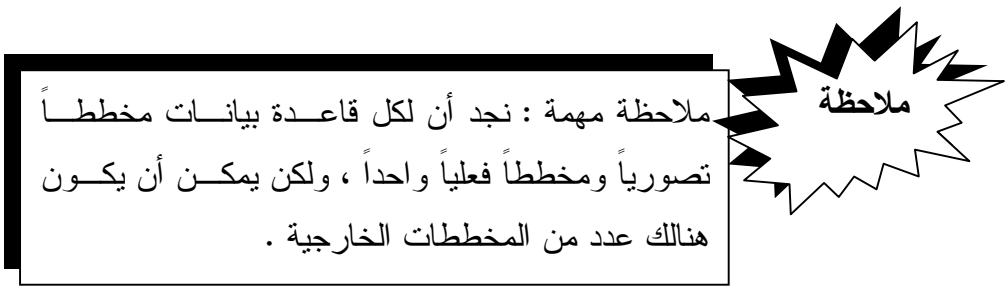
ويسمى أحيانا بالمخطط المنطقي **Logical schema** وفيه يتم وصف البيانات التي سوف تخزن في أجزاء نموذج البيانات، أي أننا نعرف في هذا المستوى كل ال**Relations** التي سوف تخزن في قاعدة البيانات و كذلك الخصائص **(Attributes)** التي تتكون منها.

ت- المخطط الفعلي Physical Schema

وفي هذا المستوى يتم وصف إضافي مفصل يصف عملية التخزين الفعلي (الحقيقي) لـ**Relation** على وسائط التخزين المختلفة مثل الأقراص والأشرطة، وهذا يساعد في عملية إنشاء الفهارس والملفات .

ث- **المخطط الخارجي External Schema**

وفى هذا المستوى يتم وصف عدد من المناظر **(views)** ويتم تحديدها حسب حاجة المستخدمين، وتخصص لهم في شكل مجموعات أو في شكل أفراد تمنح حسب الصلاحيات الممنوحة لهم.



مثال:

هذا المثال يوضح كيف يمكن أن نصف قاعدة البيانات باستخدام تلك المستويات. نجد أن قاعدة بيانات خاصة بطلاب الجامعة نحتاج لتخزين البيانات المتعلقة بالطلاب والمقررات الدراسية ، وتسجيل المواد للطلاب .

أولاً : وصف قاعدة بيانات على المخطط التصوري **Conceptual Schema:**

Student (Sid: *Integer*, Name: *String*, Age: *Integer*)
Course (Cid : I*nteger* , Cname : *String* , Credit : *Integer*)
Std_Cours(Sid : *Integer* , Cid : *Integer*, Grade: *Integer*)

في هذا المستوي يتم وصف كل العلاقات (**Relations**) و التي سوف تستوعب البيانات.

ثانياً: وصف قاعدة البيانات على المخطط الفعلي **Physical Schema:**

على سبيل المثال يمكن أن نصف تخزين البيانات في ملفات غير مرتبة (**Unsorted files**)، وأيضا سوف نقوم بفهرسة كل ملف بواسطة مفتاح العلاقة **Indexes on** (**Primary Key**).

ثالثاً: وصف قاعدة البيانات على المخطط الخارجي **External Schema**

على سبيل المثال يمكن أن ننشئ المنظر **View:**

View : student into { Name : *String* , Age: *Integer* } .

❑ التحكم في تزامن العمليات Concurrency Control

نجد أن أحد مزايا **DBMS** لزيادة الأداء هي عملية التزامن في تنفيذ المعاملات، وذلك لأن التعامل مع وحدات التخزين مثل الأقراص تتكرر وذلك يسبب بطءاً نسبياً ، ومن المهم أن نجعل وحدة المعالجة المركزية **CPU** تقوم بخدمة مستخدمين آخرين،

ويجب على **DBMS** الحيلولة دون ظهور أي نوع من التضاربات التي قد تحدث نتيجة التنفيذ المتزامن كما يجب أن نجعل المستخدم يعمل على النظام و كأنه يعمل لوحده .

❑ المعاملات Transactions

تنفذ الـ **DBMS** المهام في شكل معاملات **Transactions**. و منظور **DBMS** تمثل سلسلة من الأحداث. و كل حدث إما أن يكون قراءة أو كتابة على كائنات قاعدة البيانات (**Database Objects**). و تضمن الـ**DBMS** أن جميع أحداث المعاملة قد نفذت بالكامل أو لم ينفذ أي حدث على الإطلاق، و ذلك لضمان إنسجام قاعدة البيانات (**Database Consistency**) و تسمي خاصية الضمان بـ**Atomicity**. و تقوم الـ **DBMS** بتنفيذ مجموعة من المعاملات بشكل متزامن لزيادة الأداء. و توضع مجموعة من المعاملات في شكل جدولة **Schedule** بحيث يوزع زمن التنفيذ على أحداث المعاملات.

❑ الأشخاص الذين يتعاملون مع DBMS

① المستخدمون **Users**.

② الباعة **Vendors**.

③ مبرمجو تطبيقات قواعد البيانات **Database Application Developer**.

④ مدير قاعدة البيانات **Database Administrator (DBA)** .

أسئلة تقويم ذاتي



1/ اذكر الأشخاص الذين يتعاملون مع **DBMS** .

7. نموذج الكيانات العلائقي Entity Relationship Model

عزيزي الدارس، عند تصميم قاعدة البيانات يجب أن يكون لها تصور واضح قبل تصميمها و لوضع هذا التصور نستخدم نموذج الكيانات العلائقي **Entity Relationship Model (ER)** ونقوم بالإجابة على عدد من الأسئلة وهي :ما كيفية

التعرف على الكيانات **Entities** والعلاقات **Relationship** بين الكيانات، وما الخصائص التي يجب أن توصف بها الكيانات (الخواص) **(Attributes)** التي سوف يحتويها كل كيان وما الخصائص التي تستخدم لوصف العلاقة؟، وكذلك يجب تحديد كل قيود (**Integrity Constraints**) التي تضمن سلامة البيانات.

يتم إستخلاص كل هذه المعلومات من طبيعة تدفق البيانات، و القواعد التي تحكم هذه البيانات يجب أن نمثل أي مخطط **Schema** بواسطة **ER Diagrams**، وأيضا يجب أن نحول أي نقطة أو أي جزء من **ER Diagrams** إلى النموذج العلائقي **Relational Schema** باستخدام جمل **SQL** .

1.7 نموذج الكينونات العلائقية ER Model

1.1.7 الكيان Entity

هو عبارة عن كائن في العالم الحقيقي، ويمكن أن يميز عن بقية الكائنات، و يعرف نفسه بنفسه و لا يعتمد وجوده على كيان آخر. و تستخدم الخصائص **(Attributes)** لوصف الكيانات. فعلى سبيل المثال يعتبر الطالب كياناً في قاعدة بيانات الجامعة، و يمكن وصف هذا الكيان بواسطة الخصائص التالية: الاسم ، العنوان ، الهاتف...الخ

2.1.7 مجموعة الكيانات Entity Set

هي عبارة عن مجموعة من الكيانات المتشابهة ويجب أن يتحقق الأتي :

أ- كل الكيانات في مجموعة الكينونة لديها نفس الخواص.

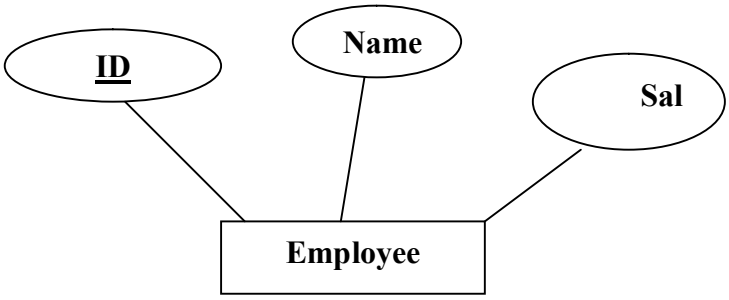
ب- لكل مجموعة كيان مفتاح **Key** .

ت- أي خاصية من الخواص لديها مجال أو **Domain**

ث- ونستخدم الخواص لوصف الكيان. أما المفتاح فيستخدم للوصول إلي بقية الخواص الموجودة في الكيان ومثال على المفتاح : رقم الطالب الأكاديمي.

3.1.7 تعريف المفتاح Key

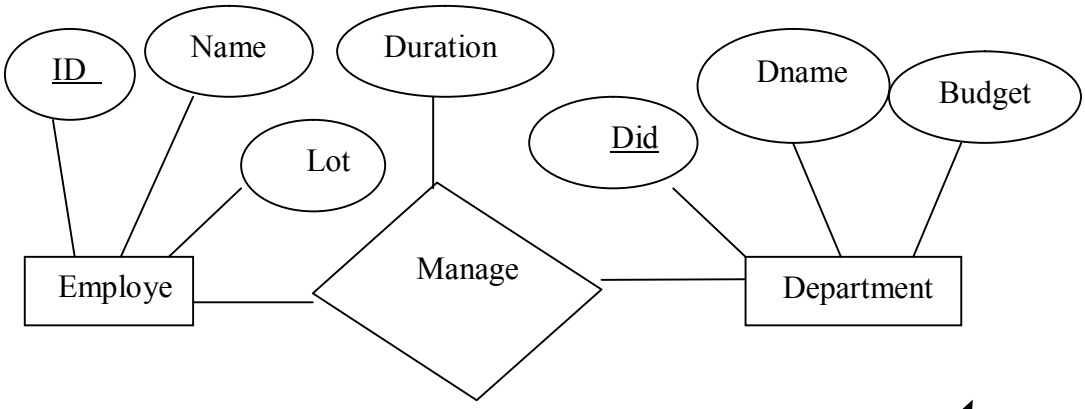
وهو عبارة عن أقل مجموعة ممكنة من الخواص وبصورة وحيدة تعرف الكيان . ويمكن أن يكون هنالك عدد من المفاتيح ولكننا نختار منها واحد يمثل المفتاح الأساسي . يتم تمثيل الكيان في مخطط النموذج العلائقي **Entity Relationship Diagram (ER)** باستخدام مستطيل يكتب عليه اسم الكيان و شكل بيضاوي لتمثيل الخصائص التابعة للكيان و يوضع خط أسفل اسم حقول المفتاح. و المثال التالي يوضح ذلك.



كيان الموظف Employee Entity

4.1.7 العلاقة Relationship

تستخدم العلاقة للربط بين كيانين أو أكثر، و أيضا يمكن أن تكون هنالك خصائص تصف العلاقة، وتوضع العلاقات المتشابهة في مجموعة العلاقة **Relationship set** و نفس مجموعة الكيانات يمكن أن تشترك في مجموعات مختلفة من العلاقات أو بقواعد مختلفة في نفس المجموعة . يتم تمثيل العلاقة في **ER**بواسطة معين و توجد خطوط مستقيمة بين الكيانات المرتبطة بالعلاقة. المثال التالي يوضح علاقة **Manage** بين كيان الموظف **Employee** و كيان الإدارة **Department**

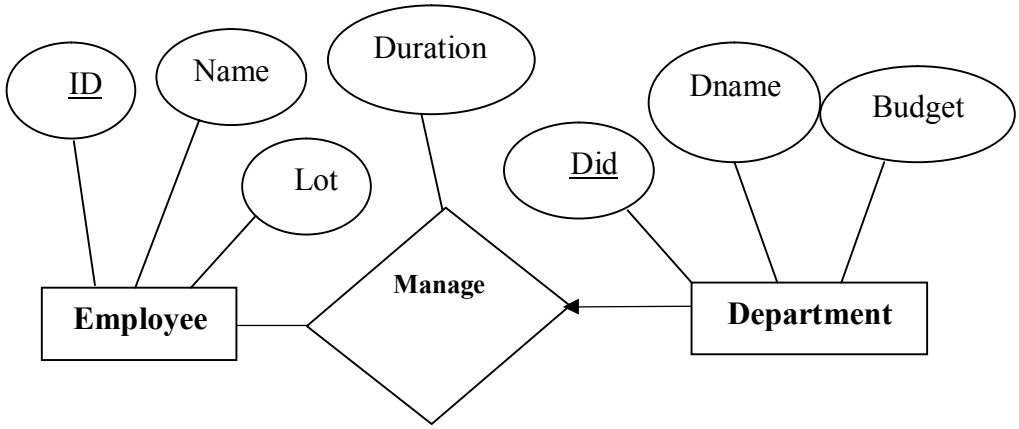


ملاحظة

ملاحظة: تحتوي العلاقة على الخاصية **Duration** التي تسجل فترة إدارة الموظف للإدارة.

5.1.7 قيد المفتاح Key constraint

عزيزي الدارس، يستخدم قيد المفتاح لتوضيح نوع العلاقة بين الكيانات وهي أن يطلب الكيان كياناً واحداً على الأكثر في العلاقة. مثلا إذا كان هنالك موظف يعمل في عدة إدارات (متعدد إلى واحد) **Many-to-One** أو أن نقول إن الإدارات لديها عدد من الموظفين (واحد إلى متعدد) **One-to-Many** أو أن نقول أي إدارة يكون لها على الأكثر مدير واحد ففي هذه الحالة نسمي أن هنالك **Key Constraint** . وتمثل بواسطة سهم من كيان الإدارة إلى العلاقة كما هو موضح في الشكل التالي:



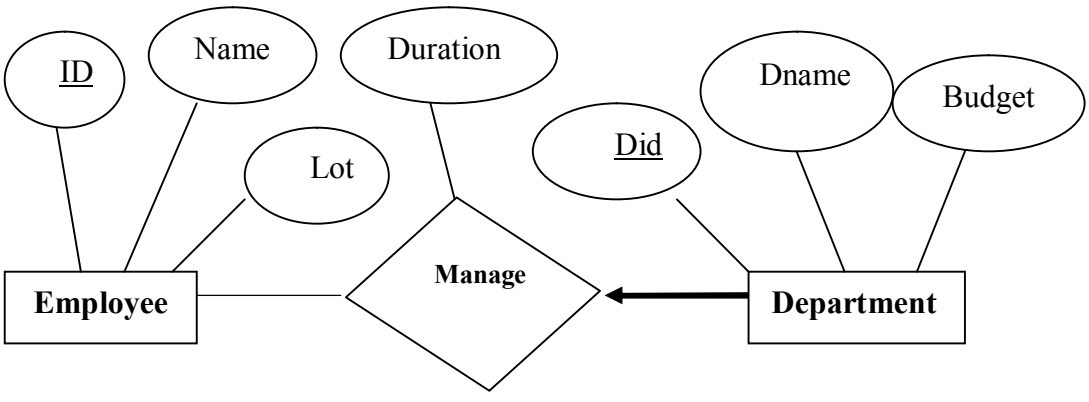
مع العلم أن الموظف يمكن أن يدير أكثر من إدارة، و لكن الإدارة يجب أن تدار بواسطة مدير واحد.

وفي حالة أن الموظف يستطيع أن يعمل في عدة إدارات وكل إدارة يعمل بها عدد من الموظفين . في هذه الحالة تسمى العلاقة (**Many –to –Many**) . و إذا وضعنا قيداً آخر أن الموظف يدير إدارة واحدة فقط و الإدارة تدار بواسطة مدير واحد تسمى العلاقة (**One – to –One**).

6.1.7 القيد التشاركي Participation constraint

في قيد المفتاح نجد أن لكل إدارة مديراً واحداً على الأكثر والسؤال هنا : هل لكل إدارة مدير ؟ وإن كانت الإجابة يجب أن يكون لكل إدارة مدير بمعني أن وجود الإدارة يعتمد على أن يكون هنالك مدير ففي هذه الحالة نسمي الطلب الإلزامي للمدير بـ**Participation Constraints** قيداً تشاركياً أي أن الكيان يطلب و بشكل إلزامي كياناً آخر يشاركه في الإدارة. فمثلا نقول إنه يجب أن يكون لكل إدارة مدير، ونطلق على هذه العلاقة **(Participation Constraint (Total** ليس بالضرورة إن لكل موظف إدارة يديرها في هذه الحالة يسمى **(Partial))** .

و يتم تمثيل القيد ألتشاركي في مخطط **ER** بخط عريض غامق. و لإضافة هذا القيد في المثال السابق الخاص بقيد المفتاح بحيث نقول يجب أن يكون لكل إدارة مدير واحد على الأكثر، و يكون تمثيل ذلك كما يلي:



7.1.7 الكيان الضعيف Weak Entity

هو عبارة عن كيان لا يستطيع أن يعرف نفسه بنفسه، و يعتمد وجوده على كيان آخر.

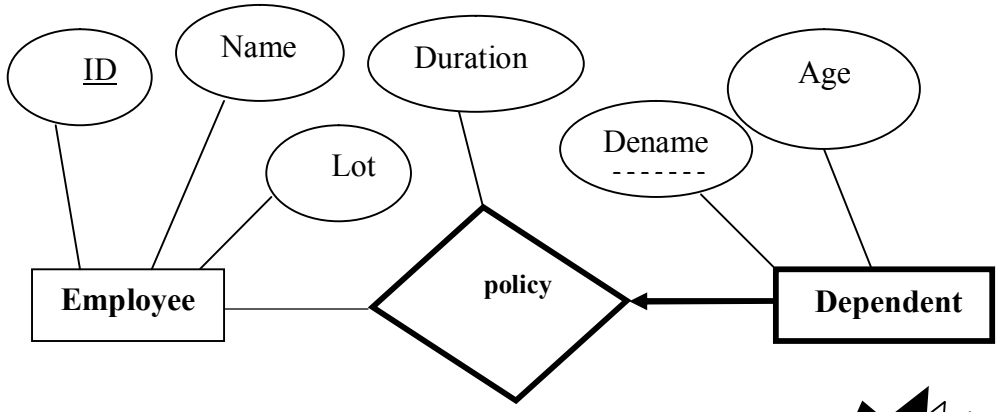
- يمكن أن يعرف فقط بصورة وحيدة باستخدام المفتاح الأساسي لكينونة أخرى أساسية (الكينونة الأساسية (المالكة) **Owner entity**) .

يجب أن تكون العلاقة بين مجموعة الكيانات المالكة ومجموعة الكيان الضعيف من نوع **One –to –Many** . ويجب أن تكون العلاقة التشاركية بالنسبة للكيان الضعيف كلية **Total.**

يتم تمثيل الكيان الضعيف في **ER** مثل الكيان العادي، و يختلف في أن المستطيل يكون بخط عريض، و أيضا العلاقة تكون بخط عريض.

مثال:

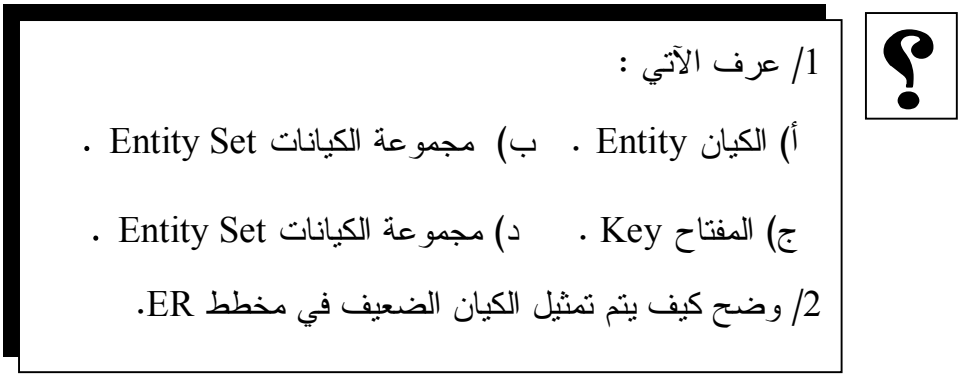
المثال التالي يوضح كياناً ضعيفاً لتمثيل بيانات التابعين لكيان الموظفين. و العلاقة بينهما سياسة العلاج التي تمنح للموظف بواسطة شركة التأمين الصحي:



ملاحظة

نلاحظ أن المفتاح الأساسي في الكيان الضعيف تم تحديده بوضع خطوط متقطعة و ذلك لأن المفتاح الأساسي يتم تعريفه باستخدام هذا الحقل مع المفتاح الأساسي للكيان المالك.

أسئلة تقويم ذاتي



1/ عرف الآتي :

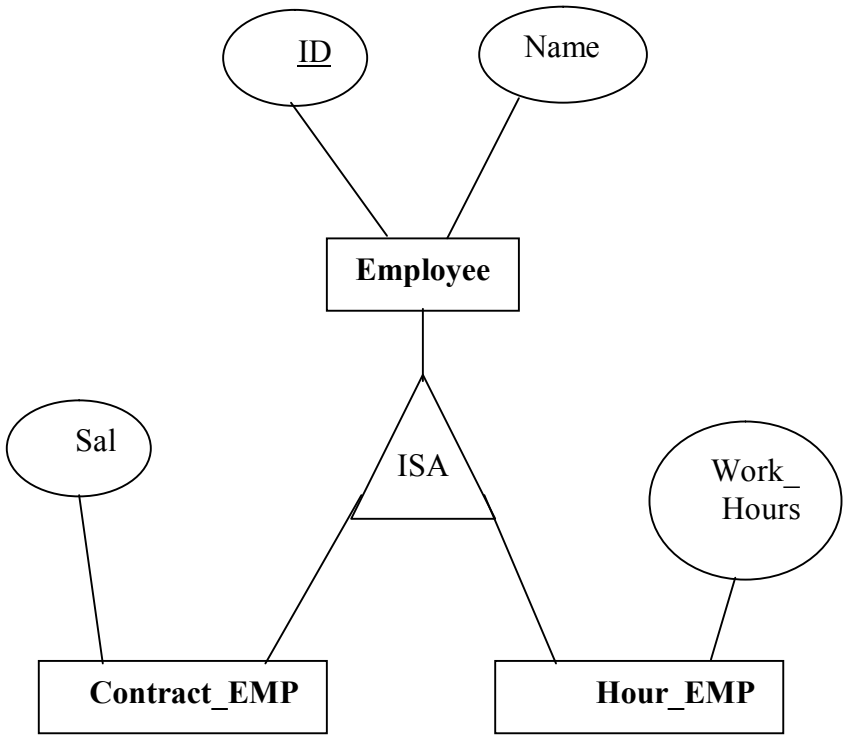
أ) الكيان Entity . ب) مجموعة الكيانات Entity Set .

ج) المفتاح Key . د) مجموعة الكيانات Entity Set .

2/ وضح كيف يتم تمثيل الكيان الضعيف في مخطط ER.

8.1.7 هرمية ISA : ISA Hierarchy

عزيزي الدارس ، في بعض الحالات قد تشترك الكيانات في بعض الخواص مثل الوراثة في مفهوم البرمجة الشيئية **Object (Oriented Concepts)** بحيث يستطيع أن يرث مجموعة من الأصناف صنف آخر. و على سبيل المثال في قاعدة بيانات الشركة هنالك موظفون بنظام العقود، بحيث يوصف كل موظف برقم الموظف و الاسم و الراتب الشهري و هنالك نوع آخر من الموظفين بنظام الساعات و يوصف هذا النوع أيضا برقم الموظف و الاسم و عدد الساعات التي عملها. نلاحظ أن الكيانين يشتركان في بعض الخواص مثال (الرقم و الاسم). يمكن أن نمثل هذه المعلومات في الـ **ER** كما يلي :



يسمى الكيان **Employee** بالكيان الرئيسي الكيانـــان **Contract_EMP** و **Hour_EMP** بالفرعيين .

❑ قيد التغطية **Covering Constraint**

والمقصود بهذا القيد هو أن أي كيان موجود في الكيان الرئيسي يجـــب أن يكـــون موجود في إحدي الكيانات الفرعية. ففي نظام الموظفين مثلا . هل أي موظـــف ينتمـــي إلي مجموعة كيان الموظف يجب أن يكون موظف بنظام الساعات أو موظـــف بنظـــام التعاقد . وهذا يخضع إلي النظام الخاص بالشركة.

ملاحظة

أيضا نستخدم هرمية (**ISA**) إذا كان هنالك كيان فرعي له علاقة مع كيان آخر .

مثال (1):

تمتلك مؤسسة اقتصادية مجموعة من الفروع لبنك، ونجد أن لكل فرع عـــدداً مـــن أرقام الهواتف يمكن لأي شخص (العميل) أن يفتح حساباً في فرع أو عدد من الفروع على أن يقوم الفرع بتسجيل بياناته، وهي: رقم العميل، اسم العميل، جنســيته، عنوانـــه وبيانات ثلاثة أشخاص مقربين إلى العميل. وتشمل هذه البيانـــات الاســـم والهـــاتف. ويستطيع العميل أن يفتح حسابات مختلفة وتحدد الحسابات بالبيانات التالية:

رقم الحساب ، نوع الحساب ، الرصيد المتوفر للحساب ، وأيضا يستطيع العميل عمـــل حركات علي الحساب وتمثل بيانات الحركة برقم الحركة ، ونوع الحركة (إيداع، سحب) والمبلغ الخاص بالحركة .

المطلوب هو:

أ- وضح الكيانات و الكيانات الضعيفة إن وجدت .

ب- وضح العلاقات ونوعها بين الكيانات .

ت- وضح مفاتيح القيود اللازمة .

ث- ارسم مخطط **ER** كاملاً للأشياء السابقة .

الحل :

الكيانات :

أ- العميل { رقم ، اسم ، العنوان ، الجنسية }.

ب- الفرع { رقم الفرع ، عنوان الفرع }.

ت- الحسابات { رقم الحساب ، نوع الحساب ، الرصيد }.

ث- الحركة { رقم الحركة ، نوع الحركة ، قيمة الحركة }.

الكيانات الضعيفة :

أ- الأقارب { اسم ، هاتف } .

ب- هواتف الفرع {رقم الهاتف ، نوعه } .

مفتاح القيد :

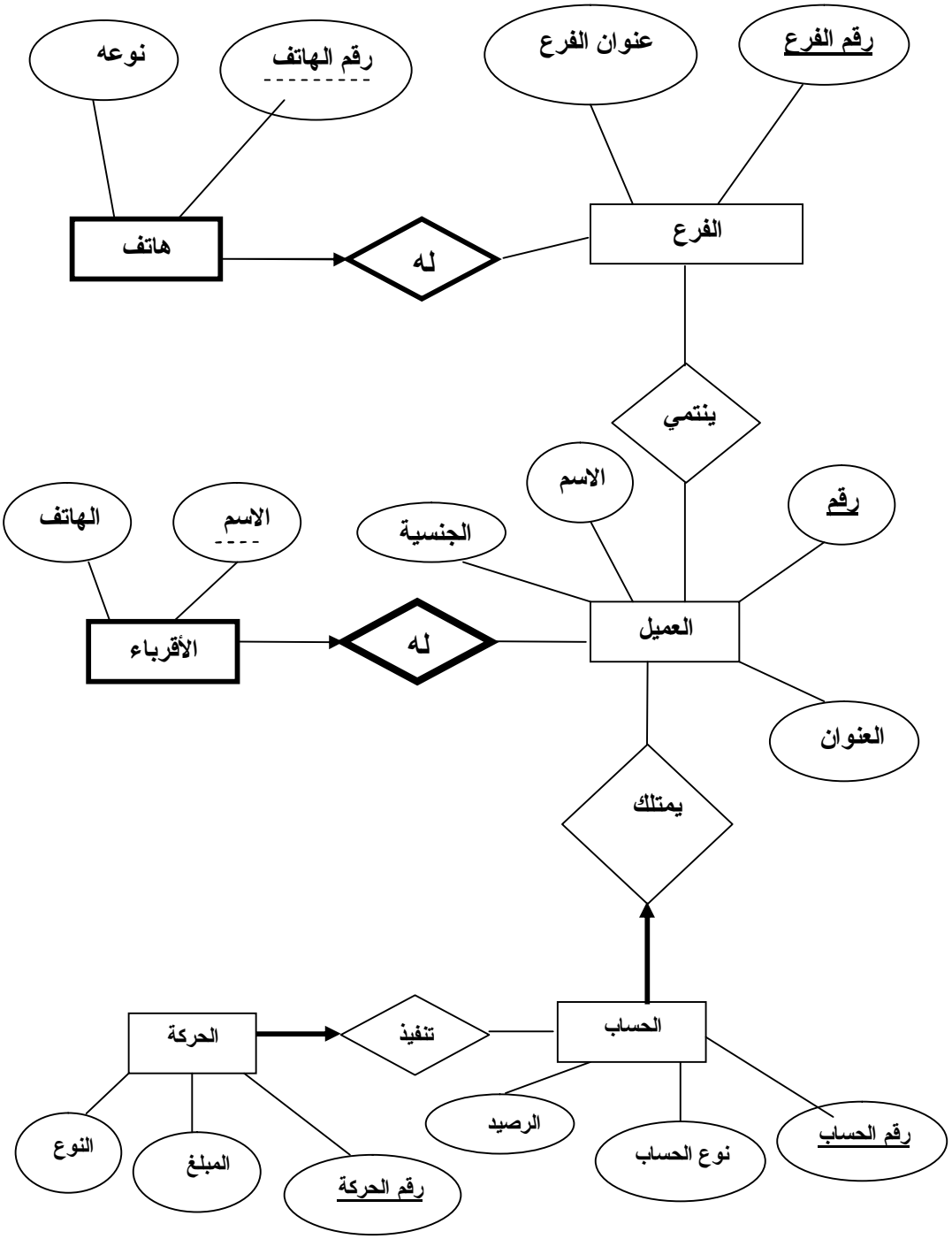
أ- الحساب يطلب ، عميلاً .

ب- الحركة تطلب حساباً .

ت- هاتف الفرع يطلب فرعاً .

ث- القريب يطلب عميلاً .

مخطط ER :


نوعه
رقم الهاتف
عنوان الفرع
رقم الفرع
هاتف
له
الفرع
ينتمي
الاسم
رقم
الهاتف
الاسم
الجنسية
العميل
الأقرباء
له
العنوان
يمتلك
الحركة
تنفيذ
الحساب
النوع
المبلغ
رقم الحركة
الرصيد
نوع الحساب
رقم الحساب

أسئلة تقويم ذاتي



1/ وضح المقصود بقيد التغطية.

8. النموذج العلائقي Relational Model

عزيزي الدارس، يعتبر النموذج العلائقي نموذجاً واسع الانتشار. وهنالك عدة شركات تقوم بتصميم أنظمة إدارة قواعد البيانات DBMS. و هنالك عدد كبير من الشركات الكبيرة تستخدمه مثل IBM ، Informix، Oracle, Microsoft. و تتكون قاعدة البيانات العلائقية Relational Database من مجموعة من العلاقات Relations ، وكل علاقة تتمثل في جزئين كما يلي:

أ- الحالة Instance:

هي عبارة عن حالة الجدول من بيانات، حيث يحتوي على عدد من الصفوف وعدد من الأعمدة .

مثال لـــInstance :

Name	Age	Level
Mohamed	22	3
Ahmed	20	1
Ibrahim	25	5

ب- المخطط Scheme :

و فيه يتم تحديد اسم العلاقة Relation Name و كذلك الخصائص Attributes الخاصة بالعلاقة.

مثال لـــSchema :

Student (Name: string, Age: integer, Level: integer)

1.8 لغة الاستفسار (الاستعلام) العلائقية

Relational Query Language

عزيزي الدارس، لغة الاستفسار العلائقية تعتبر احدى الأدوات الأساسيات الخاصة بالنموذج العلائقي. وبطريقة بسيطة وأداء قوي نستطيع أن نستعلم عن البيانات المخزنة في الجداول Tables. و يمكن لـDBMS أن تغير خطة الحل من خلال إعادة ترتيب شكل الاستعلام حتى نحصل على نسبة عالية من الأداء علما بأن النظام يمتلك وحدة تسمى Query Optimizationلعمل ذلك.

2.8 لغة الاستعلام الهيكلية

Structured Query Language (SQL)

هي عبارة عن لغة قياسية يتعامل معها و يدعمها عدد من الأنظمة. و نستطيع من خلالها أن نتعامل مع قاعدة البيانات مثل أن نستفسر عن بيانات معينة (Retrieving Data)، أو إدخال بيانات جديدة (Inserting Data) ، أو حذف بيانات (Deleting Data) ، أو تعديل بيانات (Modifying Data) أصلاً موجودة، و في هذا الكتاب سوف نتناول هذه اللغة بمزيد من التفصيل. وتم تطوير هذه اللغة من قبل شركة IBM عام 1970 م ونظرا لعمل هذه اللغة في عدة تطبيقات جاءت الحاجة لعمل نسخ قياسية، ولذلك نجد أن هنالك عدة نسخ لهذه اللغة منها :

SQL 86 و SQL89و SQL 92 و SQL99.

1.2.8 الاستعلامات

المقصود بالاستعلام هو عرض بيانات مخزنة في قاعدة البيانات دون تغير محتوى هذه البيانات. مثلاً لإيجاد بيانات الطلاب الذين أعمارهم تساوي 18 سنة نكتب الاستعلام التالي :

```
Select * form   students  S
 Where   S.age   =18 ;
```

وإذا كنا نريد أن نستعرض بعض الحقول وليس كل الحقول مثـل name و age فإننا نستبدل السطر الأول من الاستعلام بالشكل :

```
Select S. name , S.age
Where S.age =18 ;
```

ويمكن أن يقرأ الاستعلام من عدة جداول.

2.2.8 إنشاء الجداول

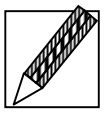
الجدول هو عبارة عن الوعاء الذي يخزن البيانات. و يـتم إنشـاؤه بواسـطة جملـة Create في لغة SQL

مثال(1)

أ- لإنشاء جدول Student نكتب الأمر التالي :

```
Create Table Student
( Sid : char (20 ) ,
( name : char (20) ,
Login : char (10) ,
age : integer ,
gpa : real )
```

تدريب (1)



اكتب جملة SQL لإنشاء جـدول اسـمه Book يحتـوي علـى الحقول التالية : رقم الكتاب ، اسم الكتاب ، سعر الكتاب .



ونلاحظ في عملية إنشاء الجداول أننا نحدد نوع بيانات الحقل وكذلك المجال . وعند إدخال أي سجل أو تعديل سجل موجود أصلا نجد أن DBMS تقوم بالتأكد من أن البيانات المدخلة تتوافق مع نوع الحقول . وإذا لم يكن هنالك توافق فسيتم رفض هذه البيانات.

ب- حذف وتعديل الجداول :

عند كتابة الأمر التالي :

Drop table Students

نجد أن هيكل البيانات(جدول) وكذلك البيانات تحذف وبشكل نهائي .

و الأمر

Alter table Students
Add column (Address: integer)

هذا الأمر سوف يعدل جدول Students ويضيف حقلاً جديداً يسمى Address على مستوى كل السجلات، ويكون محتوى الحقل فارغاً (Null) وتعني القيمة فارغة .

ت- القيود التكاملية (I C) Integrity Constraints :

هي عبارة عن قيود تفرض على البيانات قبل تخزينها في الجداول ويتم تعريف القيود التكاملية في المخطط Schema . نقول إن الجدول شرعي أو سليم_(Legal) إذا كان متحققاً فيه كل الشروط التكاملية(IC) ويجب أن لا تسمح DBMS بوجود جداول لا تتحقق فيها تلك الشروط (القيود). ويتم وضع هذه القيود لكي تمثل معاني للبيانات في ارض الواقع وكذلك لتفادي الأخطاء عند الإدخال .

ث- المفتاح الأساسي Primary Key :

هو عبارة عن مجموعة من الحقول موجودة في Relation العلاقة ويجب أن يتحقق فيه الآتي :

1. أن يكون أقل مجموعة ممكنة عن طريقها نستطيع الوصول إلى أي سجل ، وأيضا يجب إن لا يتكرر على مستوى جميع الحقول. والمجموعة التي تحتوي على المفتاح الأساسي نطلق عليها Super key.

- وعلى سبيل المثال في جدول الطلاب Students :

المفتاح Key هو Sid .

ونطلق علي المجموعة { Sid , name } ⟶ Super Key .

و يتم اختيار المفتاح الأساسي من قبل DBA .

ج- المفتاح الأساسي والمفاتيح المرشحة Primary Key and Candidate Keys :

يمكن أن يكون لدينا عدد من المفاتيح تصلح أن تكون مفاتيح أساسية . ولكن يجب أن نختار واحداً فقط من هذه المفاتيح ، ويتم هذا الاختيار بواسطة DBA .

ملاحظة

الفرق بين المفتاح الأساسي Primary Key و قيد عدم التكرار Unique هو : أن Unique يمكن أن يكون فارغاً (Null)، بينما المفتاح الأساسي لا يتكرر و لا يكون فارغاً.

ح- المرجعية و المفتاح الأجنبي Foreign Key , Referential Integrity:

المفتاح الأجنبي هو عبارة عن مجموعة من الحقول في جدول واحد، ونستطيع من خلال هذه المجموعة الوصول إلى سجلات موجودة في جدول آخر. ويجب أن تكون هذه الحقول في الجدول الآخر بمثابة مفاتيح أساسية. وهي تشبه المؤشرات .

المثال التالي يوضح ذلك

Students

Sid	name	Age
10	Mohammed	18
20	Ali	20
30	Ahmed	30
40	Othman	22

Enrollment

Sid	Cid	Grade
10	1000	A
20	2000	B
30	3000	C
10	4000	D

Enrollment يمثل جدول تسجيل المقررات للطلاب حيث Cidهو رقم المقرر.

ومثال على المفتاح الأساسي في جدول الطلاب نجد أن حقل رقم الطـــلاب (Sid) هـــو عبارة عن مفتاح أساسي في جدول الطلاب . أما في جدول التســجيل نجــد أن رقــم الطالب هو عبارة عن مفتاح أجنبي . ويعتبر هذا الحقل (رقم الطالــب فـــي جــدول التسجيل) مرجعاً لبيانات الطالب . والحقل الرابط بين الجدولين هـــو شـــبيه بالارتبـــاط التشعيبي في صفحات (HTML) .

- ***تحقيق المرجعية بين الجداول Enforcing Referential Integrity :***

– إذا كان رقم الطالب في جدول التسجيل عبارة عن مفتاح أجنبـــي ويمثـــل مرجعـــاً لبيانات الطلاب في جدول الطلاب Students فنجد أنه في حالة إدخال بيانات تســـجيل مادة لطالب غير موجود في جدول الطلاب سوف يـــرفض النظـــام تســـجيل المـــادة للطالب.

– ماذا يحدث إذا أردنا أن نحذف الطالب الذي سجلت له مواد؟

قد تحدث واحدة من الاحتمالات التالية:

i. كل بيانات التسجيل الخاصة بالطالب سوف تحذف .

ii. يرفض حذف السجل .

iii. وضع قيمة افتراضية لقيمة الحقل (Sid) في جدول تسجيل المواد .

iv. وضع قيمة فارغة (Null) في حقل (Sid). وقيمة (Null) المقصود بها أن القيمة غير معروفة أو لم يتم إدخالها .

خ- المناظر Views :

- المنظر هو عبارة عن علاقة ولكننا نقوم بتخزين تعريف المنظر بدلا من تخزين السجلات . فإذا أردنا أن نعمل منظراً يحتوي على اسم الطالب ومعدله بشرط أن لا يتجاوز عمر الطالب (21سنة) نكتب الأمر التالي:

```
Create View Std_gpa
as Select s.name , s.gpa from Student where s.age <= 21
```

- وأيضا نستطيع أن نقوم بحذف view عن طريق الأمر (Drop view) .

مثل:

```
Drop View Std_gpa
```

د- المناظر وأمن البيانات Views and Security :

نجد أن المناظر تستخدم لعرض البيانات التي نحتاجها فقط، أي أنها ملخص . وأيضاً تستخدم لحجب بعض البيانات التي لا نريد أن نظهرها . فمثلا المنظر للطلاب ذوي الأعمار الصغيرة (السابق) هو عرض لبيانات معينة وليست كل البيانات . وهذه الآلية تساعد في أمن وسرية البيانات.

أسئلة تقويم ذاتي



1/ اشرح مكونات قاعدة البيانات العلائقية.
2/ عرّف لغة الاستعلام الهيكلية .
3/ بين المقصود بالاستعلام .
4/ اذكر جملة إنشاء الجدول في لغة SQL .
5/ اشرح بالمثال عملية حذف وتعديل الجداول في لغة SQL .
6/ عرِّف اللآتي : أ/القيود التكاملية (I C) . ب/ المفتاح الأجنبي ج/ المنظر .
7/ وضِّح الفرق بين المفتاح الأساسي Primary Key و قيد عدم التكرار Unique .

3.8 التصميم المنطقي لقاعدة البيانات

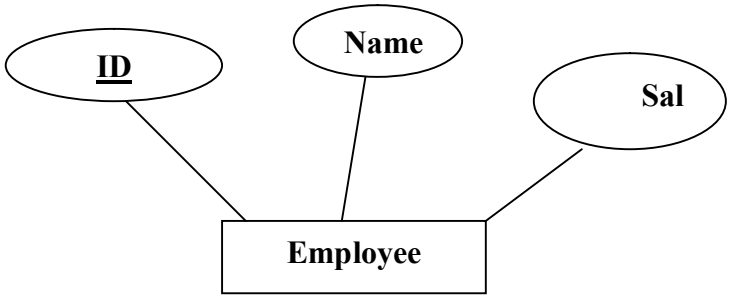
Logical Database Design

عزيزي الدارس، لتصميم قاعدة البيانات نقوم بتحويل مخطط ER بواسطة لغة الـSQL إلى النموذج العلائقي (مجموعة من الجداول) . و تكون عملية التحويل بخطوات محددة، حيث إن لكل جزء حسب نوعه في مخطط ER ما يقابله من جدول. و في ما يلي نشرح هذه الخطوات:

أ) تحويل الكيان للنموذج العلائقي Translate Entity Set to Relational Model

تمثل مجموعة الكيان في النموذج العلائقي بجدول، حيث إن كل الخصائص تمثل حقول الجدول، و مفتاح مجموعة الكيان تمثل المفتاح الأساسي في الجدول.

و مثال مجموعة كيان الموظف يمكن أن تحول إلى جدول كما يلي:



كيان الموظف Employee Entity

```
Create Table Employee
(
        ID              Number(10) ,
        Name            Char(20) ,
        Sal             Number(6,2) ,
        Constraint EmpPK  Primary Key(ID)
)
```

ب) تحويل مجموعة العلاقة إلى النموذج العلائقي

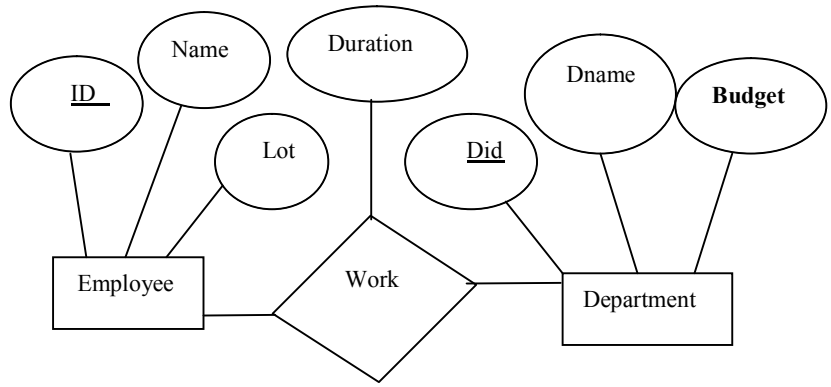
Translate Relationship Set to Relational Mode

تحول مجموعة العلاقة في النموذج العلائقي في شكل جدول، و يكون الجدول من الخطوات التالية :

- أي مفتاح أساسي في مجموعة الكيان المشاركة في العلاقة تعتبر مفتاحاً أجنبياً في جدول العلاقة.
- والمفتاح الأساسي في جدول العلاقة هو عبارة عن الحقول التي تمثل المفاتيح الأساسية في مجموعات الكيانات التي دخلت في هذه العلاقة .
- كل الحقول الوصيفية للعلاقة هي أيضا حقول في جدول العلاقة.

تدريب (2)

اكتب جدول العلاقة (Work) من المخطط التالي.

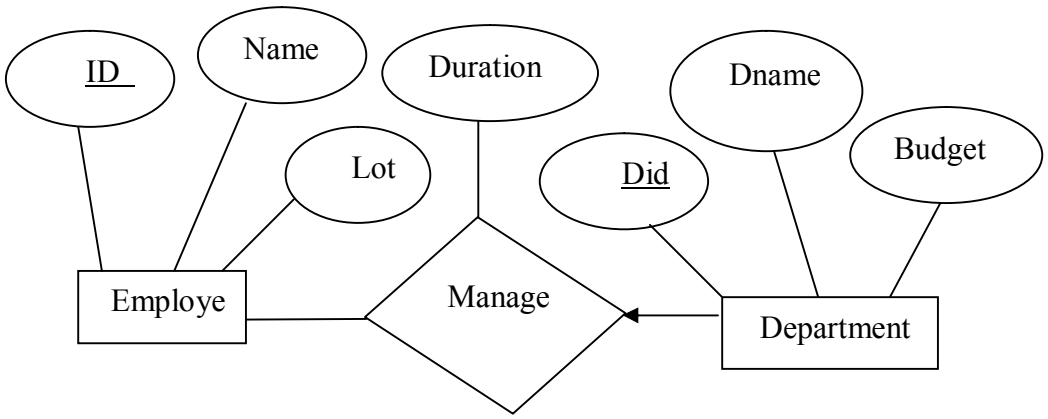


ت) تحويل قيد المفتاح إلى النموذج العلائقي Translate the Key Constraint to Relational Model

عند تحويل العلاقة التي يوجد بها key constraint فإننا نقوم بعمل التالي :

- نستخدم نفس الخطوات في الحالة السابقة في تحويل مجموعة العلاقة ولكن هنالك فرق في اختيار المفتاح الأساسي، حيث إن المفتاح الأساسي للعلاقة سيكون هو نفس المفتاح الأساسي لمجموعة الكيان التي تطلب كياناً واحداً على الأكثر من مجموعة الكيانات المشاركة في العلاقة .

المثال التالي يوضح ذلك:



في هذا المثال نجد أن كيان الإدارة تطلب موظفاً واحداً على الأكثر لكي يدير الإدارة. و لتحويل مثل هذا النوع من العلاقة نتبع نفس الحل السابق، و لكن يكون هنالك فرق واحد هو أن المفتاح الأساسي يكون نفس المفتاح الأساسي الخاص بمجموعة كيـــان الإدارات، و يكون الحل كما يلي :

```
Create Table Work
(
        ID      Number(10) ,
        Did     Number(3) ,
        Dname Char(20) ,
        Budget Number(6,2),
        Duration Date ,
        Constraint EmpFK1 Foreign Key (ID) References
Employee(ID) ,
        Constraint EmpFK2 Foreign Key (Did) References
Department(Did) ,
        Constraint EmpPK Primary Key (Did)
)
```

تدريب (3)

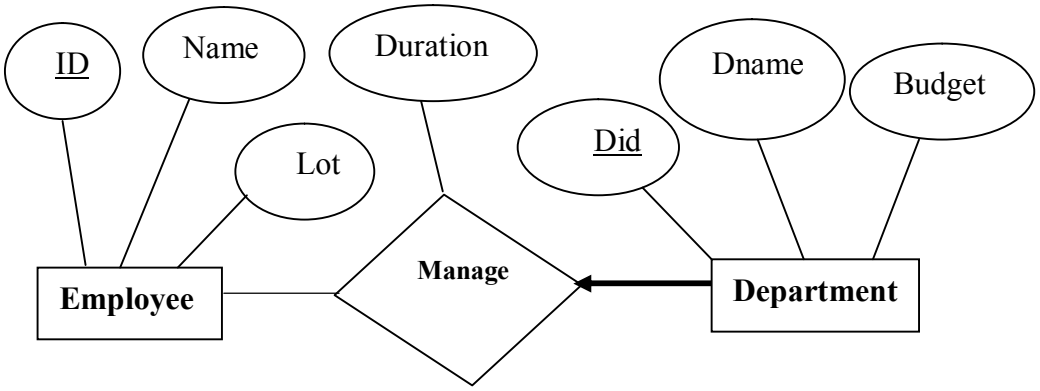
من المثال السابق بما أن جدول الإدارات و جدول العلاقة لهما نفس المفتاح الأساسي هل يمكن أن يكون هنالك حل آخر؟ وضّح ذلك.

ث) تحويل القيد التشاركي إلى النموذج العلائقي

Translate the Participation Constraint to Relational Model

إذا كان هنالك Participation constraint Total فذلك يعني أنه لابد من وجود قيمة تمثل الكيان الآخر . ولتمثيل قيد Total يجب أن نكتب Not Null أمام الحقل الذي يمثل الكيان الآخر .

و المثال التالي يوضح ذلك:



لتحويل القيد التشاركي نتبع نفس الخطوات السابقة بالإضافة إلى وضع القيد Not Null أمام الحقل الذي يمثل كيان الموظف، بحيث يكون الجدول كما يلي:

```
Create Table Work_Dept
(
        Did         Number(3) ,
        Dname             Char(20),
        Budget      Number(6,2) ,
```

```
    Duration      Date,
    ID            Number(10)  Not Null,
    Constraint EmpFK Foreign Key (ID) References Employee(Did) ,
    Constraint EmpPK Primary Key (Did)
)
```

ج) تحويل مجموعة الكيان الضعيف إلى النموذج العلائقي

Translate Weak Entity Set to Relational Model

نجد أن الكيان الضعيف يكون له علاقة مع الكيان الرئيسي و هو لا يستطيع أن يعـرف نفسه بنفسه . ولتحويل الكينونة الضعيفة إلى النموذج العلائقي نجد أن عملية التحويـل سوف ينتج منها أن الكيان الضعيف والعلاقة سوف تصبح في جدول واحـد، علـى أن يحتوي هذا الجدول **على الآتي :**

① كل الخواص التي توجد في الكيان الضعيف .

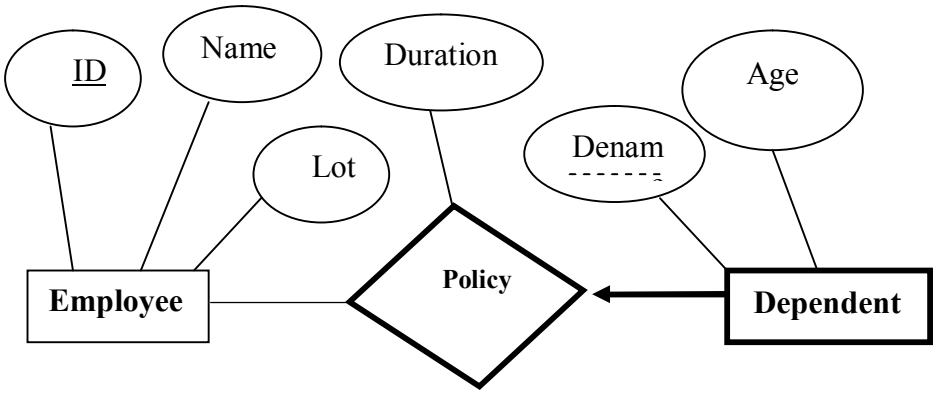
② كل الخواص التي توجد في العلاقة .

③ يجب وضع قيد على حقل المفتاح الخاص بالكيان الرئيسي، ويجب أن يكـون غيـر فارغ Not Null.

④ المفتاح الأساسي مكون من حقل المفتاح الجزئي لمجموعة الكيان الضعيف والمفتاح الأساسي لمجموعة الكيان الرئيسي.

⑤ المفتاح الأجنبي عبارة عن حقل المفتاح الأساسي في الكيان الرئيسي.

و المثال التالي يوضح ذلك:



نجد الكيان الضعيف و العلاقة ستشكل جدولاً واحداً كما يلي:

```
Create Table Plolicy_depend
(
        Dename      Char(20) ,
        Age         Number(10),
        Duration    Date,
        ID      number(10)  Not Null,
        Constraint FK  Foreign Key (ID) References Employee(ID),
        Constraint PK Primary Key (Dename , ID)
)
```

أسئلة تقويم ذاتي

1/ اشرح خطوات عملية تحويل مخطط ER بواسطة لغة SQL إلى النموذج العلائقي (مجموعة من الجداول).

الخلاصة

عزيزي الدارس ،عُرِض في هذه الوحدة الموضوعات الأساسية التالية:

* عرفت الوحدة قاعدة البيانات بأنها عبارة عن وعاء لمجموعة من البيانات تعكس نشاط منظمة أو شركة أو هيئة ذات نشاط محدد .

* كذلك عرفت الوحدة نظم إدارة قواعد البيانات بأنها مجموعة من الحزم البرمجية مهمتها إدارة و تخزين قواعد البيانات.

* استعرضت الوحدة أنواع قواعد البيانات منها : قواعد البيانات للوسائط المتعددة، أنظمة المعلومات الجغرافية، ومخازن البيانات .

* بينت الوحدة مزايا نظم إدارة قواعد البيانات منها : التحكم في ، حماية البيانات، كفاءة الوصول للبيانات، استقلالية البيانات، إدارة البيانات، تزامن الوصول ومعالجة الأخطاء .

* قدمت الوحدة مقارنة بين نظام الملفات و نظم إدارة قواعد البيانات وبينت أن نظام إدارة قواعد البيانات يستطيع أن يتعامل مع كم هائل من البيانات بقوة وكفاءة عالية وإمكانيات تستطيع أن تستوعب نمو البيانات .

* استعرضت الوحدة المخططات التجريدية : المخطط التصوري، المخطط الفعلي، المخطط الخارجي .

* بينت الوحدة الأشخاص الذين يتعاملون مع **DBMS** : المستخدمون، الباعة مبرمجو تطبيقات قواعد البيانات، مدير قاعدة البيانات .

* عرَّفت الوحدة الكيان بأنه عبارة عن كائن في العالم الحقيقي ويمكن أن يميز عن بقية الكائنات و يعرف نفسه بنفسه و لا يعتمد وجوده على كيان آخر.

* كذلك عرَّفت الوحدة مجموعة الكيانات بأنها عبارة عن مجموعة من الكيانات المتشابهة ويجب أن يتحقق الأتي : كل الكيانات في مجموعة الكينونة لديها نفس الخواص – لكل مجموعة كيان مفتاح **Key** – أي خاصية من الخواص لديها مجال أو

Domain، ونستخدم الخواص لوصف الكيان، أما المفتاح فيستخدم للوصول إلى بقية الخواص الموجودة في الكيان .

* بينت الوحدة أن العلاقة تستخدم للربط بين كيانين أو أكثر ، وأنه يمكن أن تكون هنالك خصائص تصف العلاقة وتوضح العلاقات المتشابهة في مجموعة العلاقة .

* وضحت الوحدة أن قيد المفتاح يستخدم لتوضيح نوع العلاقة بين الكيانات وهي أن يطلب الكيان كياناً واحداً على الأكثر في العلاقة .

* بينت الوحدة أنه تصميم قاعدة البيانات يجب أن يكون لها تصور واضح قبل تصميمها و لوضع هذا التصور نستخدم نموذج الكيانات العلائقي **Entity Relationship Model (ER)**.

ويجب أن نمثل أي مخطط **Schema** بواسطة **ER Diagrams**[وأيضا يجب أن نحول أي نقطة أو أي جزء من **ER Diagrams** إلي النموذج العلائقي **Relational Schema** باستخدام جمل **SQL** .

*عرفت الوحدة المفتاح **Key** بأنه عبارة عن أقل مجموعة ممكنة من الخواص وبصورة وحيدة تعرف الكيان .

* بينت الوحدة المقصود بقيد التغطية وهو أن أي كيان موجود في الكيان الرئيسي يجب أن يكون موجود في إحدي الكيانات الفرعية.

* بينت الوحدة أن قاعدة البيانات العلائقية تتكون من مجموعة من العلاقات، وكل علاقة تتمثل في جزئيين هما: **الحالة Instance**:عبارة عن حالة الجدول من بيانات حيث يحتوي على عدد من الصفوف وعدد من الأعمدة – **والمخطط Scheme** : و فيه يتم تحديد اسم العلاقة Relation Name و كذلك الخصائص Attributes الخاصة بالعلاقة.

*عرفت الوحدة لغة الاستعلام الهيكلية (SQL) :بأنها عبارة عن لغة قياسية يتعامل معها و يدعمها عدد من الأنظمة و نستطيع من خلالها أن نتعامل مع قاعدة البيانات

مثل أن نستفسر عن بيانات معينة ، أو إدخال بيانات جديدة ، أو حذف بيانات ، أو تعديل بيانات أصلاً موجودة .

* كما بينت الوحدة أن المقصود بالاستعلام هو عرض بيانات مخزنة في قاعدة البيانات دون تغير محتوى هذه البيانات.

* أوضحت الوحدة أن الجدول هو عبارة عن الوعاء الذي يخزن البيانات. و يتم إنشاؤه بواسطة جملة Create في لغة SQL .

* بينت الوحدة أن القيود التكاملية (I C) هي عبارة عن قيود تفرض على البيانات قبل تخزينها في الجداول، ويتم تعريف القيود التكاملية في المخطط Schema .

* أوضحت الوحدة أن المنظر هو عبارة عن علاقة، ولكننا نقوم بتخزين تعريف المنظر بدلا من تخزين السجلات .

* ذكرت الوحدة أنه لتصميم قاعدة البيانات نقوم بتحويل مخطط ER بواسطة لغة الـSQL إلى النموذج العلائقي (مجموعة من الجداول). و تكون عملية التحويل بخطوات محددة .

لمحة مسبقة عن الوحدة التالية

عزيزي الدارس ، فرغت من دراسة الوحدة الأولى التي عالجت موضوع أساسيات قواعد البيانات العلائقية، وسوف تتعرف بعون الله في الوحدة التالية من هذا المقرر على موضوع " التطبيق بواسطة لغة **SQL** "، حيث نبدأ بشرح كيفية الاتصال بقاعدة البيانات،ونتناول جمل لغة **SQL**، وهذه الجمل تقسم حسب طبيعة مهامها، ونتناول الأقسام الرئيسية لهذه الجمل ، كما نتناول قيود السلامة.

آمل أن تجدها وحدة مفيدة،. ووفقك الله

حلول التدريبات

تدريب (1)

الحل:

```
Create Table   Book
(no: integer  ,
name  :  char  (20 ),
Price:  real)
```

تدريب (2)

```
Create Table Work
(
        ID Number(10) ,
        Did Numeber(3) ,
        Duration Date ,
        Constraint  EmpFK1  Foreign  Key  (ID)  References
Employee(ID) ,
        Constraint  EmpFK2  Foreign  Key  (Did)  References
Department(Did) ,
        Constraint EmpPK Primary Key (ID,Did)
)
```

تدريب (3)

يمكن أن ندمج الجدولين في جدول واحد ويكون الحل كما يلي:

```
Create Table Work_Dept
(
        ID              Number(10) ,
        Did             Number(3) ,
        Dname                   Char(20),
        Budget          Number(6,2) ,
        Duration        Date,
        Constraint   EmpFK   Foreign   Key   (ID)   References
Employee(Did) ,
        Constraint EmpPK Primary Key (Did)
```

مسرد المصطلحات

قاعدة البيانات (Database)

عبارة عن وعاء لمجموعة من البيانات تعكس نشاط منظمة أو شركة أو هيئة ذات نشاط محدد فعلي.

نظم إدارة قواعد البيانات (DBMS)

مجموعة من الحزم البرمجية (**Software Packages**) مهمتها إدارة و تخزين قواعد البيانات.

قواعد البيانات للوسائط المتعددة (Multimedia database)

يتيح هذا النوع من قواعد البيانات تخزين البيانات المتمثلة في الصورة ومقاطع الفيديو والرسائل الصوتية، و توفر آلية للتعامل مع هذا النوع من البيانات لكي يتسنى التعامل معها.

أنظمة المعلومات الجغرافية (Geographic information system)

وهي معنية بتخزين وتحليل الخرائط الجغرافية والصور المختلفة من الأقمار الصناعية .

مخازن البيانات (DATAWARE HOUSE)

وهي مستخدمة من قبل الشركات الكبيرة وتحتوي علي بيانات ضخمة وتتيح عملية استخلاص وتحليل البيانات للمساعدة في عملية اتخاذ القرار .

الكيان Entity

هي عبارة عن كائن في العالم الحقيقي، ويمكن أن يميز عن بقية الكائنات و يعرف نفسه بنفسه، و لا يعتمد وجوده على كيان آخر.

مجموعة الكيانات Entity Set

عبارة عن مجموعة من الكيانات المتشابهة، ويجب أن يتحقق الآتي : كل الكيانات في مجموعة الكينونة لديها نفس الخواص، لكل مجموعة كيان مفتاح **Key** ، أي خاصية من الخواص لديها مجال أو **Domain** ، ونستخدم الخواص لوصف الكيان، أما المفتاح فيستخدم للوصول إلى بقية الخواص الموجودة في الكيان .

المفتاح Key

وهو عبارة عن أقل مجموعة ممكنة من الخواص وبصورة وحيدة تعرف الكيان .

الكيان الضعيف Weak Entity

هو عبارة عن كيان لا يستطيع أن يعرف نفسه، بنفسه و يعتمد وجوده على كيان آخر.

لغة الاستعلام الهيكلية Structured Query Language (SQL)

هي عبارة عن لغة قياسية يتعامل معها و يدعمها عدد من الأنظمة، ونستطيع من خلالها أن نتعامل مع قاعدة البيانات، مثل أن نستفسر عن بيانات معينة (Retrieving Data) أو إدخال بيانات جديدة (Inserting Data) ، أو حذف بيانات (Deleting Data) ، أو تعديل بيانات (Modifying Data) أصلاً موجودة.

الاستعلامات

المقصود بالاستعلام هو عرض بيانات مخزنة في قاعدة البيانات دون تغير محتوى هذه البيانات.

الجدول

هو عبارة عن الوعاء الذي يخزن البيانات. و يتم إنشاؤه بواسطة جملة Create في لغة SQL.

القيود التكاملية Integrity Constraints (I C)

هي عبارة عن قيود تفرض على البيانات قبل تخزينها في الجداول .

المفتاح الأساسي Primary Key

هو عبارة عن مجموعة من الحقول موجودة في Relation العلاقة .

المفتاح الأجنبي

هو عبارة عن مجموعة من الحقول في جدول واحد ، ونستطيع من خلال هذه المجموعة الوصول إلى سجلات موجودة في جدول آخر.

المناظر Views

المنظر هو عبارة عن علاقة ولكننا نقوم بتخزين تعريف المنظر بدلا من تخزين السجلات .


الوحدة الثانية
التطبيق بواسطة لغة SQL

محتويات الوحدة

المحتوى	رقم الصفحة
المقدمة	**49**
تمهيد	**49**
أهداف الوحدة	**50**
1. الاتصال بقاعدة البيانات	**51**
2. جُمل لغة SQL	**52**
1.2 جُملة SELECT	**54**
2.2 لغة التعامل مع البيانات	**112**
3.2 إدارة و إنشاء الجداول	**122**
3. قيود السلامة	**127**
الخلاصة	**132**
إجابات التدريبات	**136**
مسرد المصطلحات	**142**
المراجع	**144**

المقدمة

تمهيد

عزيزي الدارس،

نرحب بك إلى الوحدة الثانية من مقرر " تطبيقات قواعد البيانات" والتي تأتي بعنوان "**التطبيق بواسطة لغة SQL** " تحتوي الوحدة على ثلاثة أقسام: القسم الأول يبدأ بشرح كيفية الاتصال بقاعدة البيانات ، القسم الثاني يتناول جمل لغة **SQL** وهذه الجمل تقسم حسب طبيعة مهامها، ويتناول القسم الأقسام الرئيسية لهذه الجمل ، وهي : جملة SELECT وتستخدم هذه الجملة لاسترجاع البيانات من قاعدة البيانات، وكذلك جُمل التعامل مع البيانات Data Manipulation Language (MDL) وهي : INSERT – UPDATE– DELETE و هي معنية بعمليات الإدخال و التعديل و الحذف للبيانات، وجُمل تعريف البيانات Data Definition Language (DDL) و هي : CREATE – ALTER – DROP و هي معنية بإنشاء الجداول Tables و تعديلها و حذفها. أما القسم الثالث والاخير من الوحدة فيتناول قيود السلامة.

عزيزي الدارس، اشتملت هذه الوحدة على عدد كبير من اسئلة التقويم الذاتي والأمثلة المحولة بغرض إعانتك على الفهم، نرجو الاهتمام بها، ووفقك الله دائماً.

أهداف الوحدة



عزيزي الدارس، بعد فراغك من دراسة هذه الوحدة ينبغي أن تكون قادراً على أن:

- **تشرح** كيفية الاتصال بقاعدة البيانات.
- **تذكر** العمليات التي يمكن تنفيذها من خلال جملة SELECT.
- **تستخدم** دالة LOWER لتحويل حالة الأحرف إلى أحرف صغيرة.
- **تذكر** الدوال التي تتعامل مع الأحرف.
- **تستخدم** الدالة SYSDATE لاسترجاع التاريخ الحالي للجهاز.
- **تشرح** باستخدام الأمثلة الدوال التي تتعامل مع التاريخ.
- **تستخدم** دوال التحويل لتحويل بيانات الحقل من نوع إلى نوع آخر.
- **تعدد** أنواع الاستعلام بشرط الربط بين الجداول.
- **توضح** الدوال التي يمكن استخدامها على مجموعة من السجلات، واستخدام كل منها .
- **تستخدم** العبارة Group By.
- **تشرح** كيفية عمل الشرط على مستوى المجموعة باستخدام Having .
- **تحل** مسائل باستخدام الاستعلام الفرعي Subqueries.
- **تستخدم** جملة الإدخال INSERT لإدخال السجلات داخل قاعدة البيانات.
- **تستخدم** جملة UPDATE لتعديل بيانات السجلات.
- **تستخدم** جملة DELETE لحذف السجلات والصيغة العامة.
- **تعدد** بعض أنواع بيانات الحقول.
- **توضح** الصيغ التي تستخدم في عملية تعديل الجدول.
- **تعدد** أنواع قيود السلامة .

توطئة

عزيزي الدارس،

تعتبر لغة SQL في نظام أوراكل لغة مستوفية لكل الشروط القياسية و التي تم اعتمادها بواسطة معهد القياسيات القومي الأمريكي(American National Standards Institute (ANSI وهيئة القياسات العالمية International Standards Organization (ISO)، حيث إن المعهد ANSI والعياة ISO اعتمدا لغة الـSQL كلغة قياسية للتعامل مع قواعد البيانات العلائقية Relational Database.

في هذا الكتاب سيتم استخدام برنامج SQL-Plus لتفيذ الجمل المختلفة، علماً بأن قاعدة البيانات المستخدمة أوراكل Oracle 9*i*.

1. الاتصال بقاعدة البيانات Database Connection

عزيزي الدارس،

حتى يتمكن المستخدم من الاتصال بقاعدة البيانات لابد أن تكون له صلاحية Privilege لعمل ذلك. و الشكل (1-1) يوضح عملية إدخال المستخدم لاسم المستخدم User ID و كلمة المرور Password واسم قاعدة البيانات Database Name حتى يتمكن من الولوج إلى قاعدة البيانات.



الشكل (1-1) شاشة الاتصال مع قاعدة البيانات

كما هو موضح في الشكل فإن المستخدم هنا SCOTT وهو منشأ من قبل نظام أوراكل بغرض التدريب. وكلمة المرور هي Tiger واسم قاعدة البيانات هي DBTarig.

بعد إدخال المستخدم لكلمة المرور بصورة ناجحة ينتقل البرنامج للشاشة في الشكل (2-1)، وهي تدل على أن المستخدم قد اتصل بقاعدة البيانات بنجاح.



الشكل (2-1)

وكما يظهر في الشاشة يتستقبل الموجه SQL> الأوامر الموجهة لقاعدة البيانات حتى يتم تنفيذها بواسطة نظام أوركل.

أسئلة تقويم ذاتي



1/ اشرح كيفية الاتصال بقاعدة البيانات.

2. جُمل لغة SQL

عزيزي الدارس، لكي يستطيع المستخدم التعامل مع قاعدة البيانات فإنه ملزم باستخدم جُمل الـSQL ويجب أن يلتزم بمفردات اللغة وكذلك القوعد التركيبية (Syntax Structures). وتقسم تلك الجُمل حسب طبيعة مهامها. والجدول (1–1) يوضح تلك الجُمل ومهام كل قسم.

الجُمل Statements	الوصف Description
جملة SELECT	تستخدم هذه الجملة لاسترجاع البيانات من قاعدة البيانات.
الجُمل: INSERT UPDATE DELETE	تعرف هذه الجُمل بجُمل التعامل مع البيانات Data Manipulation Language (MDL) وهي معنية بعمليات الإدخال والتعديل والحذف للبيانات.
الجُمل CREATE ALTER DROP	تعرف هذه الجُمل بجُمل تعريف البيانات Data Definition Language (DDL) وهي معنية بإنشاء الجداول Tables وتعديلها وحذفها.

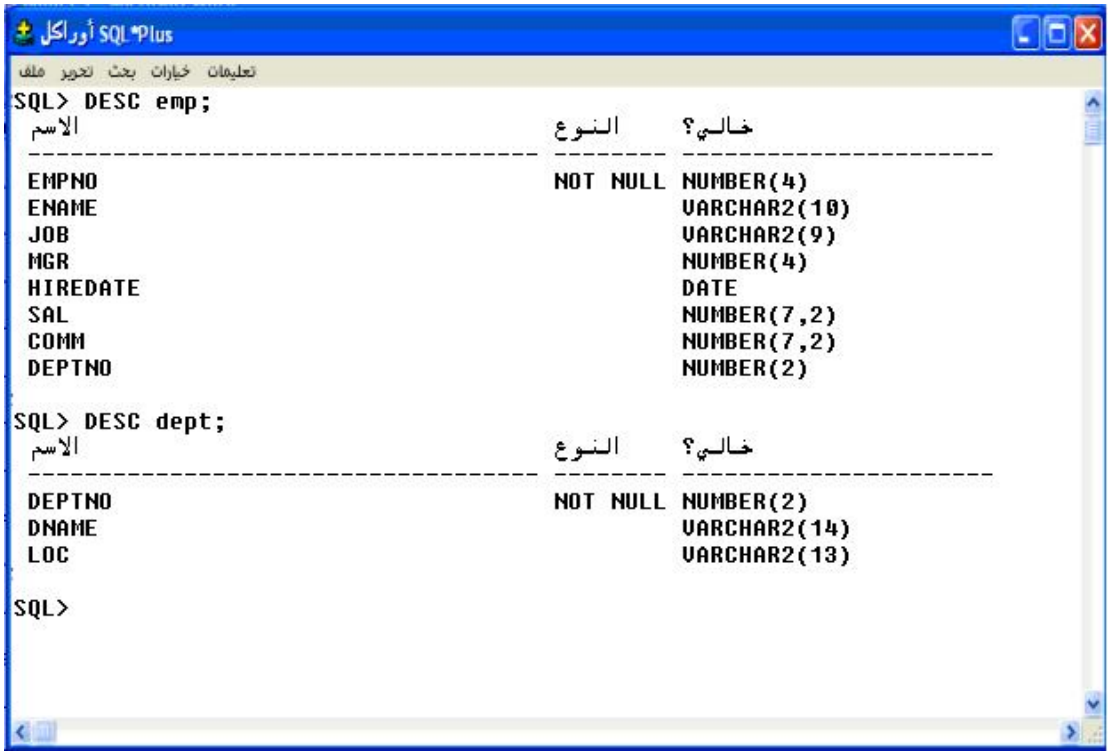
الجدول (1-1) جُمل لغة SQL



هنالك أقسام أخرى متعلقة بالتحكم و منح الصلاحيات، ولكن في هذا الكتاب سوف نتناول فقط الأقسام الرئيسية التي وردت في الجدول أعلاه.

الجداول المستخدمة في الكتاب

سوف نستخدم عدد من الجداول تم إنشاؤها من قبل نظام أوراكل و بها بيانات بهدف التدريب. و الشكل (3-1) يوضح محتوى الجدول EMP و الذي يمثل بيانات الموظفين، و الجدول DEPT و الذي يمثل الإدارات التي يعمل بها الموظفون. و لعرض هيكل تلك الجداول تم استخدام الأمر DESCRIBE و تم اختصاره في DESC.



الشكل (3-1) الجداول EMP و DEPT

و أريد أن أنوه إلى أن المفتاح الأساسي Primary Key في جدول EMP هو الحقل empno و المفتاح الأساسي في جدول DEPT هو الحقل deptno. و توجد علاقة بين الجدولين حيث أن الحقل deptno في جدول EMP يعتبر مفتاحاً أجنبياً Foreign Key من خلاله يمكن الرجوع لـ deptno في جدول Dept.

1.2 جُملة SELECT

تستخدم هذه الجملة لاسترجاع البيانات المخزنة في قاعدة البيانات. و كنتيجة لتنفيذ الجملة نحصل على مجموعة من السجلات تمثل الحل، و يمكن أن يكون ناتج التفيذ خالياً من أي سجل. و من خلال جملة Select نستطيع أن ننفذ عدداً من العمليات و هي كما يلي:

أ- الإسقاط Projection

هذه العملية تمكننا من عرض حقول معينة في الجدول وتمكننا كذلك من إمكانية إسقاط الحقول غير مطلوبة.

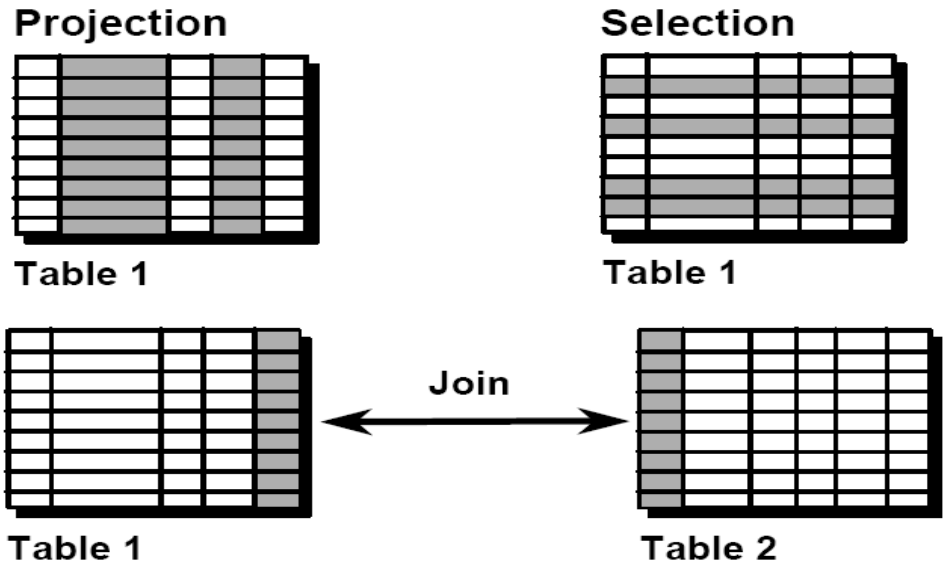
ب- الاختيار Selection

هذه العملية تمكننا من عمل فرز لبعض السجلات من الجدول، والتي تحقق شرطاً معيناً يتم تحديده في الجملة.

ت- الربط Joining

تتيح هذه العملية استرجاع البيانات من عدة جداول.

و الشكل (4-1) يوضح تلك العمليات.



شكل (4-1)

و الصيغة العامة لكتابة جملة Select هي:

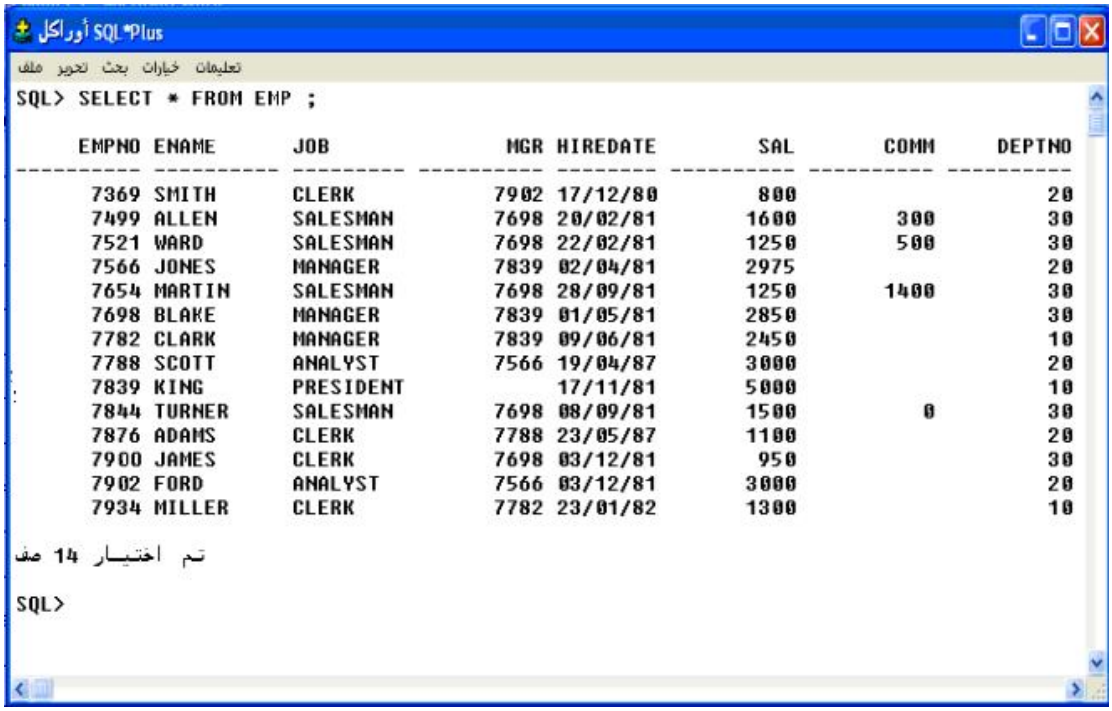
```
SELECT    *|{[DISTINCT] column|expression [alias],...}
FROM      table;
```

حيث إن: Column تمثل الحقل المراد استرجاعه. و يمكن إسترجاع أكثر من حقل أو استخدام الرمز * لتمثيل جميع الحقول. و Table يمثل جدول البيانات المراد استرجعها. وعبارة Distinct تعني عدم إظهار السجلات المتطابقة كلياً و alias لتغير اسم الحقل في عرض النتائج من الاستعلام. و سوف نستعرض في ما يلي عدداً من الأمثلة التي تعكس الحالات المختلفة لاستخدام جملة Select.

مثال (1):

اكتبجملة SQL و التي تستعرض بيانات كل الموظفين.

الحل:



من الملاحظ أننا استخدمنا الرمز * لتمثيل كل حقول الجدول، و يمكن أيضا أن نكتب كل الحقول بدلاً عن النجمة والنتيجة تكون حسب ترتيب ظهور الحقول في الجملة.

أسئلة تقويم ذاتي

1/ وضح أقسام جمل لغة SQL ومهام كل قسم.

2/ بين العمليات التي يمكن تنفيذها من خلال جملة Select .

3/ اكتب الصيغ العامة لكتابة جملة Select .

تدريب (1)

اكتب جملة استعلام تقوم باسترجاع أسماء الموظفين و مرتباتهم و أرقام الإدارات التي يعملون بها .

❐ العمليات الحسابية Arithmetic Operations

يمكن في جمل الاستعلام استخدام عمليات حسابية على الحقول الرقمية و العمليات التي يمكن استخدامها هي :

1. + لعملية الجمع.
2. – لعملية الطرح.
3. * لعملية الضرب.
4. / لعملية القسمة.

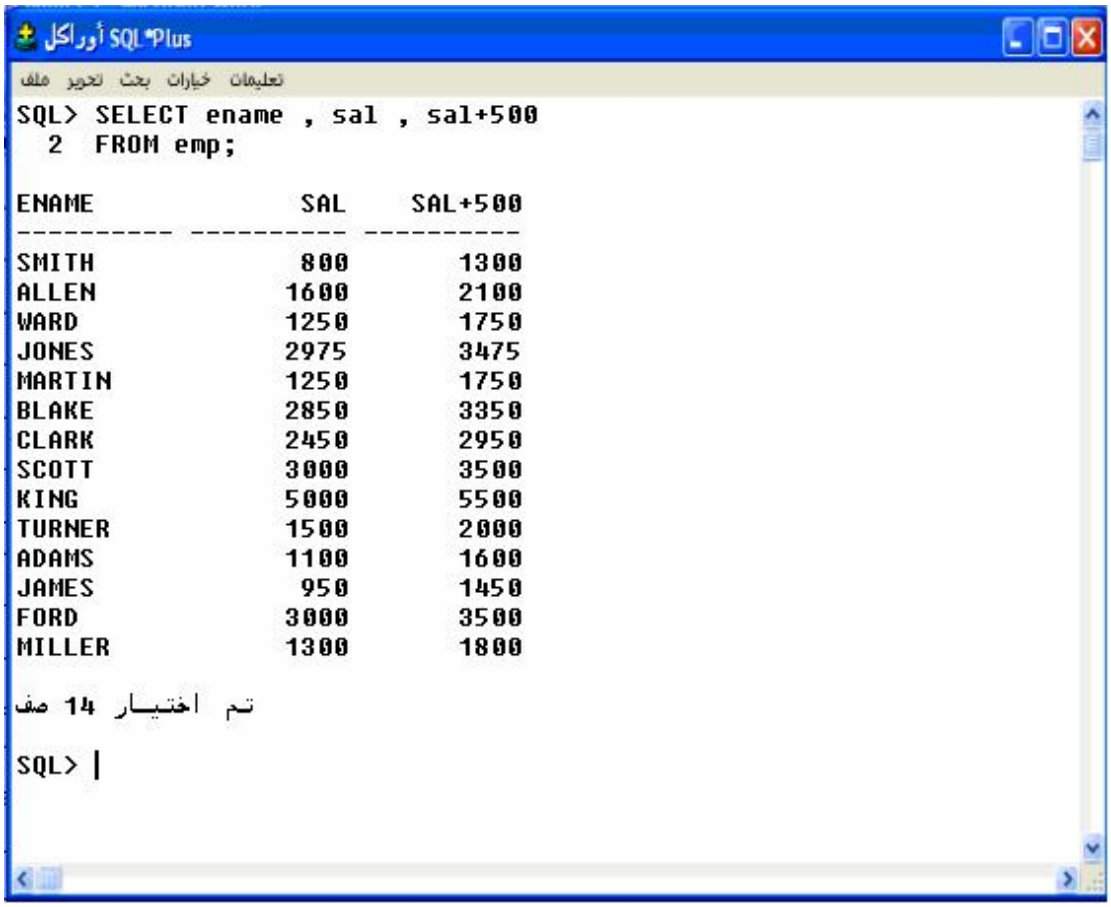
ملاحظة

إن إجراء هذه العمليات مع جملة Select لا تؤثر في أصل البيانات المخزنة في قاعدة البيانات.

مثال (2)

اكتب استعلامايقوم بعرض أسماء الموظفين و مرتباتهم و إضافة 500 جنيه إلى مرتباتهم:

الحل:

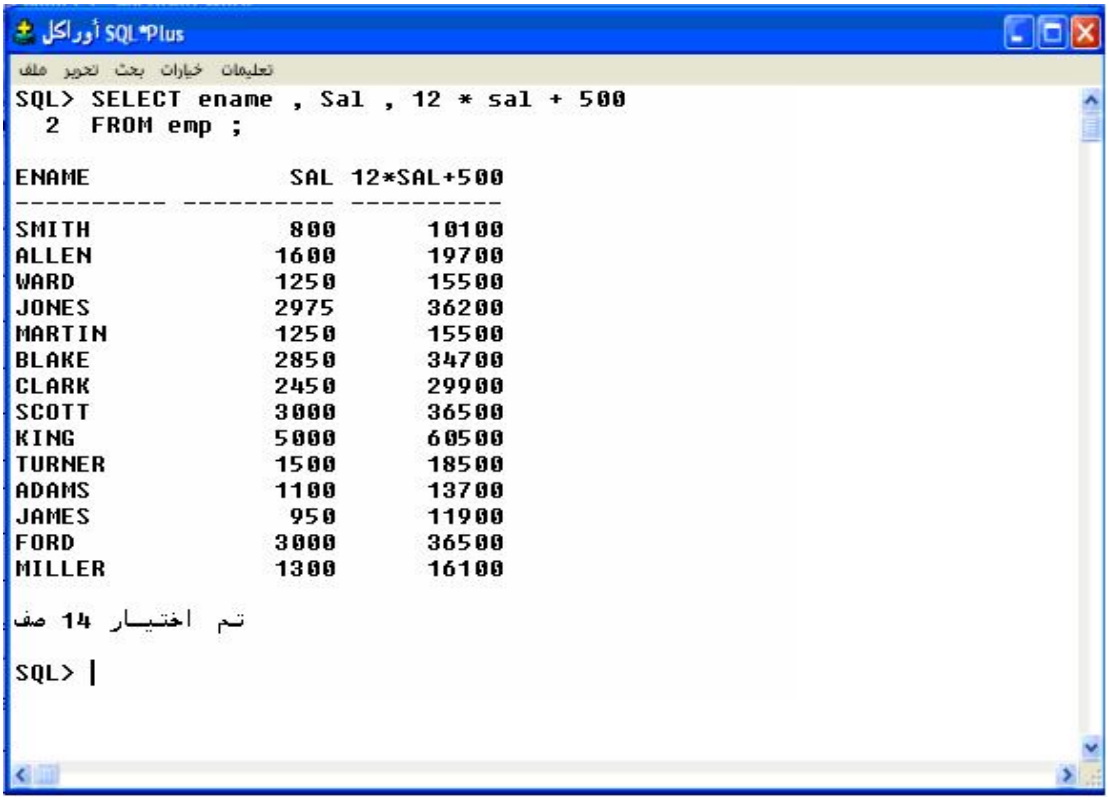


عند تنفيذ هذه العمليات نجد أن الضرب و القسمة لها أولوية من الجمع و الطرح، بمعنى أن الضرب و القسمة تنفذ أولاً، ثم الجمع و الطرح.

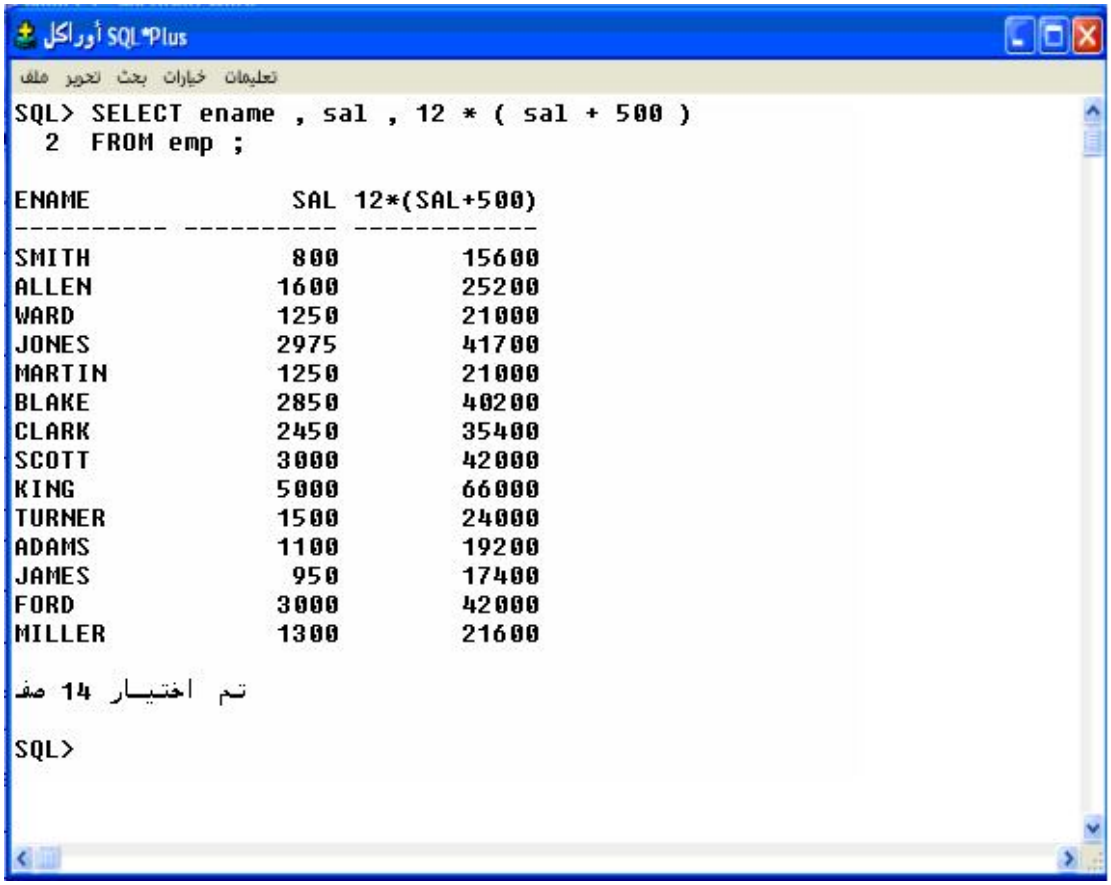
مثال (3)

اكتب استعلامايقوم بعرض أسماء الموظفين و مرتباتهم و مرتباتهم السنوية بعد إضافة مبلغ 500 جنية.

الحل:



نلاحظ أن هذا الحل غير صحيح، و ذلك لأن عملية الضرب تمت قبل الجمع و لايجاد الحل الصحيح نستخدم الأقواس لتغير أولوية تنفيذ العملية ليصبح الجمع قبل الضرب كما يلي:



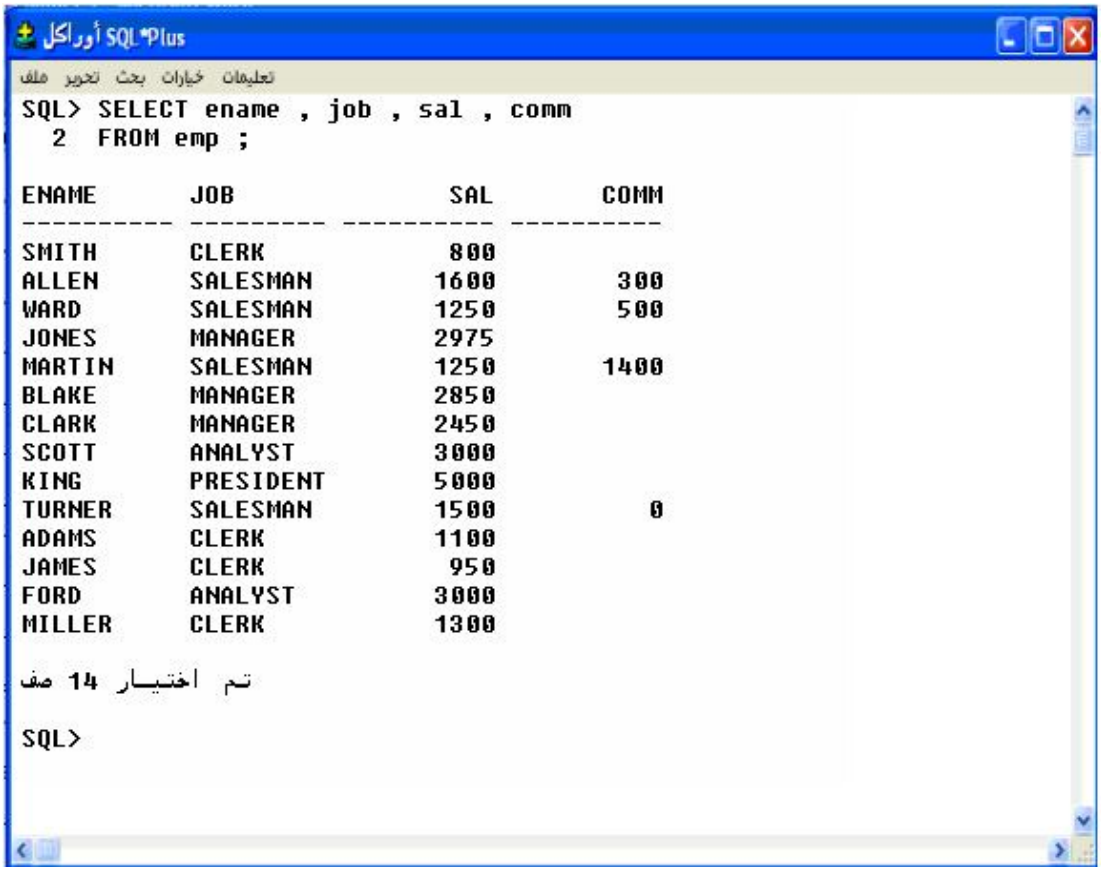
❐ القيمة غير المعرفة NULL Value

القيمة Null تعني أن هذه القيمة غير معرفة أو غير متوفرة، و هي في أي حال من الأحوال لا تعني العدد صفر في الحقول الرقمية و فراغ space في الحقول الحرفية. يمكن تعريف أي حقل على أساس أنه null value ما لم يكن هنالك قيد سلامة Integrity Constraint معرف على هذا الحقل سواء اكان NOT NULL أو أن الحقل معرف كمفتاح أساسي Primary Key.

مثال (4):

المثال التالي يظهر الحقل COMM الذي يحتوي على Null في بعض السجلات الناتجة من الاستعلام.

الحل:

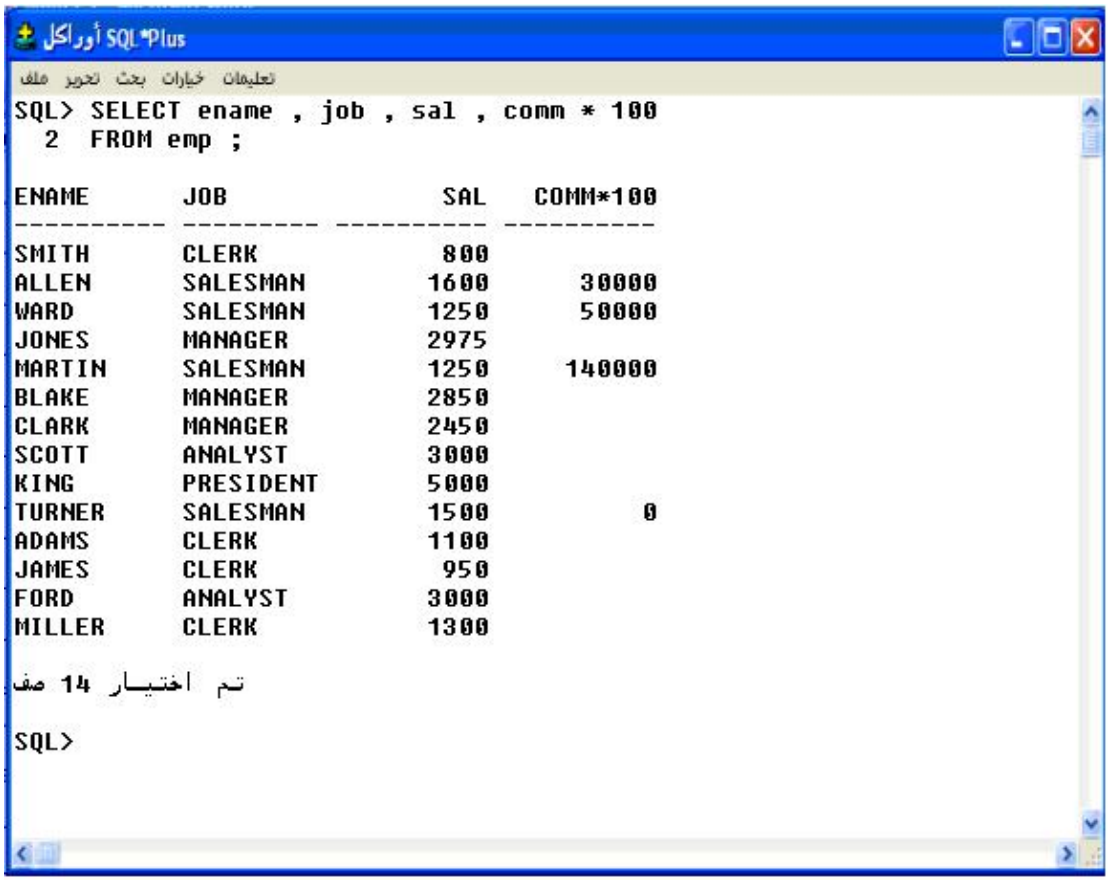


❐ استخدام Null مع العمليات الحسابية

عند إجراء أي عملية حسابية على حقل يحتوي على Null فإن ناتج التنفيذ سوف يكون أيضا Null، و المثال التالي يوضح ذلك.

مثال (5):

المثال التالي يوضح استعلاماً تتم فيه إجراء عملية حسابية على الحقل COMM.



كما يظهر في الحل فإن ناتج التنفيذ ظل أيضا Null بالنسبة للحقول التي كانت تحتوي على Null قبل إجراء العملية الحسابية.

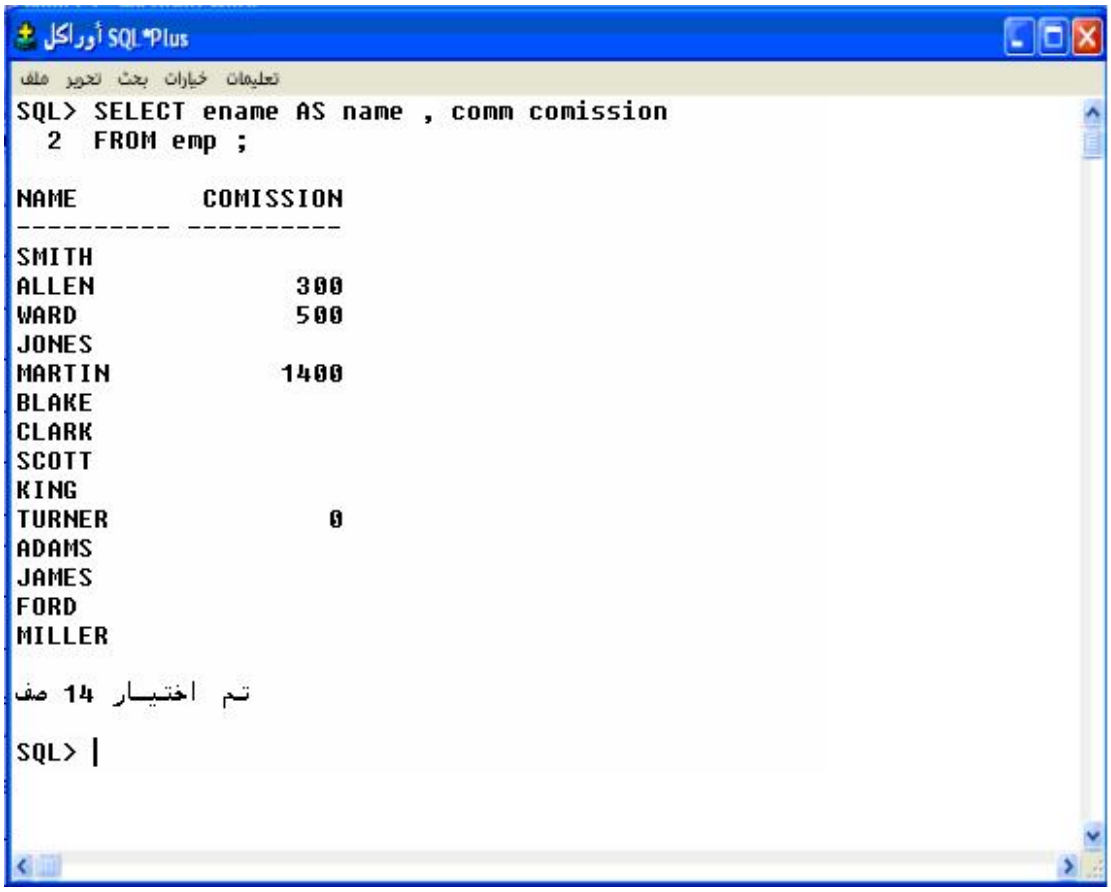
❐ تغيِّر اسم الحقول Alias

من خلال جملة الاستعلام يمكن أن نغير اسم الحقل في عرض ناتج الاستعلام مع العلم أن اسم الحقل يظل كما هو في قاعدة البيانات. و المثال التالي يوضح ذلك.

مثال (6):

اكتبجملة استعلام تقوم بعرض حقول اسم الموظف تحت اسم NAME و الحافز تحت اسم COMMISSION:

الحل



ملاحظة

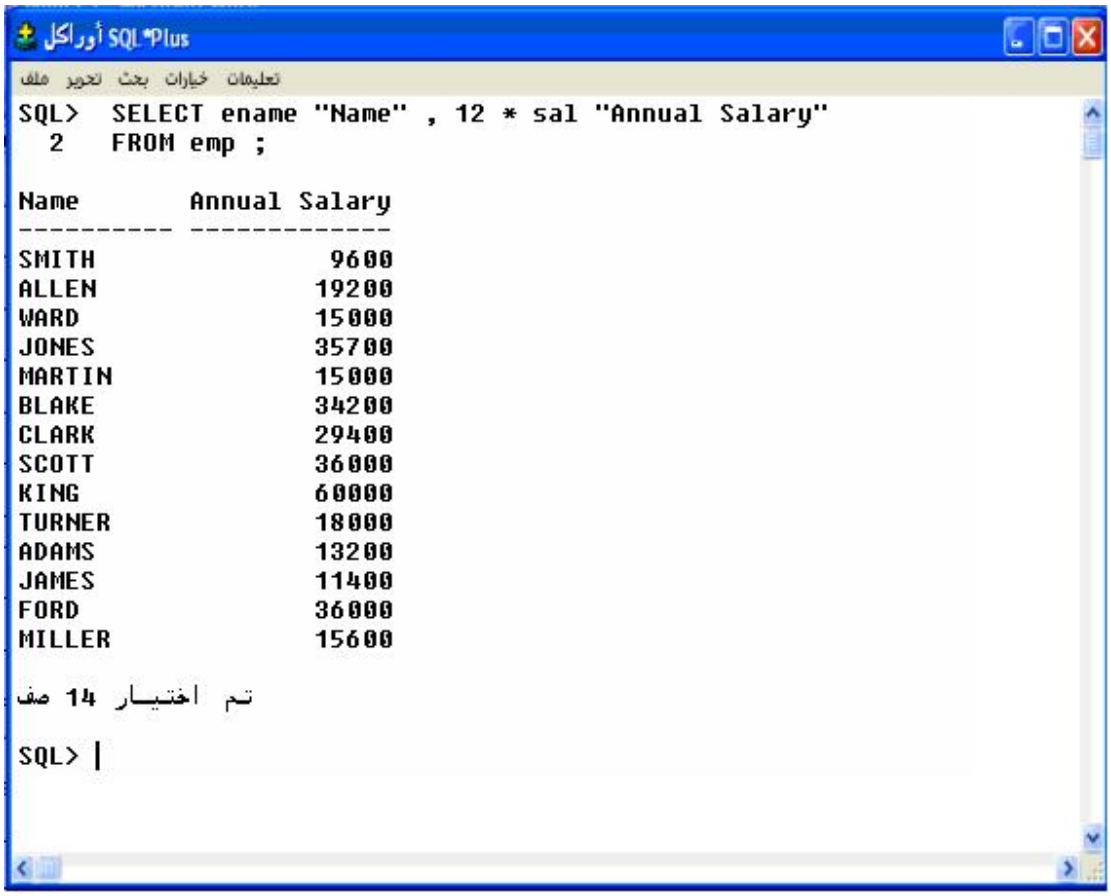
نلاحظ في الشاشة السابقة أننا نكتب اسم الحقل كما هو ثم كلمة AS أو يمكن حذفها ثم نكتب الاسم الجديد للحقل.

يمكن أيضا أن نعرض ناتج تعبير حسابي تحت اسم مكون من أكثر من كلمة واحدة أو كلمة واحدة تظهرحسب حالة كتابتها شريطة أن تكون محصورة بين علامتي تنصيص.

مثال (7):

اكتب جملة استعلام تظهر اسم الموظف تحت اسم Name و المرتب السنوي للموظف تحت اسم Annual Salary :

الحل



أسئلة تقويم ذاتي

1/ يمكن في جمل الاستعلام استخدام عمليات حسابية على الحقول الرقمية. اذكر هذه العمليات التي يمكن استخدامها وبين أولويات تنفيذها .

2/ أشرح بالمثال استخدام **Null** مع العمليات الحسابية.

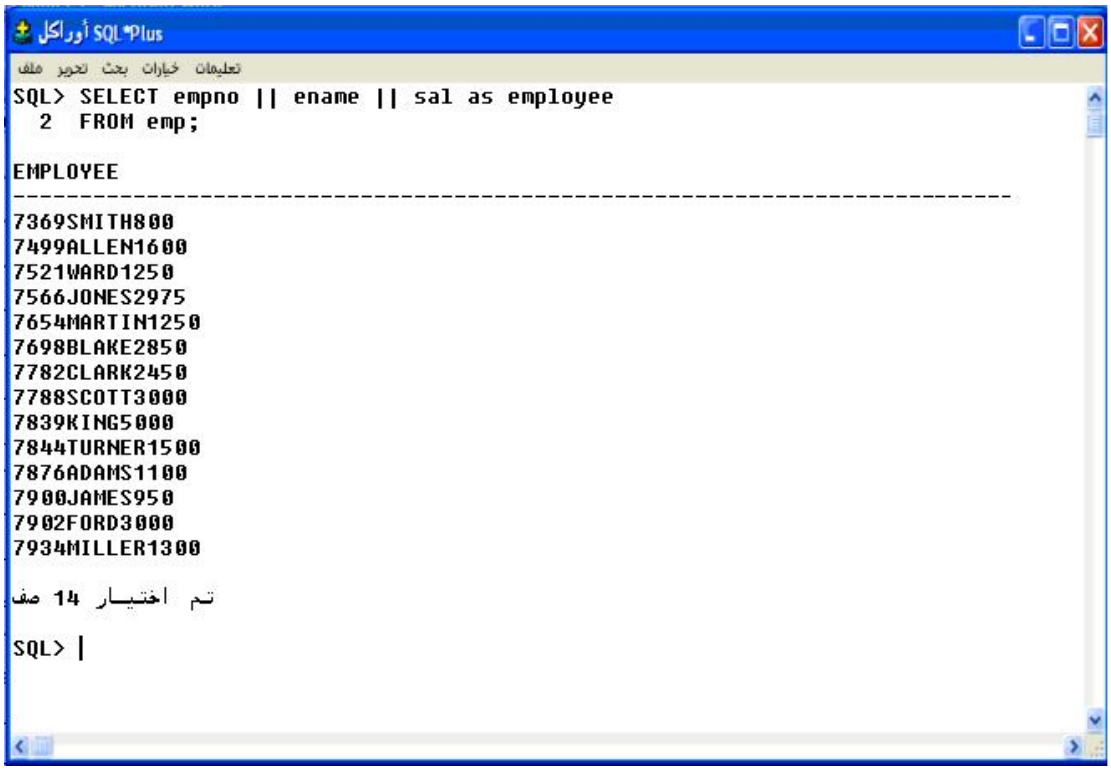
❐ العملية الإلحاقية Concatenation

في جملة الاستعلام يمكن أن نعرض مجموعة من بيانات الحقول كنص واحد و يمكن أن يتخلل ذلك النص نصوص ثابته و نستخدم لتمثيل العملية الإلحاقية الرمز ||

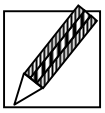
مثال (8)

أكتب استعلام يحتوي علي رقم الموظف و اسم الموظف و المرتب كنص واحد تحت اسم Employee:

الحل:



تدريب (2)



أكتب استعلام يقوم بعرض اسم الموظف ثم النص is a ثم اسم الوظيفة على أن يكون ناتج الحل تحت اسم Employee Details.

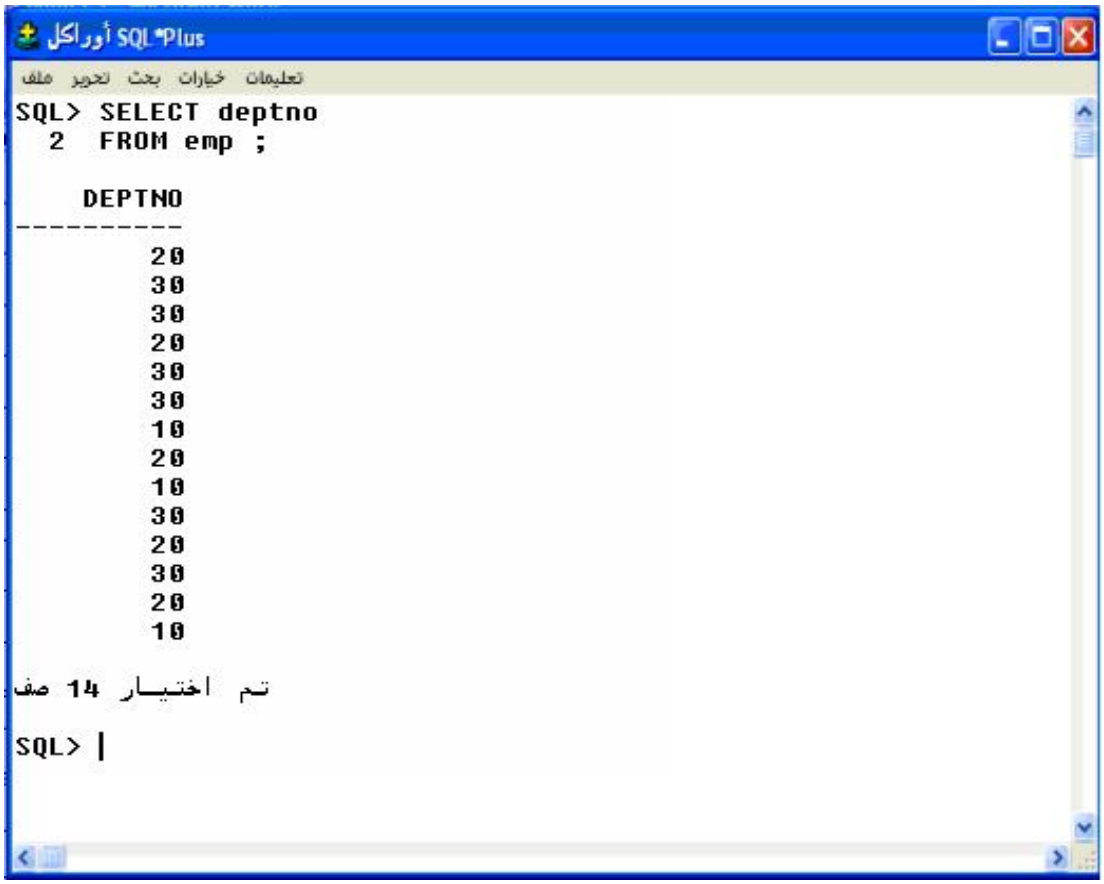
❐ العبارة DISTINCT

تستخدم هذه العبارة بصورة اختيارية لمنع تكرار ظهور السجل بصورة متكررة كلياً و لاتمنع تكرار الحقول جزئياً.

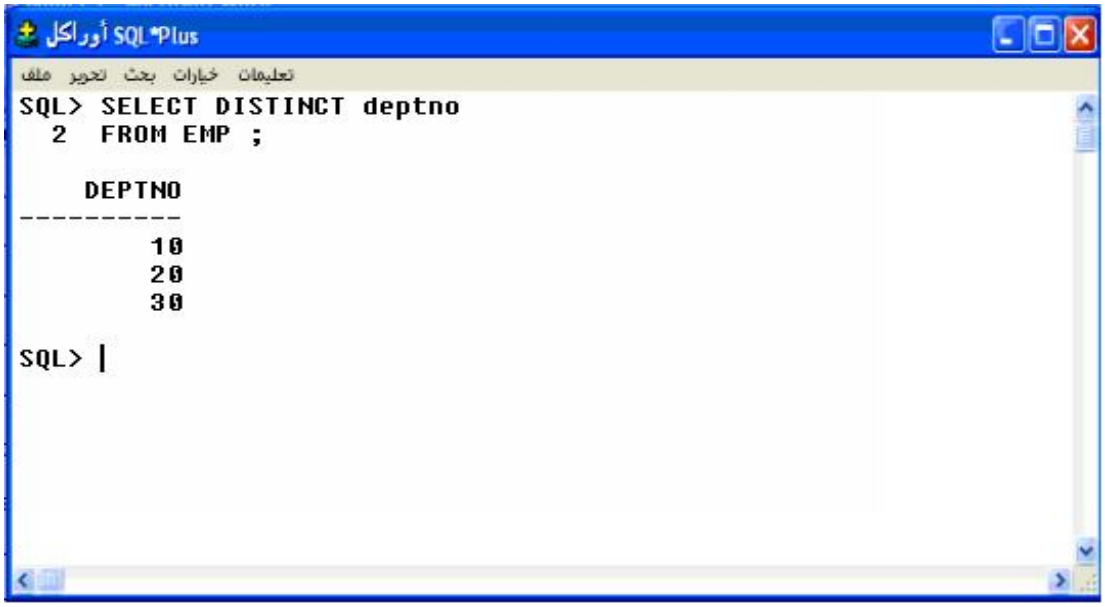
مثال(9)

أكتب استعلام يقوم بعرض ارقام الموظفين من جدول الموظفين ثم قم بكتابة الاستعلام مره اخري مع منع ظهور تكرار أرقام الإدارات.

الحل:

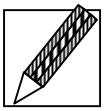


في هذه الحالة ظهر لنا 14 سجل مع ملاحظة وجود تكرار.



في هذه الحالة ظهر لنا 3 سجلات و ذلك لأننا ازلنا التكرار باستخدام عبارة Distinct و هذه الطريقة نستطيع أن نتعرف على أرقام الإدارات التي يعمل بها الموظفين من جدول الموظفين.

تدريب (3)



أكتب استعلاماً يقوم بعرض أسماء الوظائف و أرقام الإدارات من جدول الموظفين مره بدون استخدام Distinct و مرة أخرى باستخدامها.

1.1.2 فرز و ترتيب البيانات Restricting and Sorting Data

عزيزي الدارس، أثناء استرجاع البيانات من قاعدة البيانات قد نحتاج أن نقيد تلك البيانات بشروط محددة، و أيضا قد نحتاج إلى عرضها بترتيب معين. و حتى نتمكن من عمل ذلك نضيف بعض العبارات إلى جملة الاستعلام لكي يتحقق ذلك.

و كما ذكرنا في الفصل السابق فإن عملية الاختيار Selection تتيح لنا استرجاع السجلات من الجدول حسب شرط معين.

❐ عبارة Where

تستخدم هذه العبارة مع جملة الاستعلام لإضافة شرط يجب أن يتحقق مقابل السجلات حتى تكون مؤهلة أن تكون في مجموعة حل الاستعلام. و بعد إدخال عبارة Where تكون صيغة الاستعلام كما يلي:

```
SELECT   *|{[DISTINCT] column|expression [alias],...}
FROM     table
[WHERE   condition(s)];
```

حيث إن Condition(s) تمثل الشرط أو قائمة الشروط التي يجب أن تتحقق من قبل السجلات.

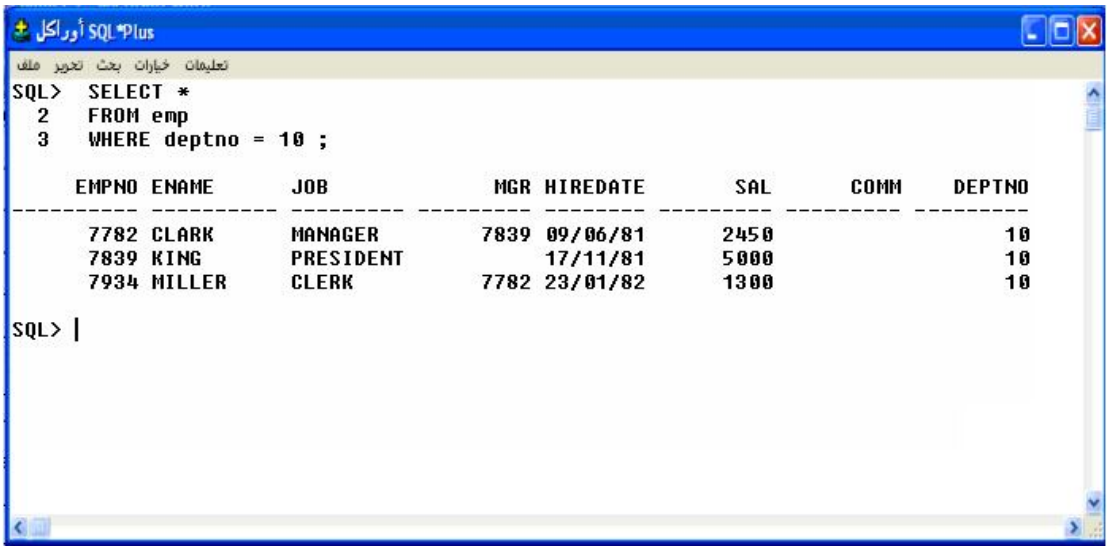
<u>و يمكن أن نستخدم العمليات التالية في تكوين الشرط</u>

- = و تمثل التساوي.
- < و تمثل أكبر من.
- =< و تمثل أكبر من أو يساوي
- > و تمثل أصغر من
- => و تمثل أصغر من أو يساوي
- <> و تمثل لا يساوي.

مثال(10)

اكتبجملة الاستعلام لإسترجاع بيانات الموظفين الذين يعملون في الإدارة رقم 10:

الحل:

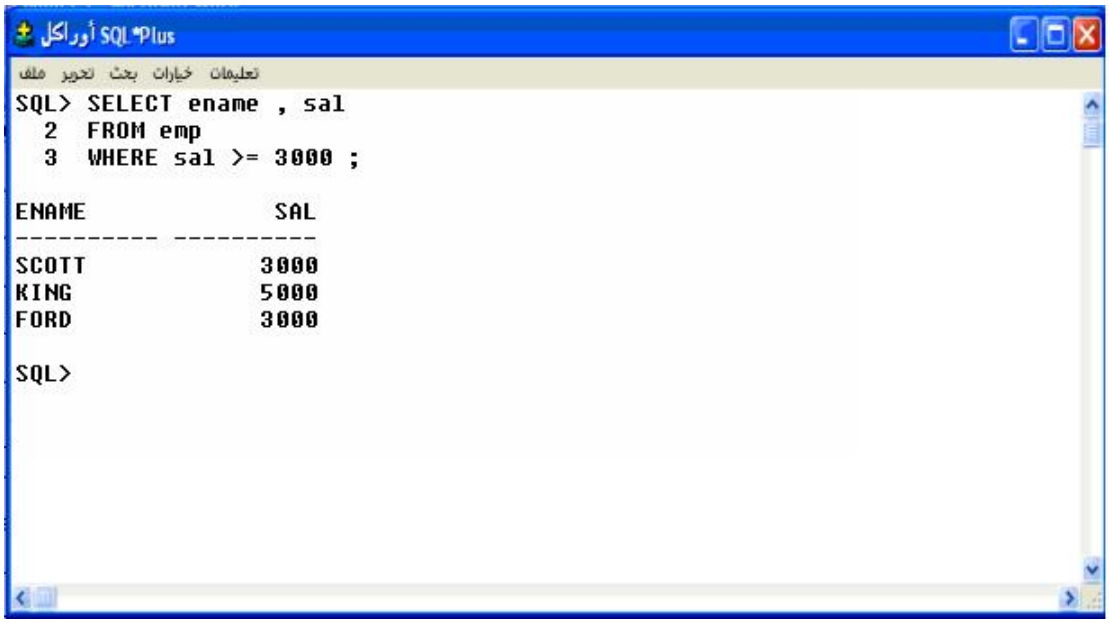


نلاحظ أن مجموعة الحل شملت كل السجلات التي حققت الشرط deptno = 10 .

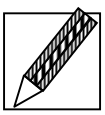
مثال(11)

اكتبجملة الاستعلام لاسترجاع اسماء و مرتبات الموظفين بحيث أن مرتبـــاتهم أكبـــر مـــن أو يساوي 3000جنيه:

الحل:



تدريب (4)



اكتب جملة استعلام لاسترجاع أسماء و مرتبات و وظائف الموظفين الذين يعملون كـــ Salesman.

أسئلة تقويم ذاتي

1/ تستخدم عبارة Where مع جملة الاستعلام لإضافة ..... يجب أن يتحقق مقابل السجلات حتى تكون مؤهلة أن تكون في مجموعة حل الاستعلام. و بعد إدخال عبارة Where تكون صيغة الاستعلام هي .....

2/ عدد العمليات التي تستخدم في تكوين الشرط.

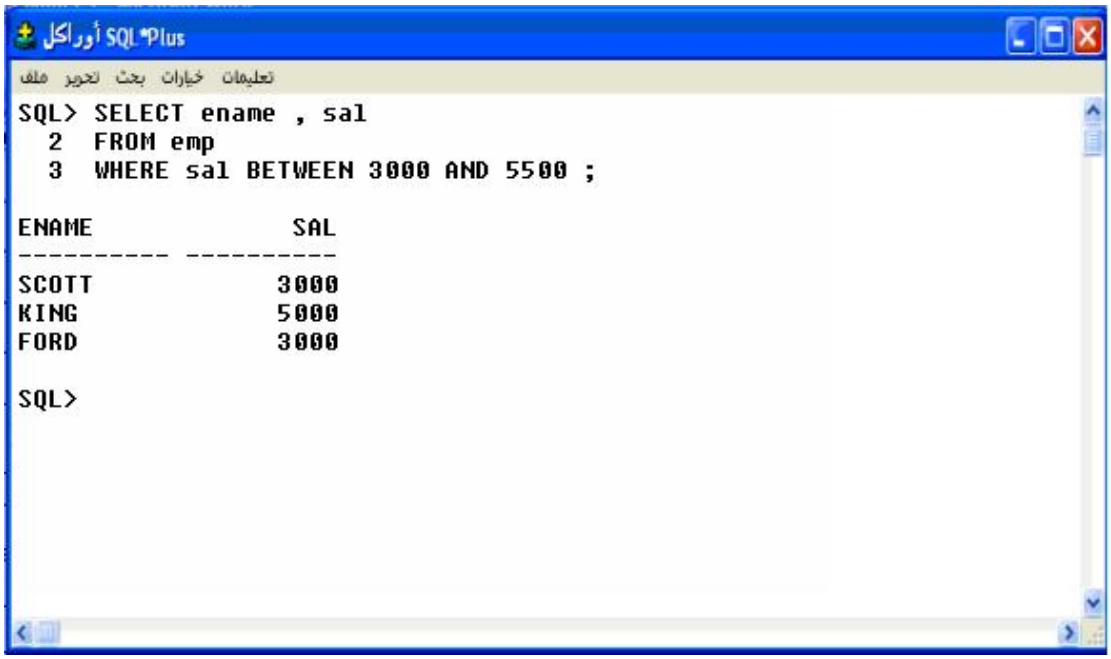
<u>و هنالك أيضا عمليات إضافية حتى تكون الشروط أكثر مرونة و هي</u>

- BETWEEN valu1 AND value2 تمثل الأعداد ابتداءً من value1 إلى value2
- IN(Set) تمثل كل الأعداد التي وردت في المجموعة Set.
- LIKE تستخدم لعمل المقارنة بين الأحرف في الحقول النصية.
- IS NULL ما إذا كان الحقل فارغ أم لا.

مثال(12)

اكتب جملة الاستعلام لاسترجاع اسماء و مرتبات الموظفين بحيث تكون مرتباتهم من 3000 إلى 5500 جنيه:

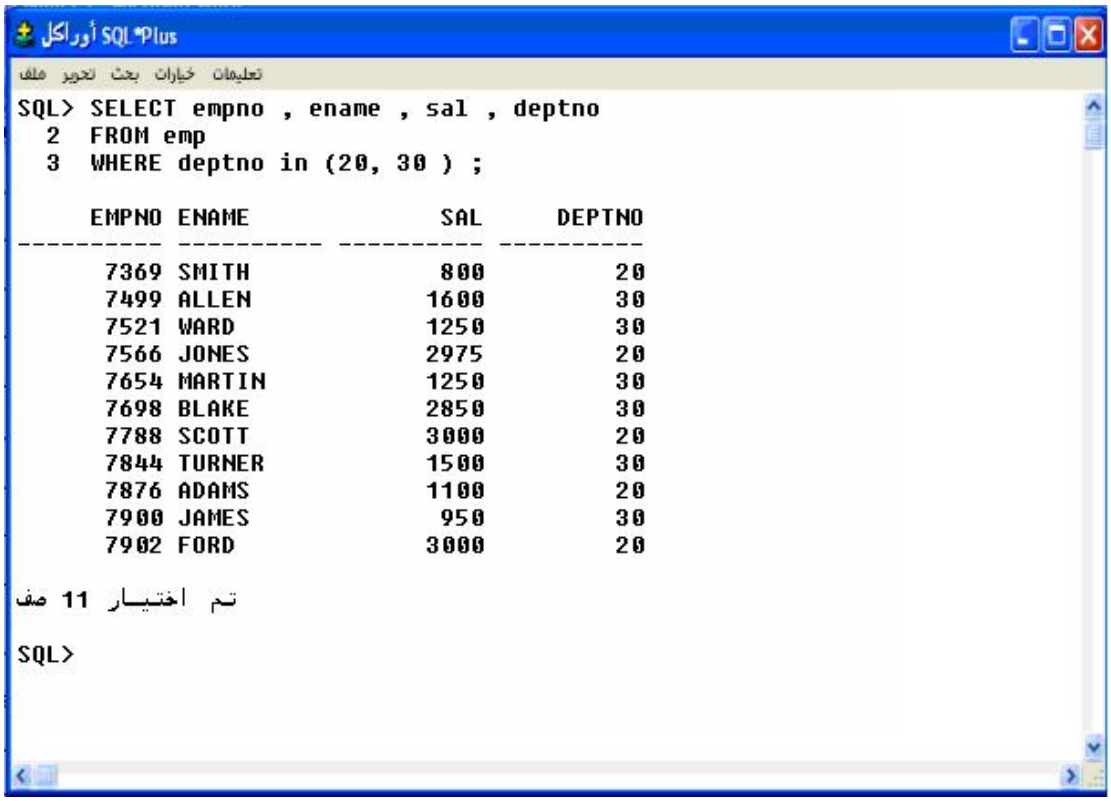
الحل:



مثال (13)

اكتبجملة استعلام لعرض ارقام و اسماء و مرتبات و ارقام إدارات الموظفين والذين يعملــون في الإدارات رقم 20 و 30 :

الحل:



كما ذكرنا سابقاً فإن عبارة LIKE تستخدم لعمل مقارنات بين القيم و الحقول الحرفية. وعند استخدام العلامة : % فذلك يعني أي متسلسلة من الأحرف أو مجموعة خالية مــن الاحــرف تستخدم لتمثيل حرف واحد.

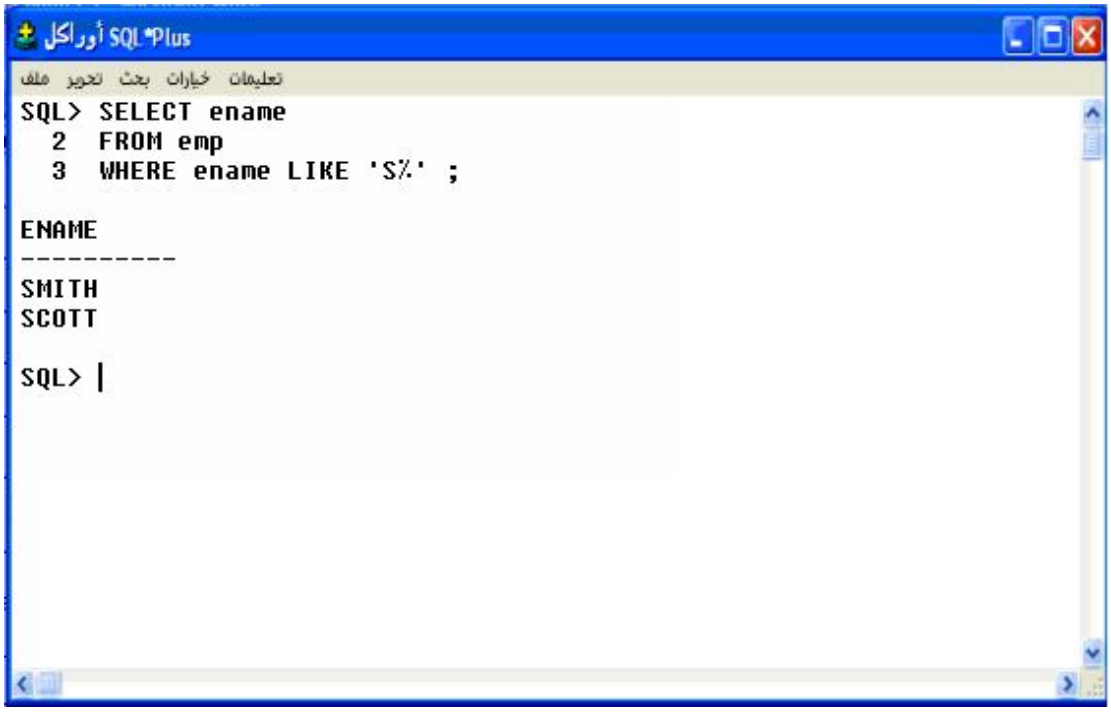
فعلى سبيل المثال:

- '%N' تعني جميع النص الذي ينتهي بحرف N
- '_H__' تعني جميع النصوص المكونة من أربعة أحرف و الحرف الثاني H

مثال(14)

اكتبجملة استعلام تسترجع أسماء الموظفين الذين تبدأ أسماؤهم بحرف S :

الحل:



تدريب (5)

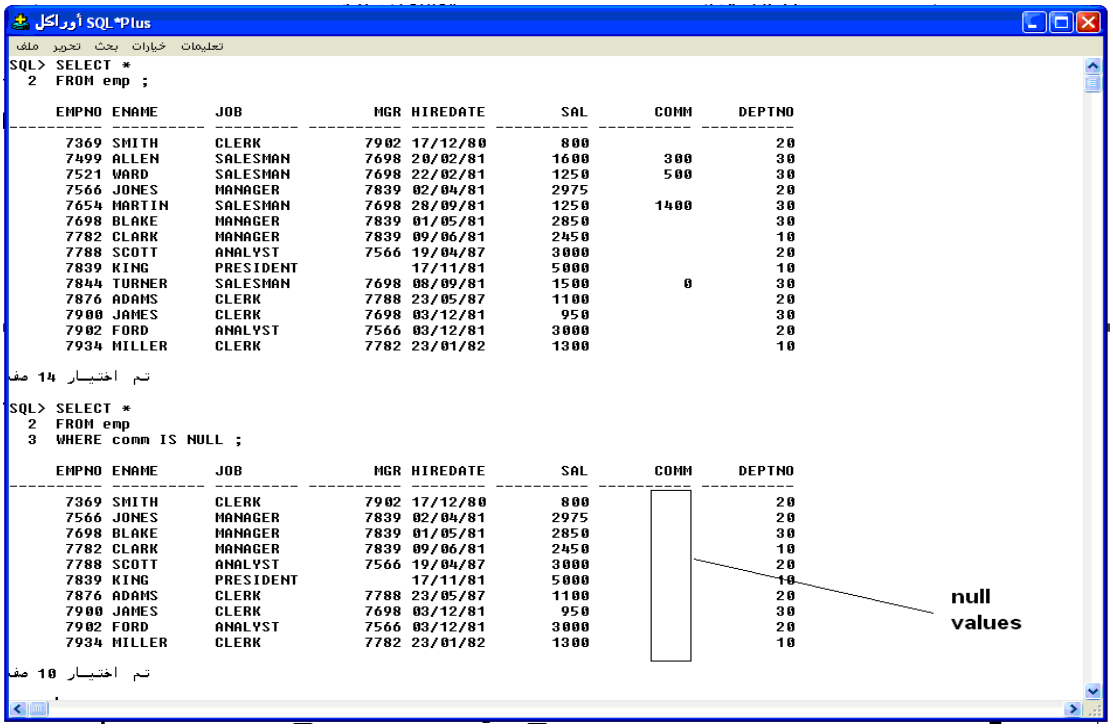


اكتب جملة الاستعلام لعرض أسماء الموظفين بحيث يكون الحرف الثاني هو حرف A ؟.

مثال(15)

اكتب جملة استعلام لعرض بيانات الموظفين بحيث أن الـــ COMM غير معرف:

الحل:



كما نلاحظ أن الاستعلام الاول يستعرض كل السجلات و الاستعلام الثاني يمثل اجابة السؤال.

عزيزي الدارس،

<u>عند ما يكون لدينا أكثر من شرط فلابد من ربط هذه الشروط. والروابط المنطقية هي كما يلي:</u>

- AND و تعني " و ".
- OR و نعني " أو ".
- NOT تستخدم للنفي.

و تخضع هذه العمليات **<u>للقواعد التالية:</u>**

أ- AND :تسترجع True إذا كان الشرطان صحيحين و عدا ذلك تسترجع False.

ب- OR :تسترجع True إذا كان أحد الشرطين صحيحاً على الأقل، و عدا ذلك تسترجع False.

ت- NOT :تسترجع True إذا كان الشرط False .

و في ما يلي الجداول التالية توضح ذلك:

AND	TRUE	FALSE	NULL
TRUE	TRUE	FALSE	NULL
FALSE	FALSE	FALSE	FALSE
NULL	NULL	FALSE	NULL

جدول (1-2) العمليات على AND

OR	TRUE	FALSE	NULL
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	NULL
NULL	TRUE	NULL	NULL

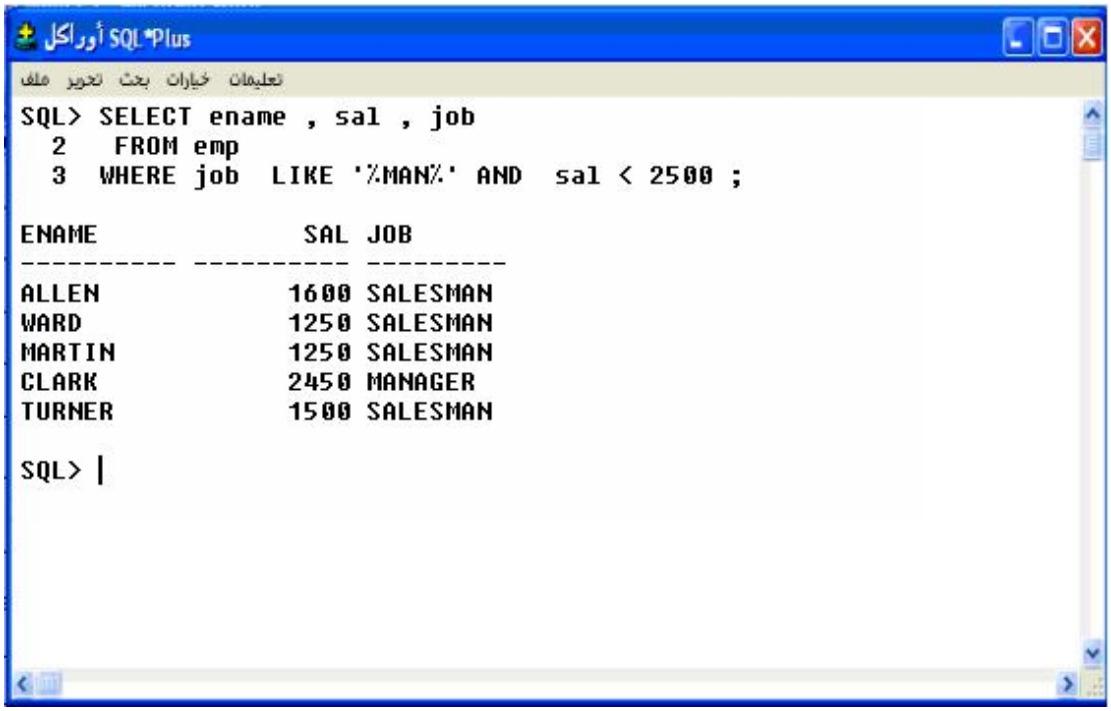
جدول (2-2) العمليات على OR

NOT	TRUE	FALSE	NULL
	FALSE	TRUE	NULL

جدول (2-3) العمليات على NOT

مثال(16)

اكتب جملة الإستعلام تقوم باسترجاع اسماء الموظفين و رواتبهم و وظائفهم بحيث يحتوي اسم الوظيفة على الحروف "MAN" و مرتباتهم تقل عن 2500 جنيه.



و يمكن تغير السؤال السابق بحيث يكون كما يلي.

مثال(17)

اكتبجملة الإستعلام تقوم باسترجاع اسماء الموظفين و رواتبهم و وظائفهم بحيث تحتوي اسم الوظيفة على الحروف "MAN" أو مرتباتهم تقل عن 2500 جنيه.

الحل:

```
SQL> SELECT ename , sal , job
  2   FROM emp
  3  WHERE job  LIKE '%MAN%' OR  sal < 2500 ;

ENAME              SAL JOB
---------- ---------- ---------
SMITH              800 CLERK
ALLEN             1600 SALESMAN
WARD              1250 SALESMAN
JONES             2975 MANAGER
MARTIN            1250 SALESMAN
BLAKE             2850 MANAGER
CLARK             2450 MANAGER
TURNER            1500 SALESMAN
ADAMS             1100 CLERK
JAMES              950 CLERK
MILLER            1300 CLERK

تم اختيار 11 صف

SQL>
```

مثال(18)

اكتب جملة استعلام لعرض بيانات الموظفين بحيث يكون الـــ COMM معرف:

الحل:

```
SQL> SELECT *
  2  FROM emp ;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- -------- ---------- ---------- ----------
      7369 SMITH      CLERK           7902 17/12/80        800                    20
      7499 ALLEN      SALESMAN        7698 20/02/81       1600        300         30
      7521 WARD       SALESMAN        7698 22/02/81       1250        500         30
      7566 JONES      MANAGER         7839 02/04/81       2975                    20
      7654 MARTIN     SALESMAN        7698 28/09/81       1250       1400         30
      7698 BLAKE      MANAGER         7839 01/05/81       2850                    30
      7782 CLARK      MANAGER         7839 09/06/81       2450                    10
      7788 SCOTT      ANALYST         7566 19/04/87       3000                    20
      7839 KING       PRESIDENT            17/11/81       5000                    10
      7844 TURNER     SALESMAN        7698 08/09/81       1500          0         30
      7876 ADAMS      CLERK           7788 23/05/87       1100                    20
      7900 JAMES      CLERK           7698 03/12/81        950                    30
      7902 FORD       ANALYST         7566 03/12/81       3000                    20
      7934 MILLER     CLERK           7782 23/01/82       1300                    10

تم اختيار 14 صف

SQL> SELECT *
  2  FROM emp
  3  WHERE comm is NOT NULL ;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- -------- ---------- ---------- ----------
      7499 ALLEN      SALESMAN        7698 20/02/81       1600        300         30
      7521 WARD       SALESMAN        7698 22/02/81       1250        500         30
      7654 MARTIN     SALESMAN        7698 28/09/81       1250       1400         30
      7844 TURNER     SALESMAN        7698 08/09/81       1500          0         30

SQL>
```



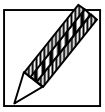
نلاحظ في الاستعلام الأول أنه تم عرض كل السجلات، و أما في الاستعلام الثاني فقد تم عرض السجلات التي تحتوي على قيم في COMM .

أسئلة تقويم ذاتي



1/ اشرح كيف يتم ربط الشروط والروابط المنطقية عندما يكون لدينا أكثر من شرط.

تدريب (6)



اكتب استعلاماً يقوم بعرض أسماء ووظائف و مرتبات الموظفين بشرط أن هولاء الموظفين لا يعملون في الإدارات Clerk , Manager , President

❐ أولويات تنفيذ العمليات

يتم تنفيذ العمليات حسب الجدول التالي:

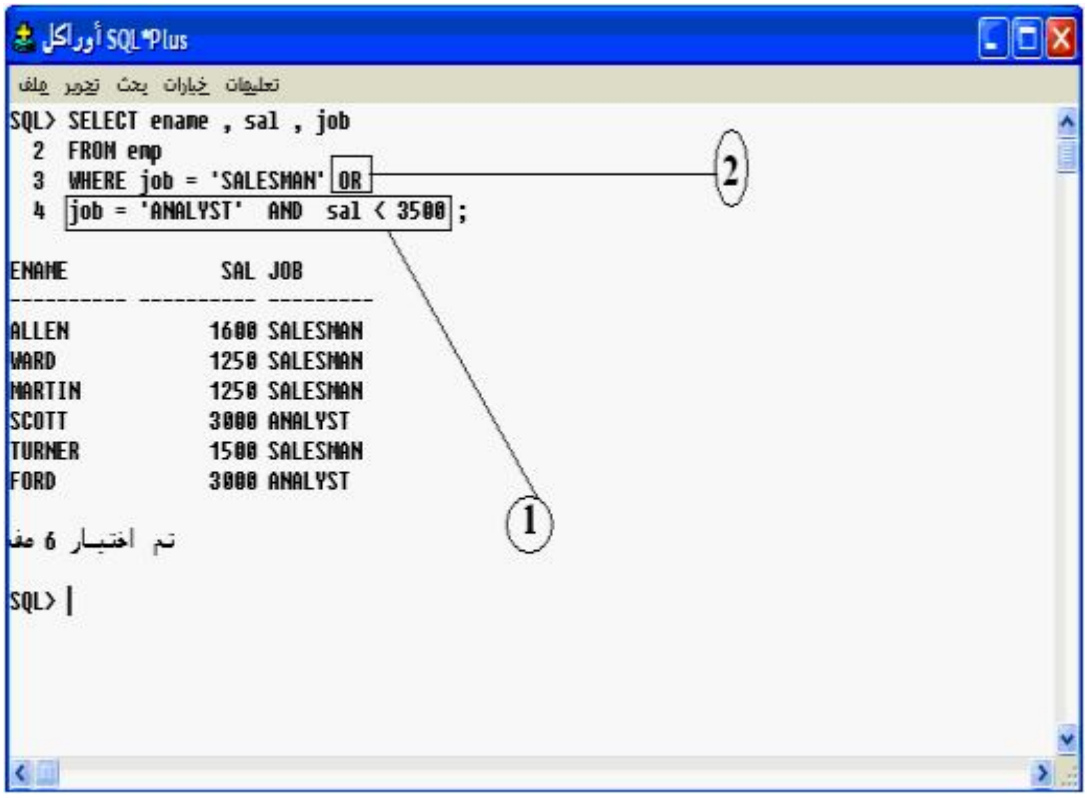
العملية	أولويت ترتيب التنفيذ
العمليات الحسابية Arithmetic Operations	1
العملية الإلحاقية Concatenation Operation	2
عمليات المقارنة Comparison Operations	3
IS [NOT] NULL , LIKE, [NOT]IN	4
[NOT] BETWEEN	5
NOT	6
AND	7
OR	8

مثال(19)

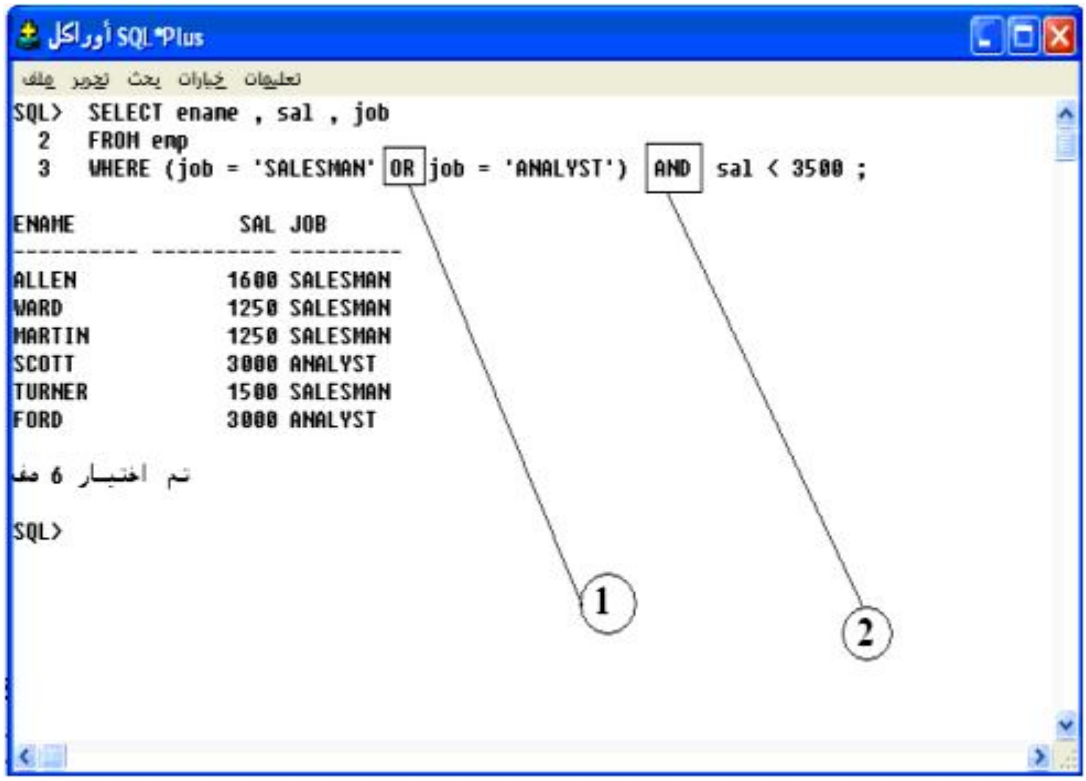
اكتب استعلاماً يقوم بعرض أسماء الموظفين و مرتباتهم بشرط أنهـــم SALESMAN H أو أنهم يعملون ANALYST و مرتباتهم تقل عن 3500

الحل:

نجد أن الشرطين الأخيرين سيتم التحقق منهما أولاً و الناتج سيتم التحقق منـــه مـــع الشـــرط الأول.



و يمكن أن نغير في ترتيب الأولويات باستخدام الأقواس بحيث تنفذ العملية OR قبـــل AND و نكتب الأمر كما يلي:



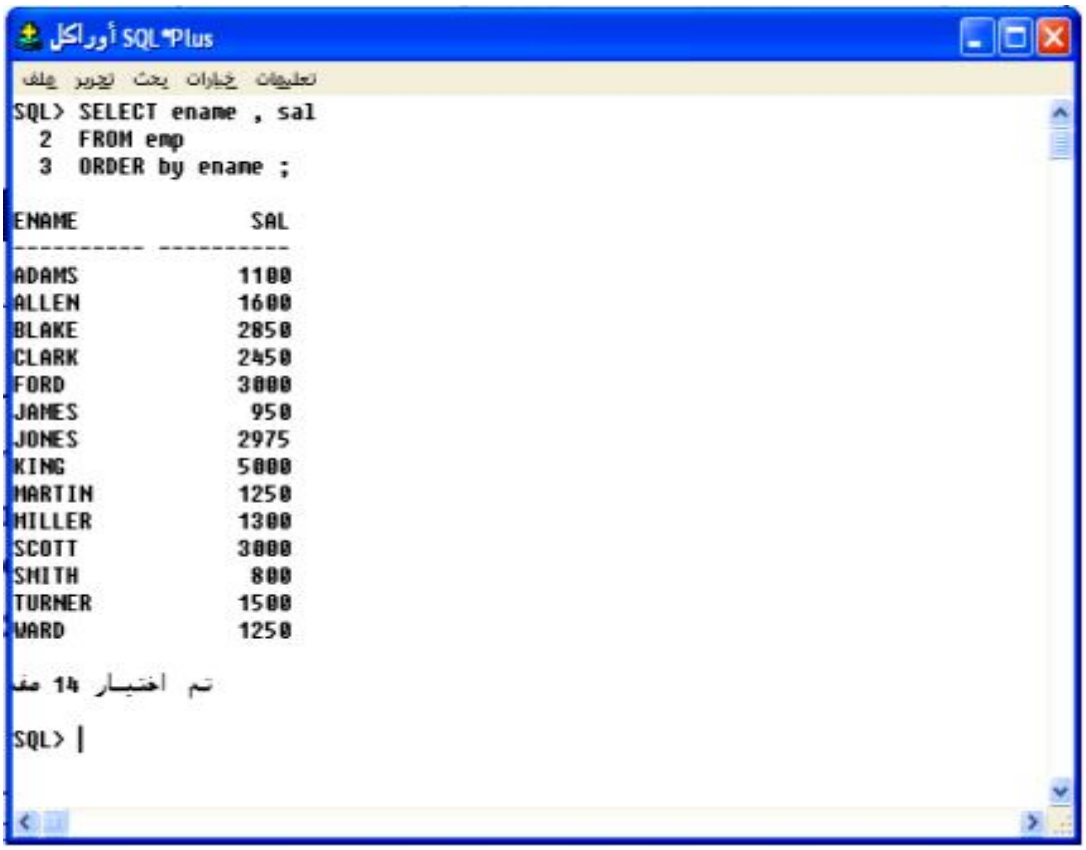
❐ ترتيب البيانات Ordering Data

يمكن أن نرتب البيانات الناتجة من الاستعلام باستخدم حقل و أكثر تصاعديا أو تنازلياً و نستخدم في ذلك عبارة Order by و للترتيب التصاعدي نستخدم الكلمة ASC و للترتيب التنازلي نستخدم DESC. و في حالة عدم استخدام أي من الكلمتين فإن الوضع الافتراضي هو ASC .

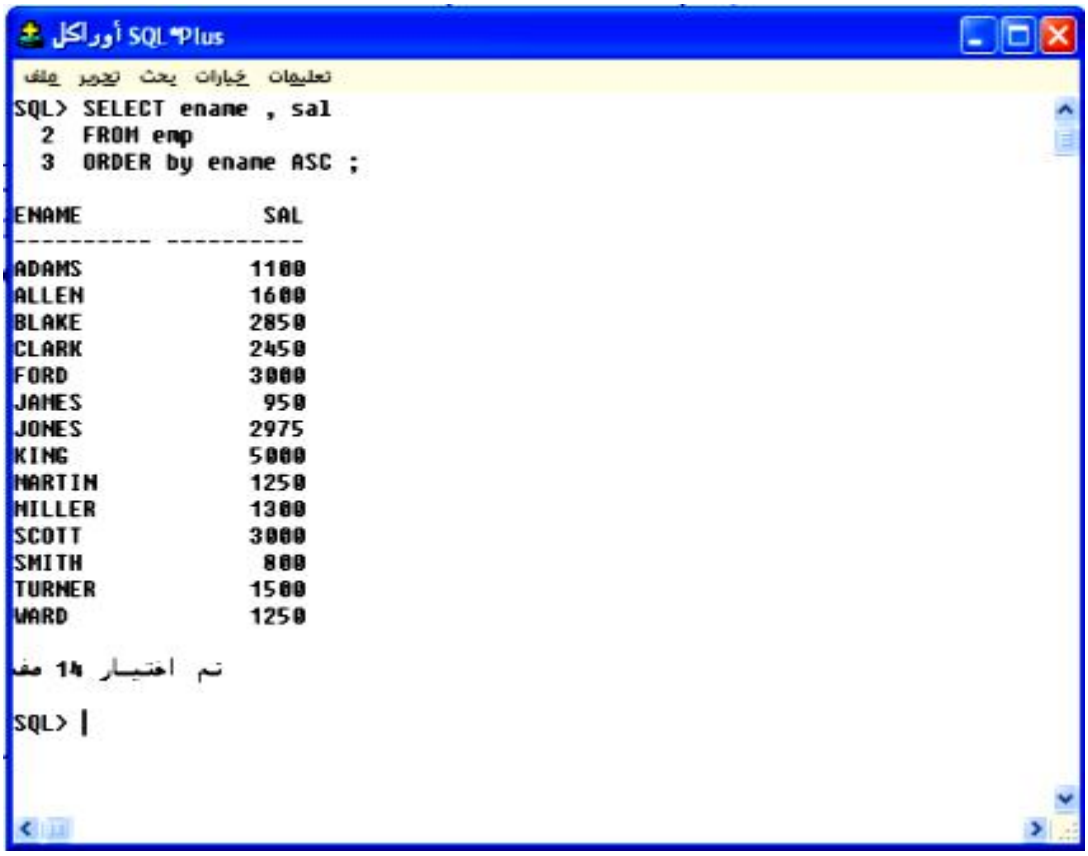
مثال(20)

قم باسترجاع أسماء الموظفين و مرتباتهم ترتيبا تصاعديا حسب الأسماء:

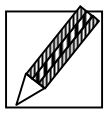
الحل:



أو يمكن أن نكتب ASC كما يلي:



تدريب (7)



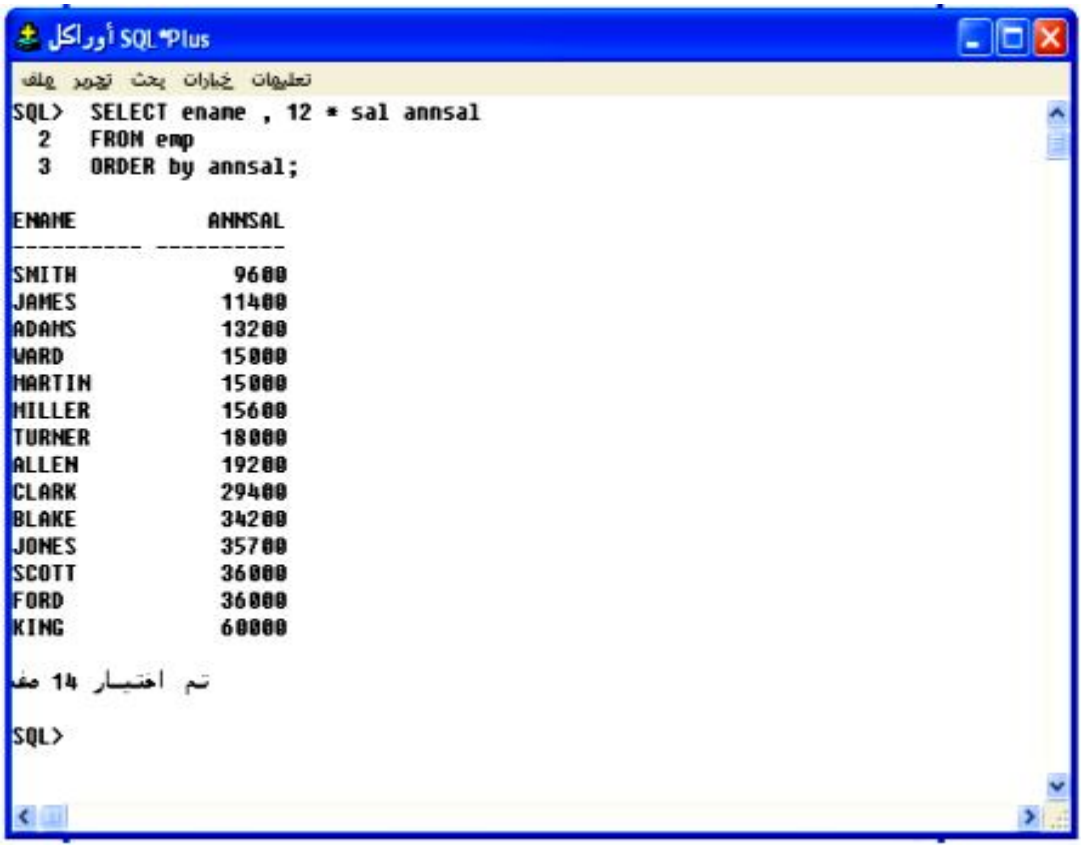
أعد كتابة المثال السابق بحيث تعرض البيانات ترتيباً تنازليا حسب الأسماء .

مثال(21)

اكتب استعلاماً يقوم بعرض اسماء الموظفين و مرتباتهم السنوية ترتيباً تصاعديا حسب مرتباتهم السنوية:

الحل:

نستخدم في هذا المثال الاسم المستعار للمرتبات السنوية Alias



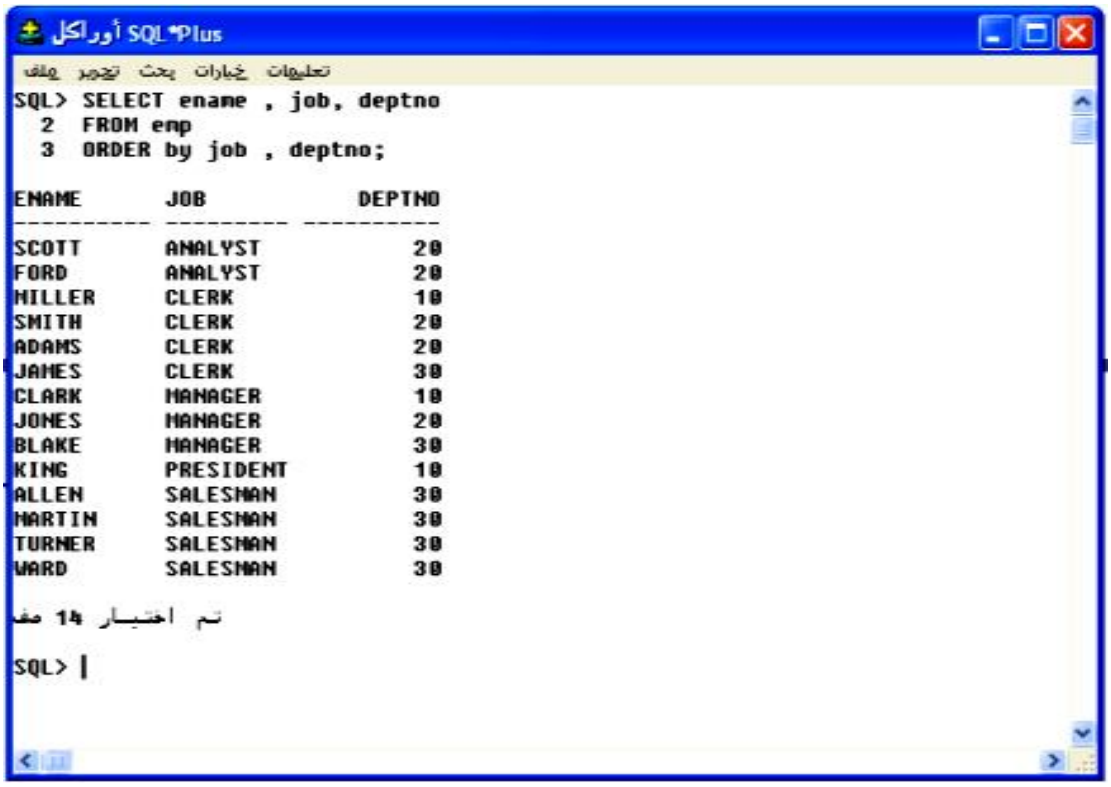
❐ ترتيب البيانات بأكثر من حقل

عزيزي الدارس، يمكن أن نرتب البيانات باستخدام أكثر من حقل على أن تكون أولوية الترتيب للحقل الذي ورد أولاً و حصل تطابق يتم الترتيب باستخدام الحقل الثاني و هكذا....

مثال(22)

اكتب استعلاماً يقوم بعرض اسماء و وظائف و أرقام إدارات الموظفين ترتيباً تصاعديا حسب الوظائف ثم أرقام الادارات:

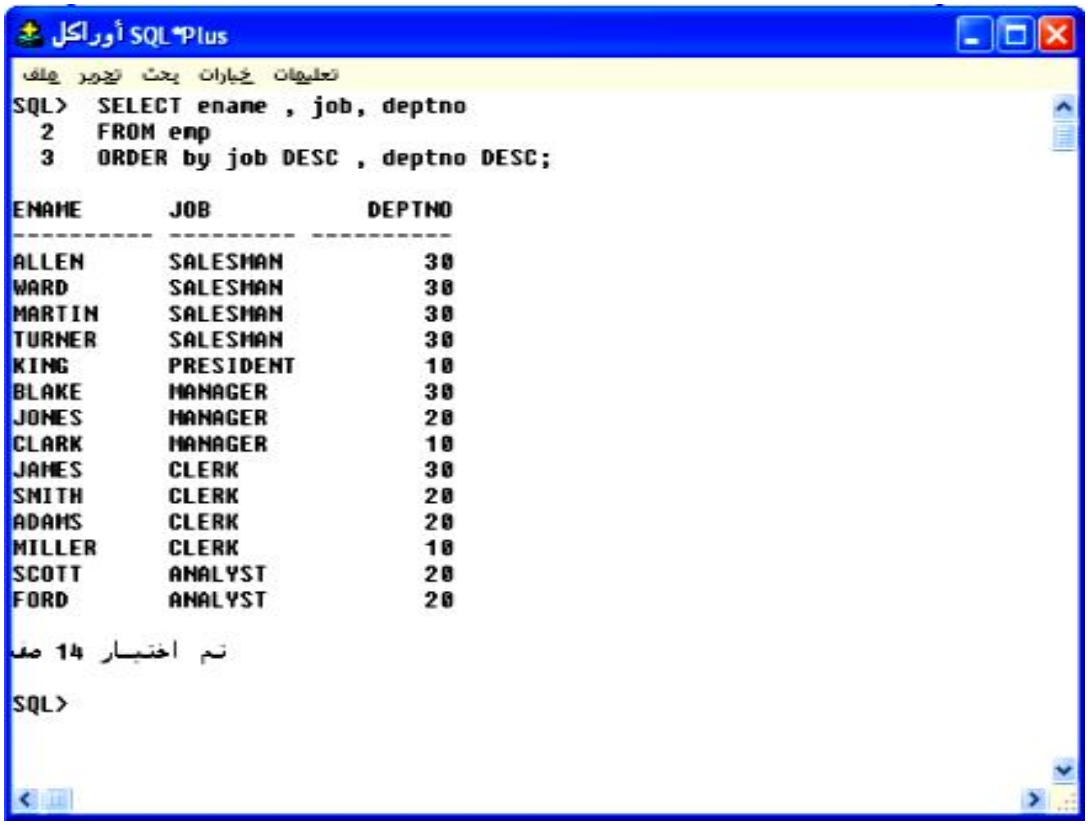
الحل:



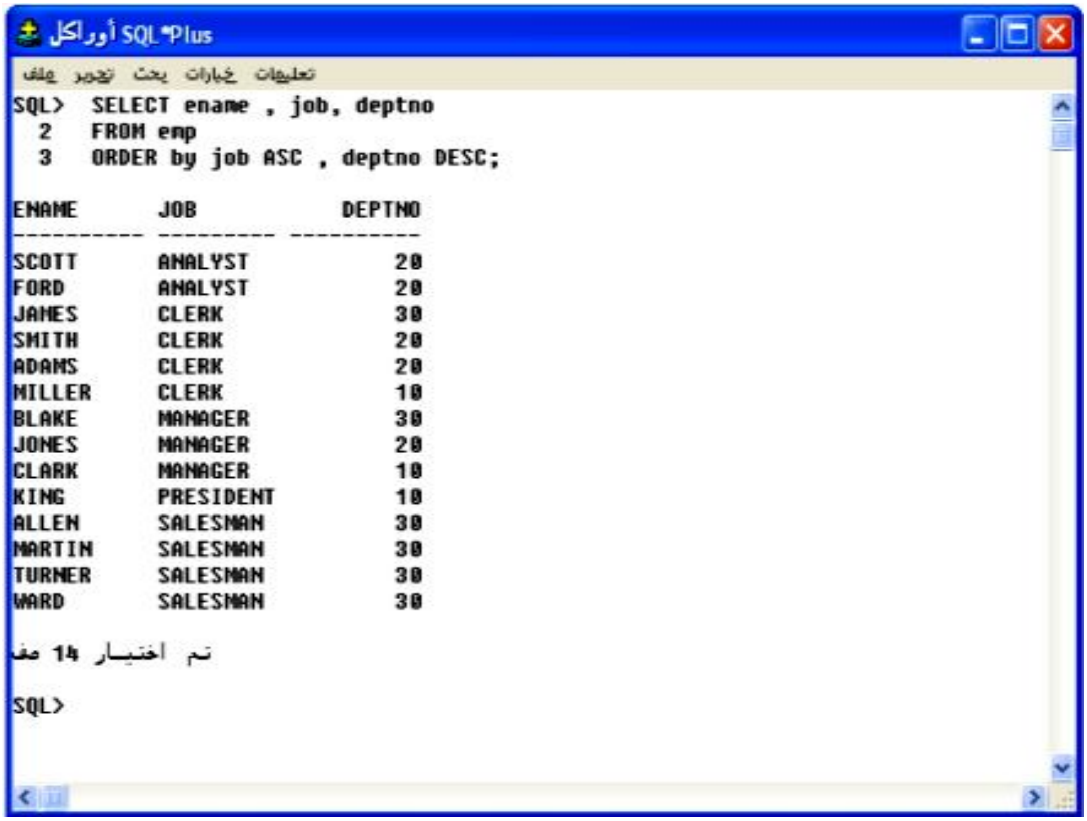
مثال(23)

أعد حل السوال السابق بحيث يكون الترتيب تنازلياً:

الحل:



ملاحظة : يمكن أن نستخدم حقل تصاعدي و الأخر تنازلي في الترتيب كما توضح الشاشة التالية:



2.1.2 دوال السجل Single-Row Function

عزيزي الدارس، تعطي الدوال التي يمكن أن تنفذ على مستوى السجلات إمكانيات عالية لتنفيذ معالجات مختلفة على السجلات و نذكر منها:

أ- تنفيذ عمليات حسابية على البيانات.

ب- تعديل أجزاء البيانات بصورة فردية.

ت- التعامل مع مجوعة من السجلات.

ث- تغير صيغ عرض البيانات من نوع تاريخ و عددي.

ج- تحويل أنواع البيانات.

هنالك نوعين لهذه الدوال متمثلة في الآتي:

① Single-Row Functions

هذا النوع من الدوال يعمل على مستوى السجلات، و يكون لدينا ناتج واحد يقابل كل سجل.

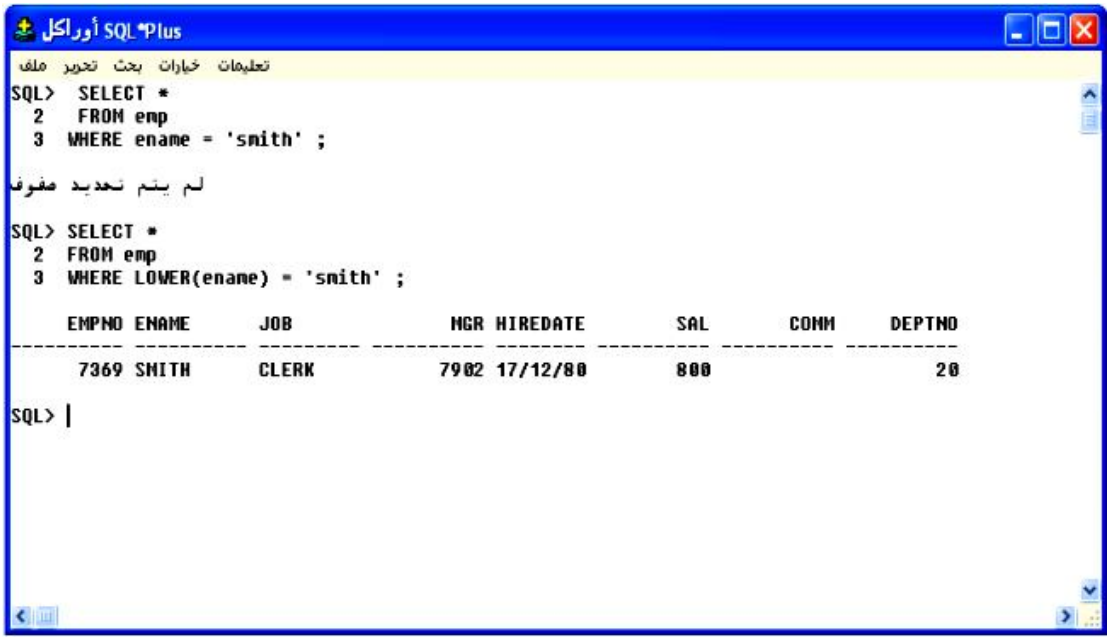
② Multi-Row Function

هذا النوع من الدول يعمل على مجموعة من السجلات و يكون لدينا ناتج واحد لكل هذه السجلات.

و في ما يلي نستعرض جانباً من هذه الدوال:

❐ 1. LOWER ():

تستخدم هذه الدالة لتحويل حالة الأحرف إلى أحرف صـــغيرة . و المثـــال التـــالي يوضـــح استخدام الدالة:



في الاستعلام الأول لا توجد سجلات تحقق الشرط و ذلك لأن حالة الأحرف غير متطابقـــة و لكن في الاستعلام الثاني نضمن أن التطابق سيتم بالنسبة لحالة الأحرف و ذلك بتحويل حالـــة الأحرف لحقل الاسم إلى Lower case.

و هنال دوال اخرى أيضا تتعامل مع حالة الأحرف، و الجدول التالي يوضح ذلك:

الدو ال	ناتج التنفيذ
LOWER('SQL Course')	sql course
UPPER('SQL Course')	SQL COURSE
INITCAP('SQL Course')	Sql Course

الجدول (1-3)

❐ دوال التعامل مع الأحرف Character Manipulation Functions

و تشمل هذه المجموعة الدوال التالية:

الناتج	الدوال
HelloWorld	CONCAT('Hello', 'World')
Hello	SUBSTR('HelloWorld',1,5)
10	LENGTH('HelloWorld')
6	INSTR('HelloWorld', 'W')
*****24000	LPAD(salary,10,'*')
24000*****	RPAD(salary, 10, '*')
elloWorld	TRIM('H' FROM 'HelloWorld')

جدول (2-3)

حيث إن :

① CONCAT لإلحاق نصين ببعضهم البعض

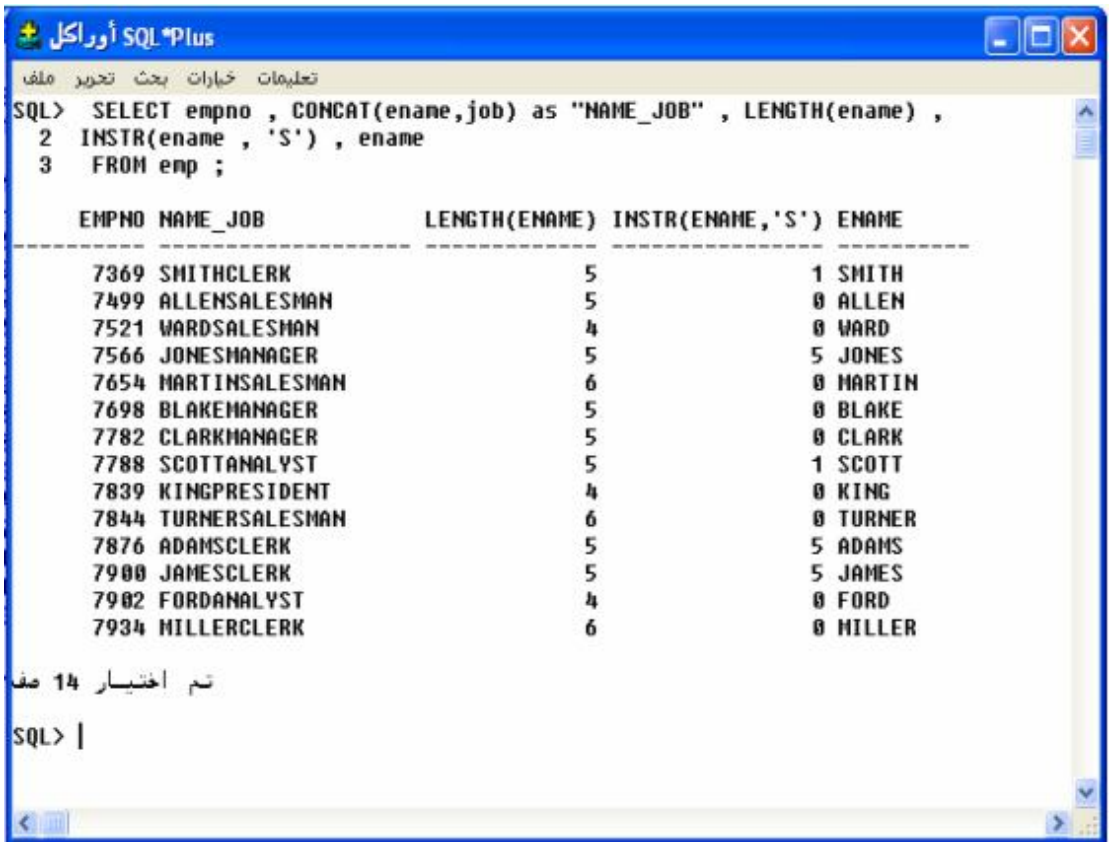
② SUBSTR لإستخلاص جزء من النص بطول معين.

③ INSTR لمعرفة موقع نص داخل النص

④ LPAD , RPAD لعمل تنسيق لقيمة الحقل بإضافة رموز من الجهة اليمنى أو الجهة اليسرى.

⑤ TRIM لحذف جزء من النص.

المثال التالي يوضح عمل بعض من تلك الدوال:



أسئلة تقويم ذاتي

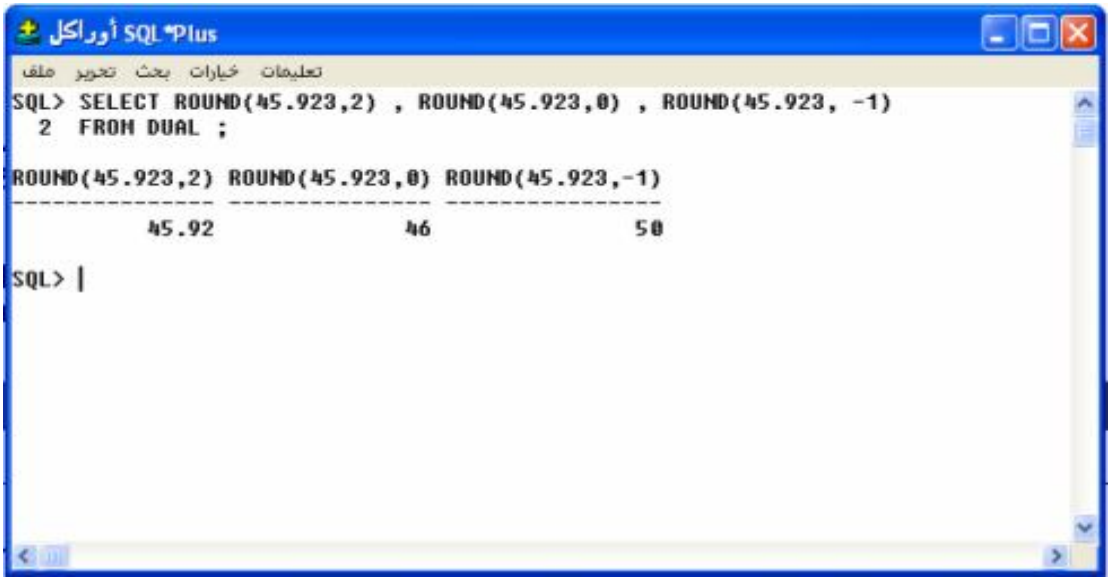
1/ تعطي الدوال التي يمكن أن تنفذ على مستوى السجلات إمكانيات عالية لتنفيذ معالجات مختلفة على السجلات، أذكر بعضاً من هذه المعالجات.

2/ أكمل الجدول التالي الذي يوضح دوالاً تتعامل مع حالة الأحرف :

الدوال	ناتج التنفيذ
..... (' SQL Course ')	Sql course
..... (' SQL Course ')	SQL COURSE
..... (' SQL Course ')	Sql Course

❐ الدالة ROUND

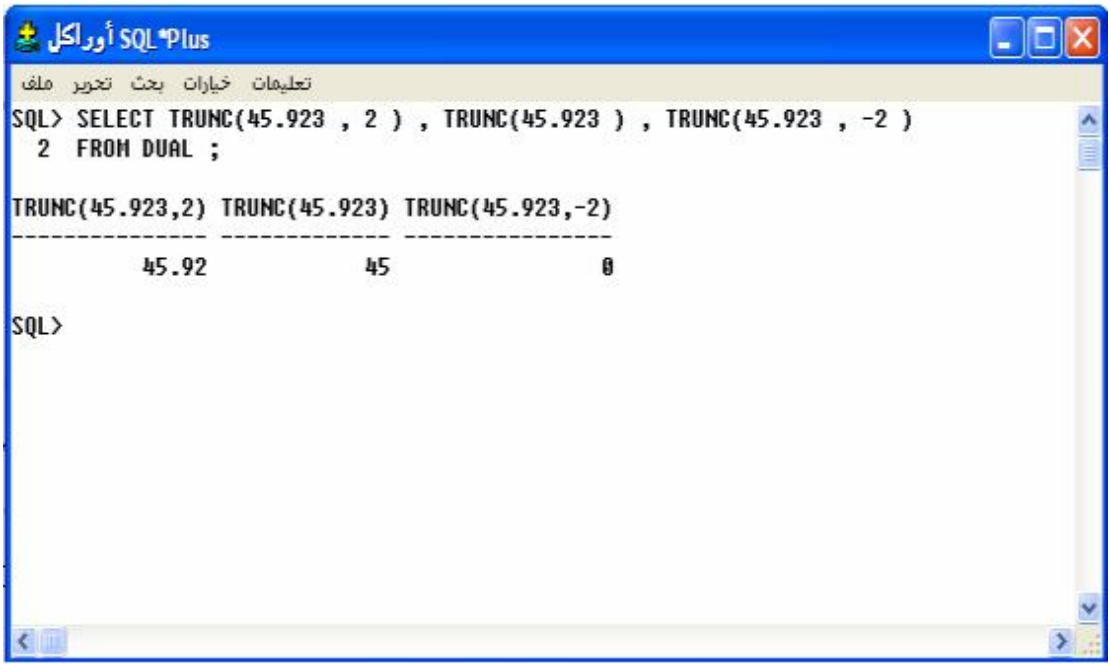
تستخدم هذه الدالة لتقريب الأعداد. و المثال التالي يوضح عمل هذه الدالة:



نلاحظ أن العدد الأول تم تقريبه لخانتين، وذلك لإن العدد 3 أصغر من 5، العدد الثاني بـــدون جزء عشري لأن خانة التقريب صفر، و العدد الاخير إلى 50، و ذلك لأن5 في 45 أقـــرب لتكون 10 و تصبح 50 و استخدمنا 1- .

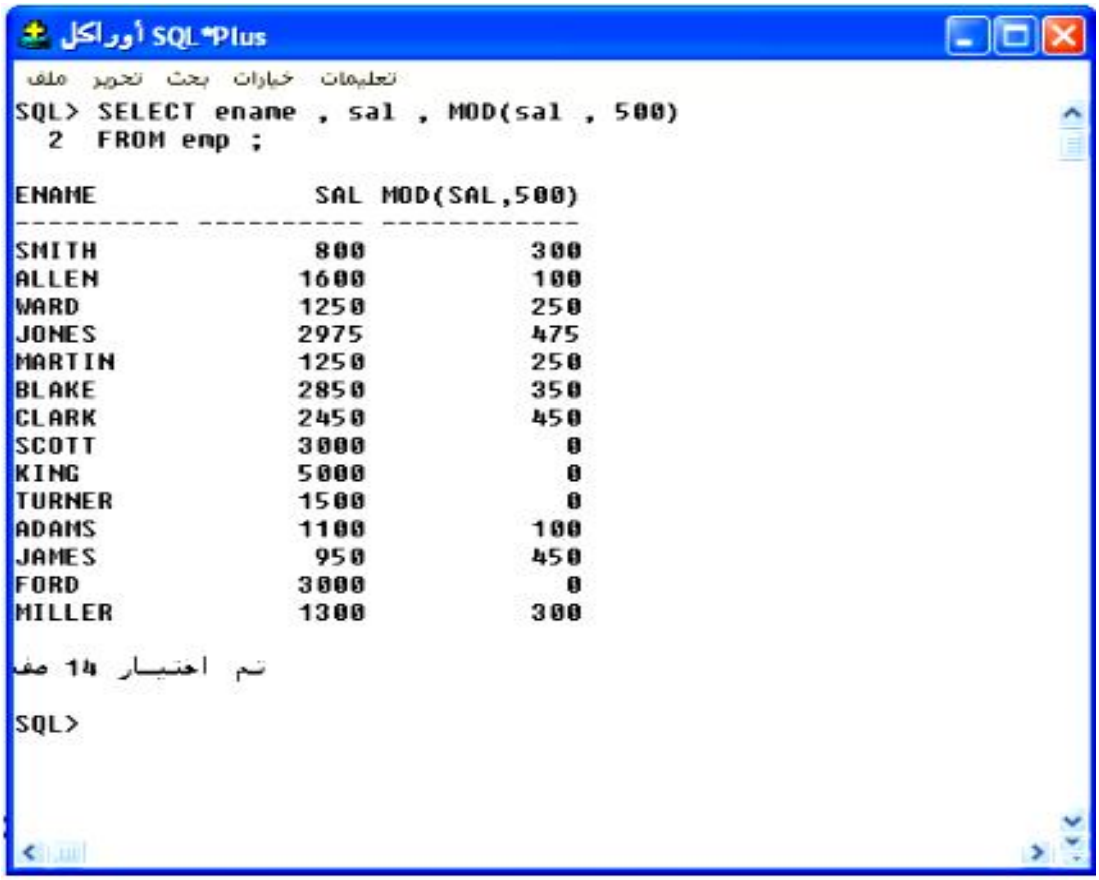
❐ الدالة TRUNC

تقوم هذه الدالة باستقطاع جزء من العدد دون التقريب، و المثال التالي يوضح ذلك:



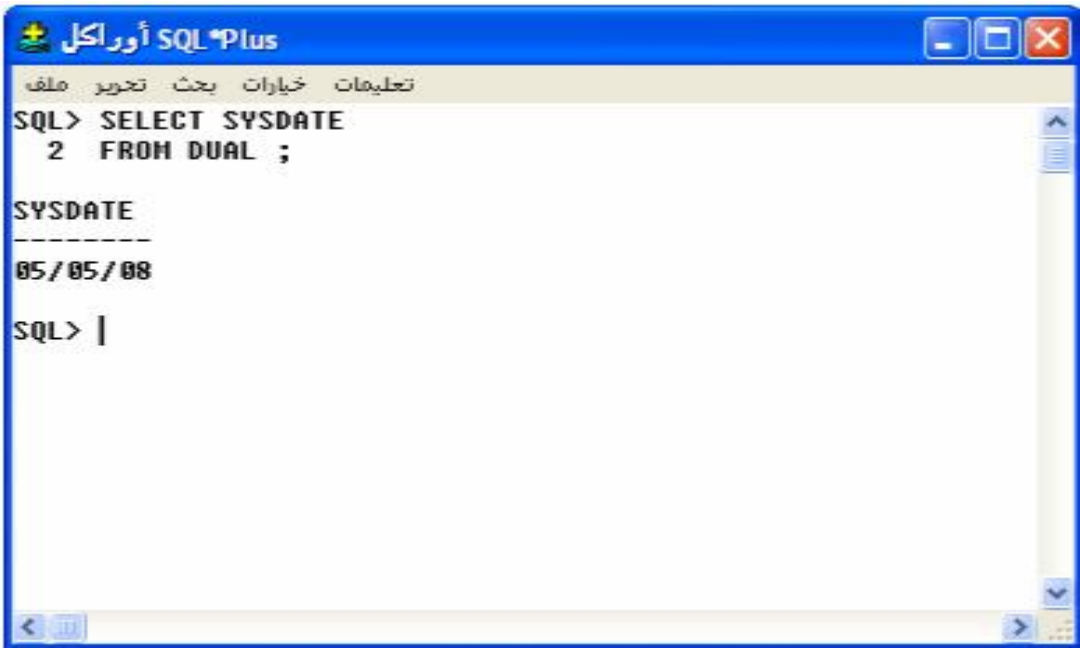
❐ الدالة MOD

تستخدم هذه الدالة لحساب باقي القسمة و المثال التالي يوضح عمل هذه الدالة :



❐ الدالة SYSDATE

تستخدم هذه الدالة لاسترجاع التاريخ الحالي للجهاز. و المثال التالي يوضح عمل هذه الدالة.



و يمكن أن نجري عمليات حسابية على حقل التاريخ بحيث يمكننا إضافة عـدد أو إنقـاص عدد من التاريخ، أو إضافة عدد من الساعة.... و هكذا. و يكون ناتج كل عملية هـو تـاريخ جديد.

❐ دوال التاريخ Date Function

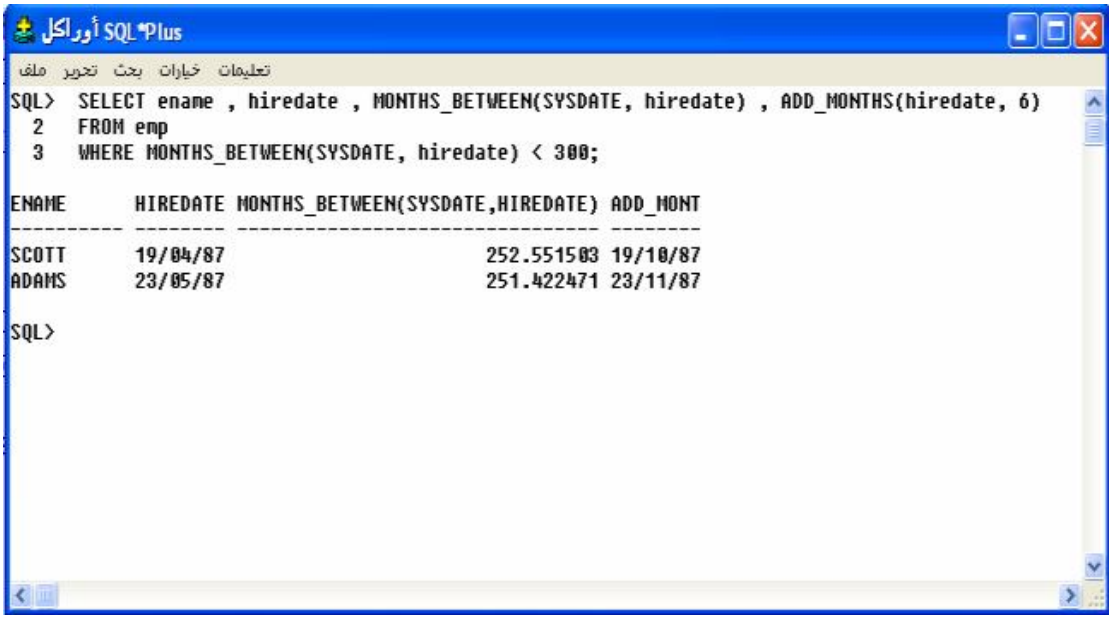
يمكن استخدام الدوال التالية للتعامل مع التاريخ:

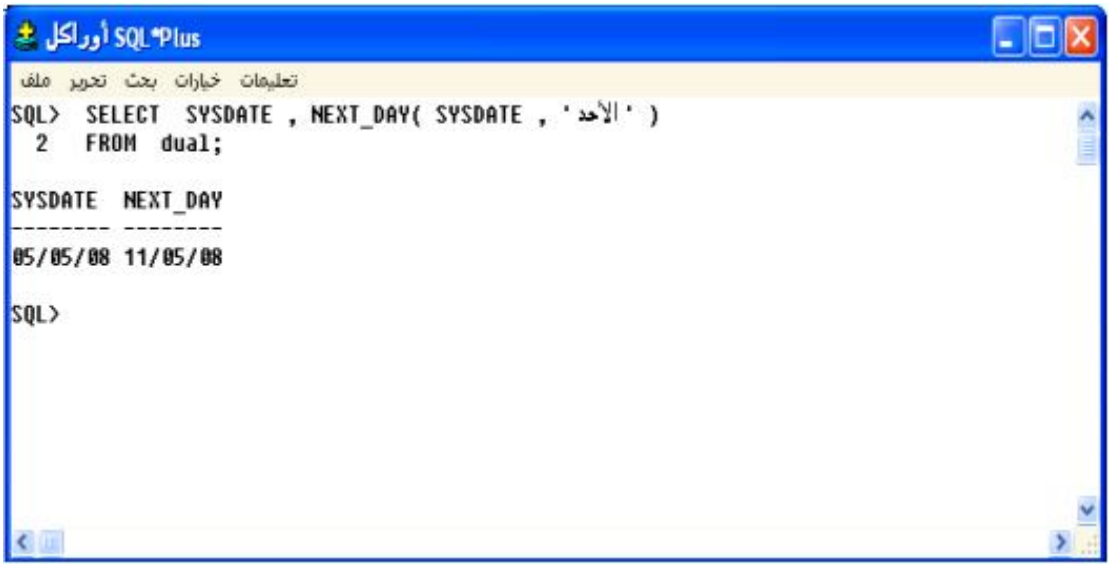
الدالة	الوصف
MONTHS_BETWEEN	لإسترجاع عدد الأشهر بين تارخين
ADD_MONTHS	أضافة عدد من الأشهر إلى تاريخ محدد
NEXT_DAY	اليوم التالي حسب تاريخ محدد
LAST_DAY	إيجاد اخر يوم في الشهر حسب تاريخ معين

الجدول (3-3)

و الأمثلة التالية توضح ناتج تنفيذ بعض من هذه الدوال:

- `MONTHS_BETWEEN ('01-SEP-95','11-JAN-94')` → `19.6774194`
- `ADD_MONTHS ('11-JAN-94',6)` → `'11-JUL-94'`
- `NEXT_DAY ('01-SEP-95','FRIDAY')` → `'08-SEP-95'`
- `LAST_DAY('01-FEB-95')` → `'28-FEB-95'`


أوراكل SQL*Plus
ملف تحرير بحث خيارات تعليمات
SQL> SELECT ename , hiredate , MONTHS_BETWEEN(SYSDATE, hiredate) , ADD_MONTHS(hiredate, 6)
2 FROM emp
3 WHERE MONTHS_BETWEEN(SYSDATE, hiredate) < 300;
ENAME HIREDATE MONTHS_BETWEEN(SYSDATE,HIREDATE) ADD_MONT
SCOTT 19/04/87 252.551503 19/10/87
ADAMS 23/05/87 251.422471 23/11/87
SQL>


أوراكل SQL*Plus
ملف تحرير بحث خيارات تعليمات
SQL> SELECT SYSDATE , NEXT_DAY(SYSDATE , 'الأحد')
2 FROM dual;
SYSDATE NEXT_DAY
05/05/08 11/05/08
SQL>

أسئلة تقويم ذاتي



1/ أكمل الجدول التالي الذي يمثل دوال للتعامل مع التاريخ :

الدالة	الوصف
MONTHS_BETWEEN	.....
ADD_ MONTHS	.....
NEXT_DAY	.....
.....	إيجاد آخر يوم في الشهر حسب تاريخ معين

❐ **دوال التحويل Conversion Functions**

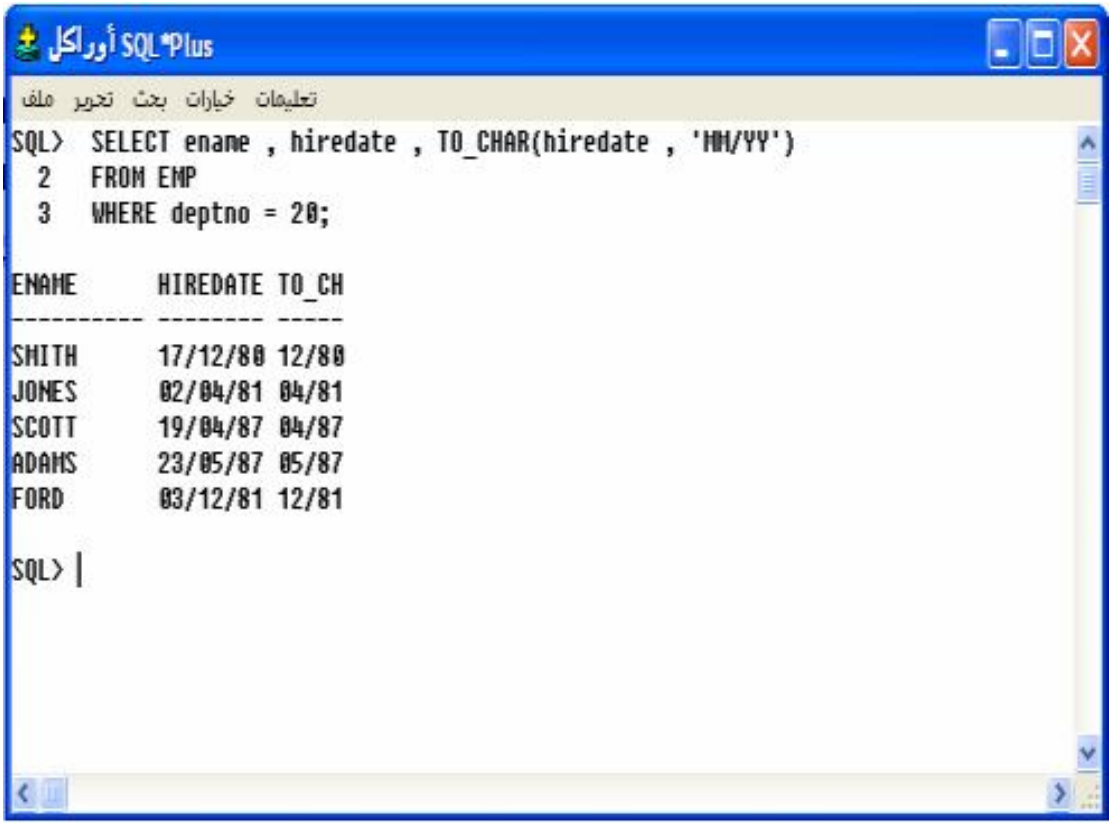
يستخدم هذا النوع من الدوال لتحويل بيانات الحقل من نوع إلى نوع أخر و الجـدول التـالي يوضح البيانات التي يمكن تحويلها:

من	إلى
VARCHAR2 or CHAR	NUMBER
VARCHAR2 or CHAR	DATE
NUMBER	VARCHAR2
DATE	VARCHAR2

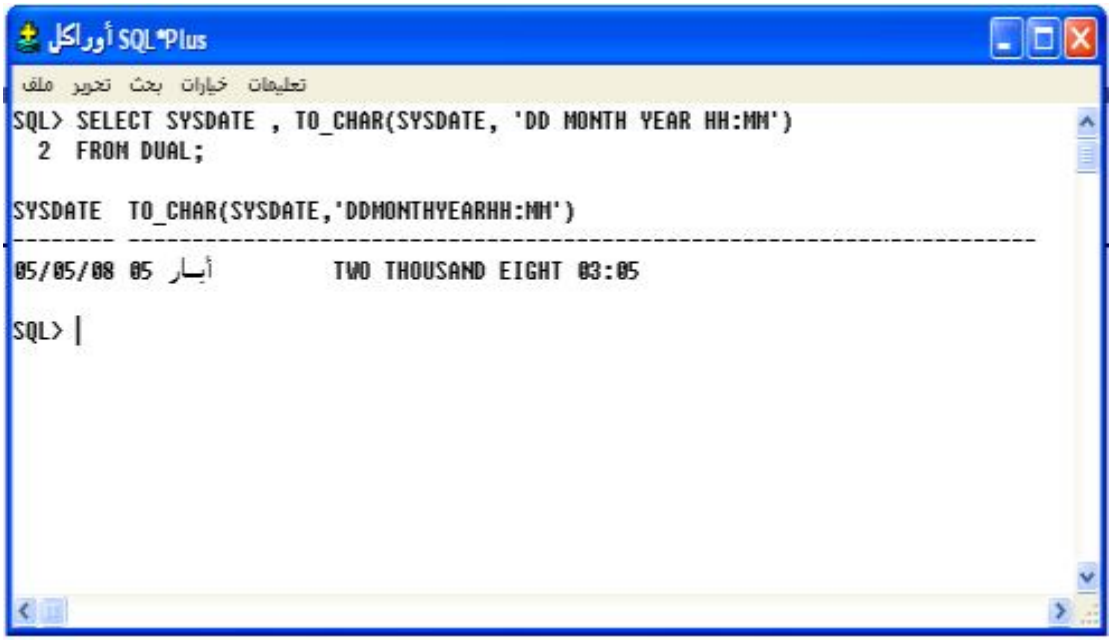
الجدول (3-4)

❐ **الدالة TO_CHAR( )**

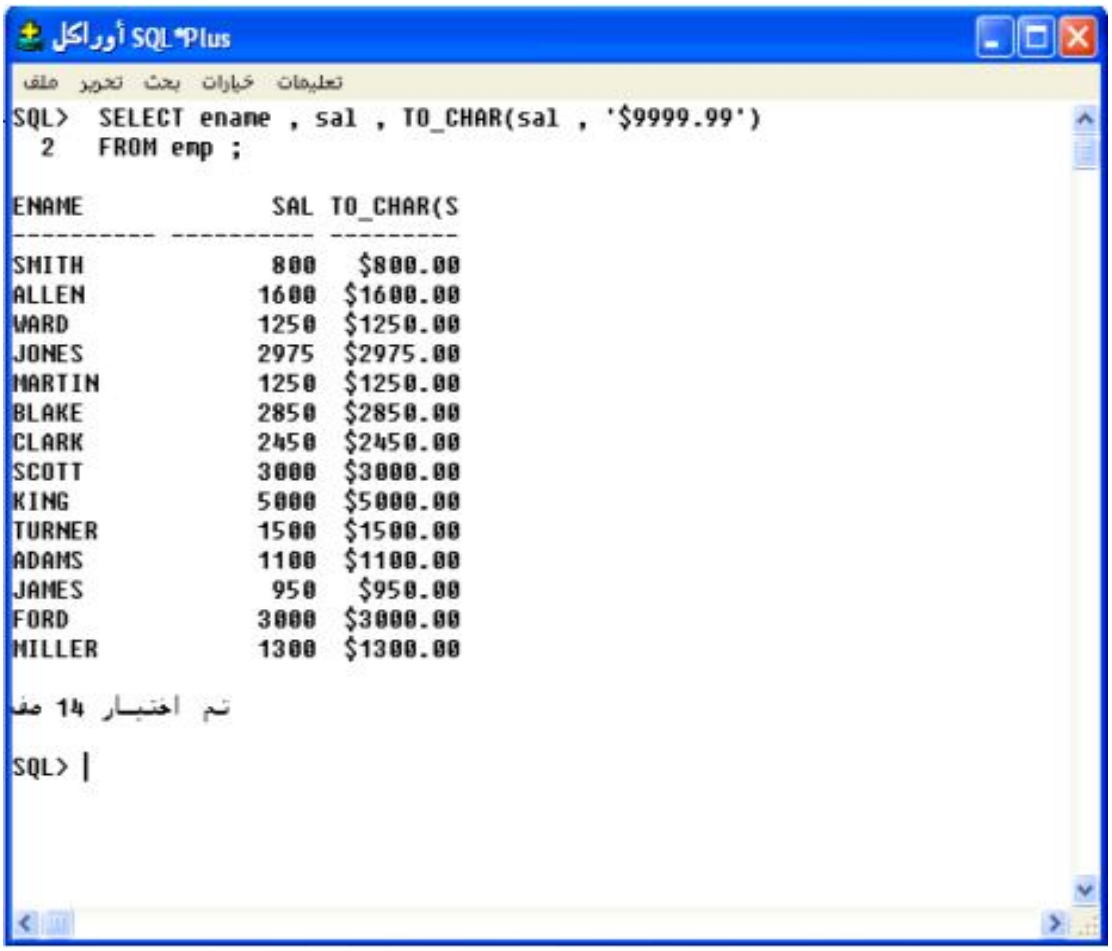
تستخدم هذه الدالة لتحويل البيانات من نوع تاريخ أو نوع عددي إلى نوع نصـي. و الأمثلـة التالية توضح ذلك:



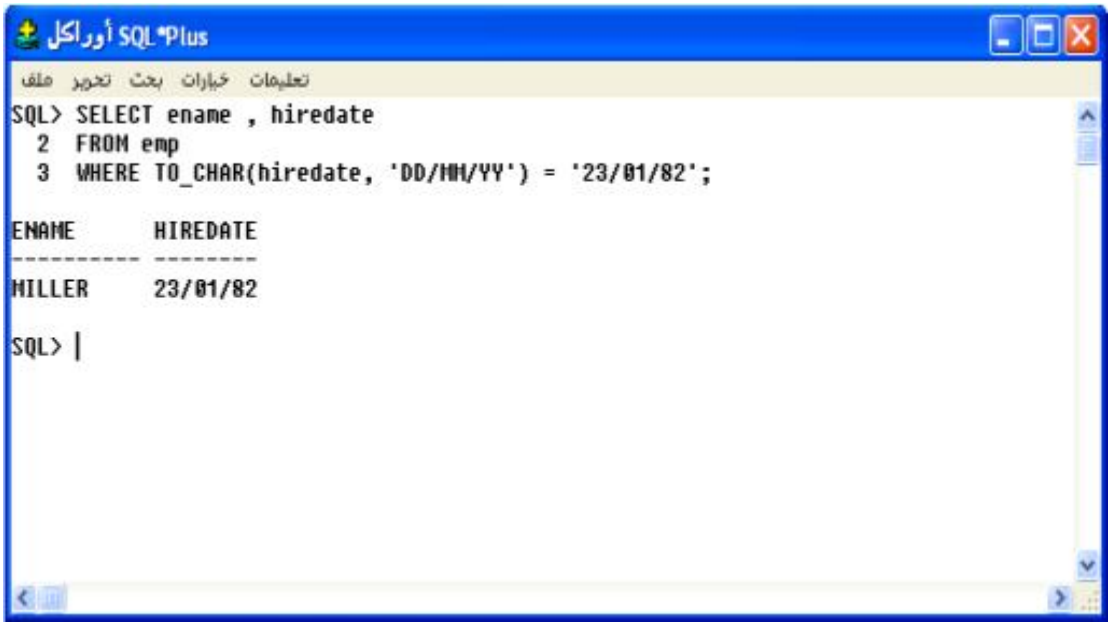
في هذا المثال تم تحويل تاريخ التعيين إلى الصيغة MM/YY اي الشهر و السنة.



في هذا المثال تم تحويل تاريخ جهاز الحاسوب إلى الشهر و السنة بصورة نصـــية، و كـــذلك الوقت.



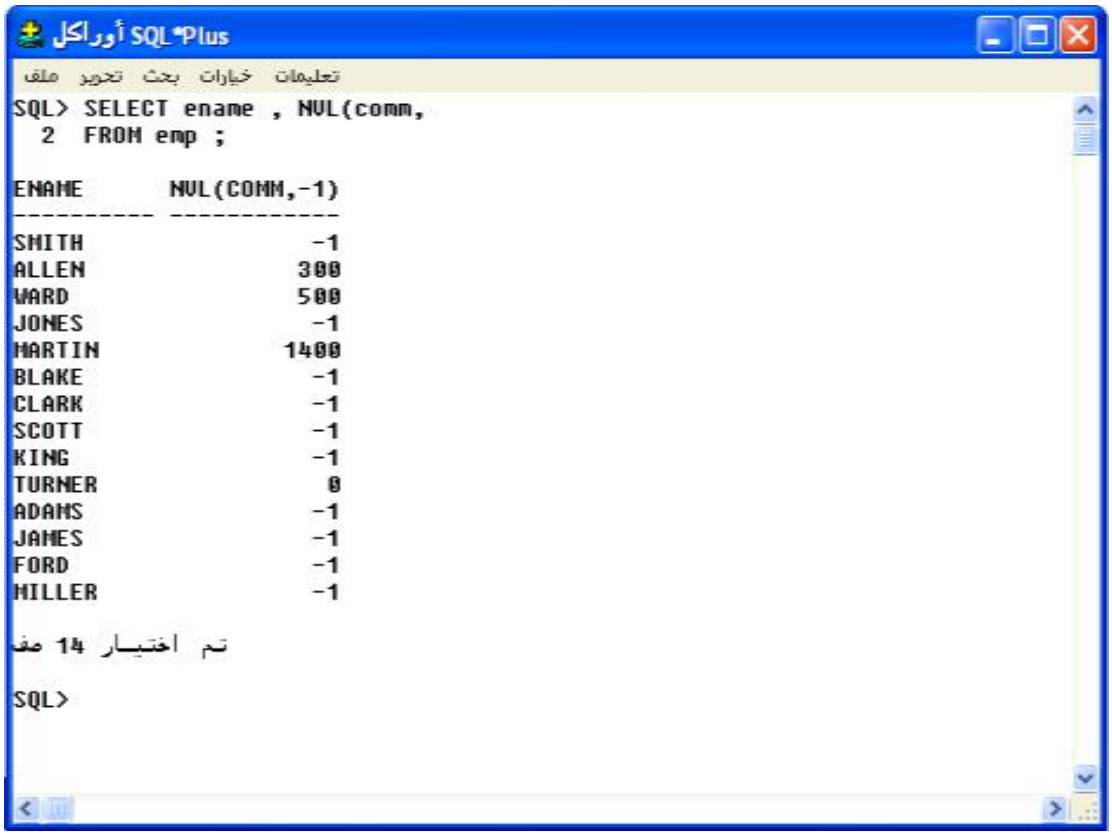
تم استخدام الدالة لتحويل الراتب في الصيغة لأربعة أعداد صحيحة و عددين عشريين و تسبق تلك الاعداد علامة $



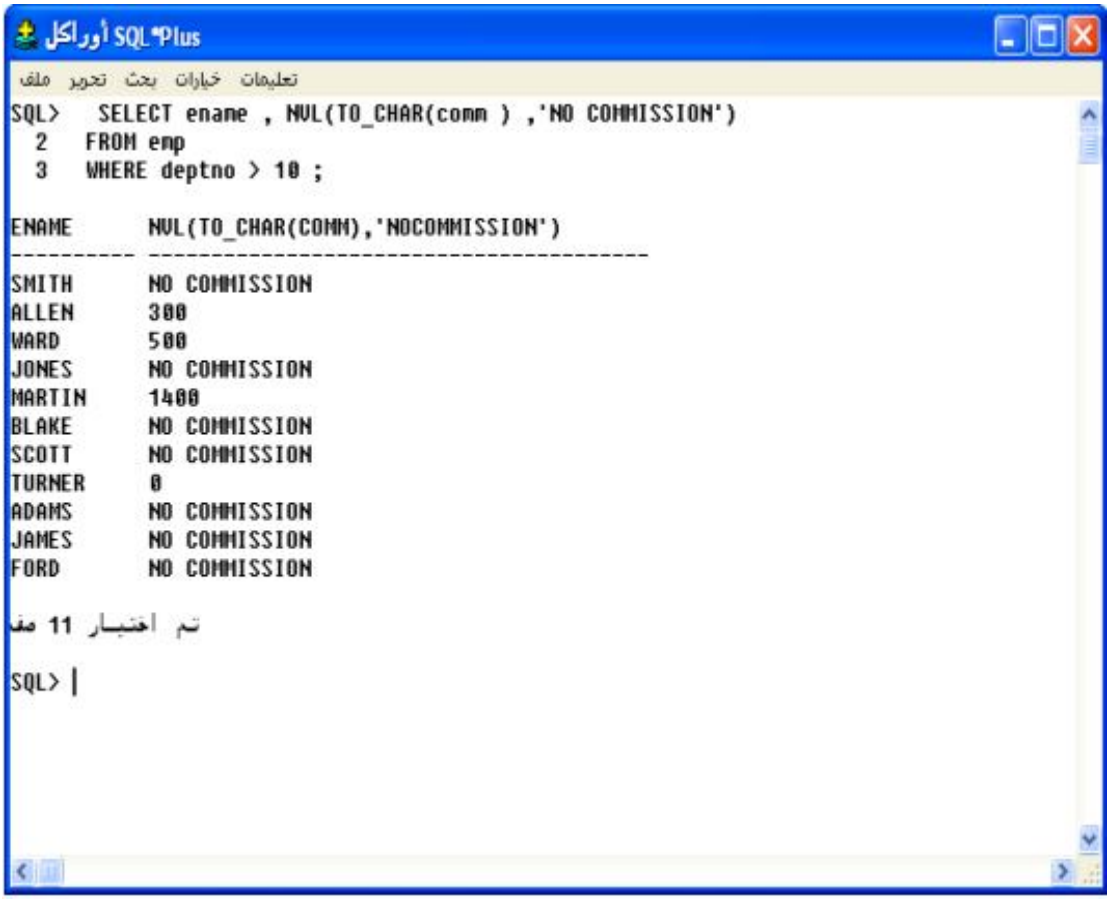
تم استخدام الدالة لمقرانتها مع قيمة نصية ثابته تمثل تاريخ معين.

❐ الدالة NVL( )

تستخدم هذه الدالة لاستبدال قيمة معينة في حالة أن قيمة الحقل غير معرفـــة. المثـــال التـــالي يوضح ذلك.



تم وضع قيمة 1- في حالة أن COMM يحمل NULL.



تم كتابة عبارة NO COMMISSION في حالة أن COMM يحتوي NULL و حتـــي تنسجم هذه الرسالة مع نوع الحقل نلاحظ أنه تم تحويل COMM إلى نص.

3.1.2 التعامل مع عدة جداول

Displaying Data From Multiple Data

عزيزي الدارس، في بعض الحالات قد تكون البيانات المطلوبة موجودة في أثر من الجـــدول. وفي هذه الحالة يتطلب عمل استعلام لكي يتعامل مع هذه الجداول. في هـــذا الفصـــل ســـوف نتناول عملية قراءة البيانات من عدة جداول على أن تكون عملية القراءة باستخدام شـــرط أو من دون ذلك. مع العلم أن القراءة يمكن أن تتم من جداول مختلفة أو نفس الجدول، علـــى أن يفتح ذلك الجدول أكثر من مرة.

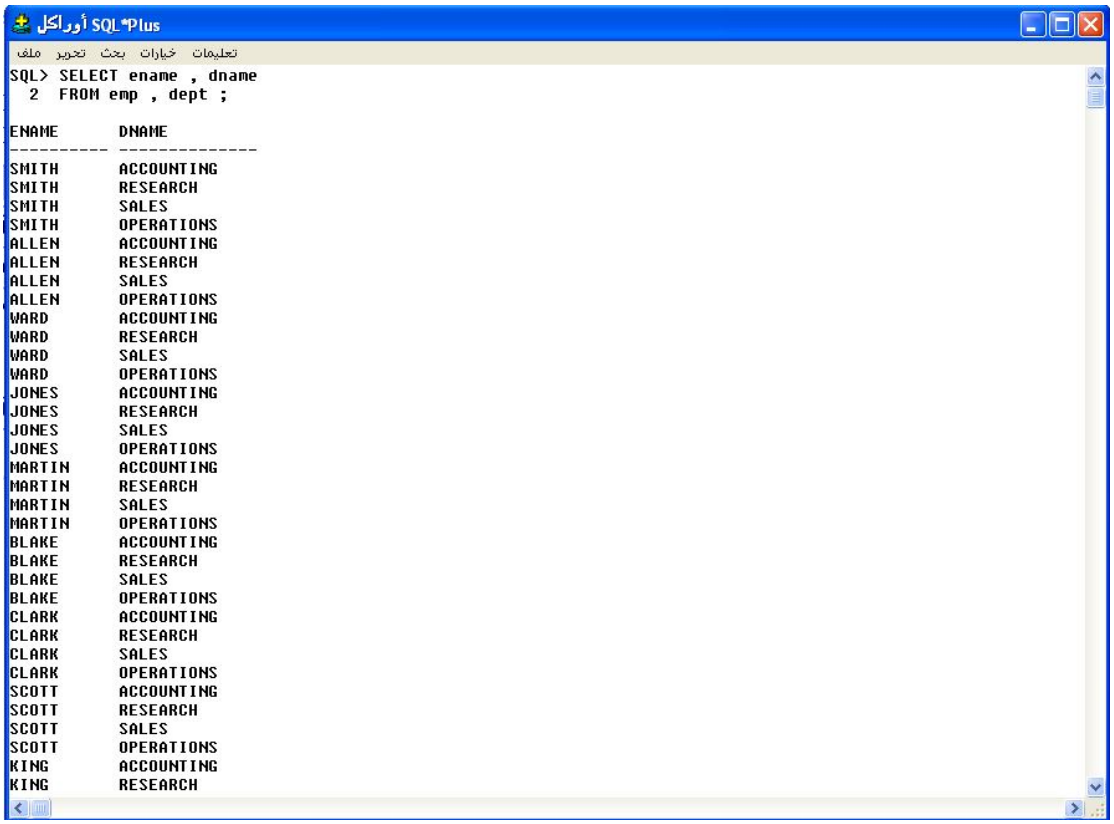
❐ **الاستعلام بدون شرط الربط بين الجداول Cartesian Product**

في هذا النوع من الاستعلام لا يوجد شرط. ويكون نـــاتج الاســـتعلام الحاصـــل الكـــارتيزي للجداول التي تم التعامل معها.

مثال(24):

اكتب جملة استعلام تقوم بعرض أسماء الموظفين و أسماء الإدارات من جدول المـــوظفين و جدول الإدارات.

الحل:




```
SQL> SELECT ename , dname
  2  FROM emp , dept ;

ENAME      DNAME
---------- --------------
SMITH      ACCOUNTING
SMITH      RESEARCH
SMITH      SALES
SMITH      OPERATIONS
ALLEN      ACCOUNTING
ALLEN      RESEARCH
ALLEN      SALES
ALLEN      OPERATIONS
WARD       ACCOUNTING
WARD       RESEARCH
WARD       SALES
WARD       OPERATIONS
JONES      ACCOUNTING
JONES      RESEARCH
JONES      SALES
JONES      OPERATIONS
MARTIN     ACCOUNTING
MARTIN     RESEARCH
MARTIN     SALES
MARTIN     OPERATIONS
BLAKE      ACCOUNTING
BLAKE      RESEARCH
BLAKE      SALES
BLAKE      OPERATIONS
CLARK      ACCOUNTING
CLARK      RESEARCH
CLARK      SALES
CLARK      OPERATIONS
SCOTT      ACCOUNTING
SCOTT      RESEARCH
SCOTT      SALES
SCOTT      OPERATIONS
KING       ACCOUNTING
KING       RESEARCH
```

في هذا المثال نلاحظ أنه تم تكرار اي اسم من جدول الموظفين مع كل اسماء الإدارات (ناتج حاصل الضرب الكارتيزي).

❐ الاستعلام بشرط الربط بين الجداول Join Condition

هنالك عدة أنواع يمكن أن تستخدم لعمل ربط بين الجداول التي ترد في الاســتعلام الواحــد و هي كما يلي:

- الربط بالتساوي Equijoin Condition
- الربط بعدم التساوي NonEquijoin Condition
- الربط الخارجي Outer Join
- الربط الذاتي Self Join

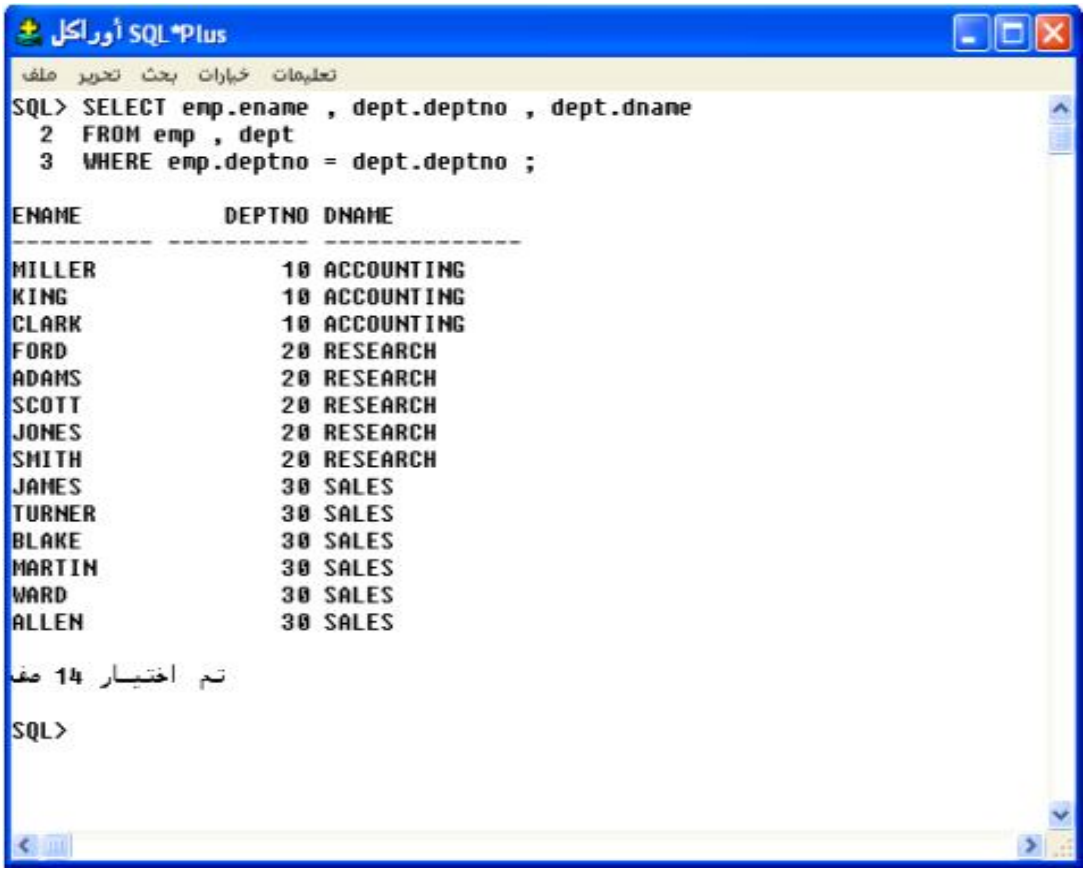
الصيغة العامة للربط بين جداول الاستعلام الواحد هي:

```
SELECT    table1.column, table2.column
FROM      table1, table2
WHERE     table1.column1 = table2.column2;
```

مثال(25)

اكتب استعلاماً يقوم بعرض أسماء الموظفين و أرقام إداراتهم و اسماء إدارتهم:

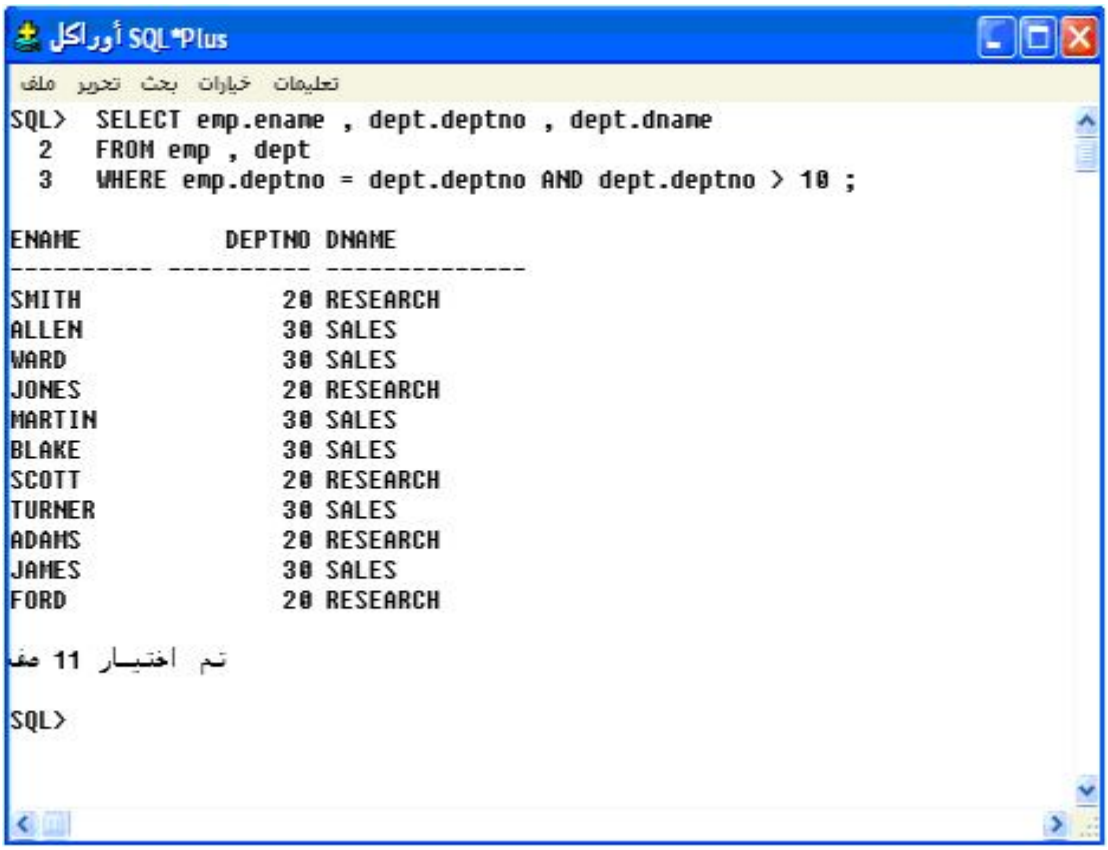
الحل:



مثال(26)

اكتب استعلاماً يقوم بعرض أسماء الموظفين وأرقام اداراتهم و أسماء إدارتهم بشرط أن تزيد ارقام ادارتهم عن 10:

الحل:



ملاحظة

يمكن الإشارة إلى الجداول في الاستعلام باستخدام أسماء مستعارة لتفادي أي إبهام قد يحدث من جراء أن أسماء الحقول متشابهة في الجداول المستخدمة.

تدريب (8)

اكتب استعلاماً يقوم بعرض أسماء الموظفين و أرقام إداراتهم و أسماء إدارتهم بشرط أن أرقام إدارتهم تساوي 10 .

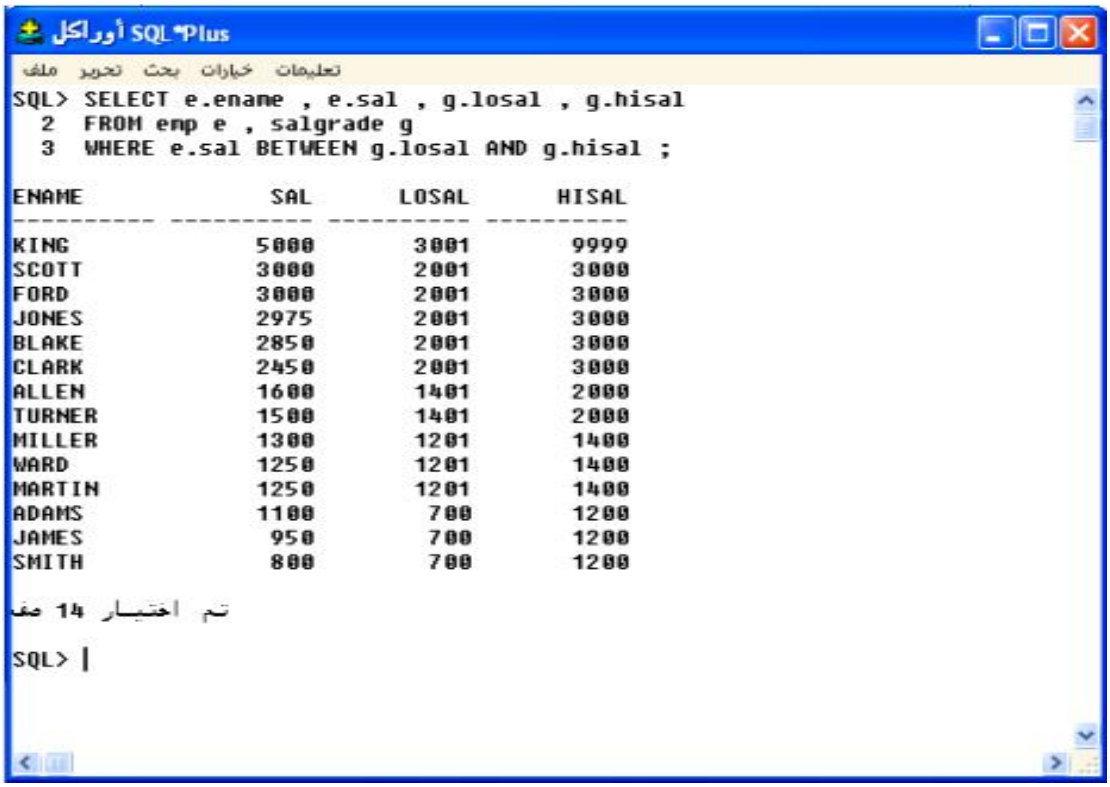
❐ الربط بعدم التساوي Nonequijoin Condition

يمكن أن يكون الربط بين الجداول بعدم التساوي كما يوضح المثال التالي.

مثال(27)

اكتب استعلام يقوم بعرض اسماء و مرتبات الموظفين من جدول الموظفين و السقف الأعلى و الأدني من المرتبات من جدول هيكل المرتبات SalGrade بحيث يكون المرتب يقع بين القيمتين العليا و الدنيا.

الحل:



❐ الربط الخارجي Outer Join

عند استخدام الربط بالتساوي نجد أن كل السجلات المرتبطة في جداول الاستعلام سوف تظهر، ولكن السجلات غير المرتبطة لا تظهر. و حتي نتمكن من إظهار كل السجلات نستخدم الربط الخارجي، و تكون الصيغة كما يلي:

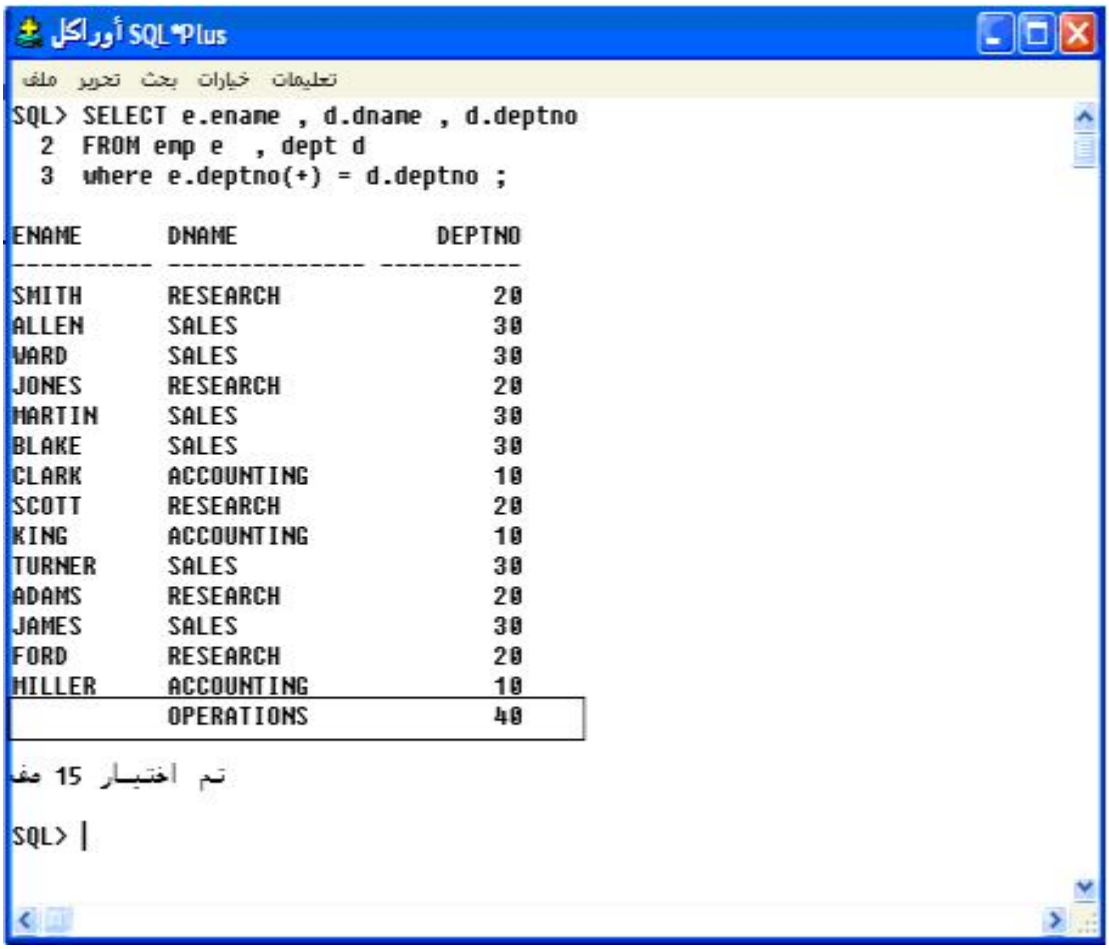
```
SELECT table1.column, table2.column
FROM   table1, table2
WHERE  table1.column(+) = table2.column;
```

```
SELECT table1.column, table2.column
FROM   table1, table2
WHERE  table1.column = table2.column(+);
```

الصيغة الأولى لإظهار كل سجلات Table2 و حتي التي ليس لها سجلات تقابلهـــا فـــي الجدول Table1 .

الصيغة الثانية لإظهار كل سجلات Table1 و حتي التي ليس لها سجلات تقابلهـــا فـــي الجدول Table2

الشاشة التالية توضح ذلك:



❐ الربط الذاتي Self Join

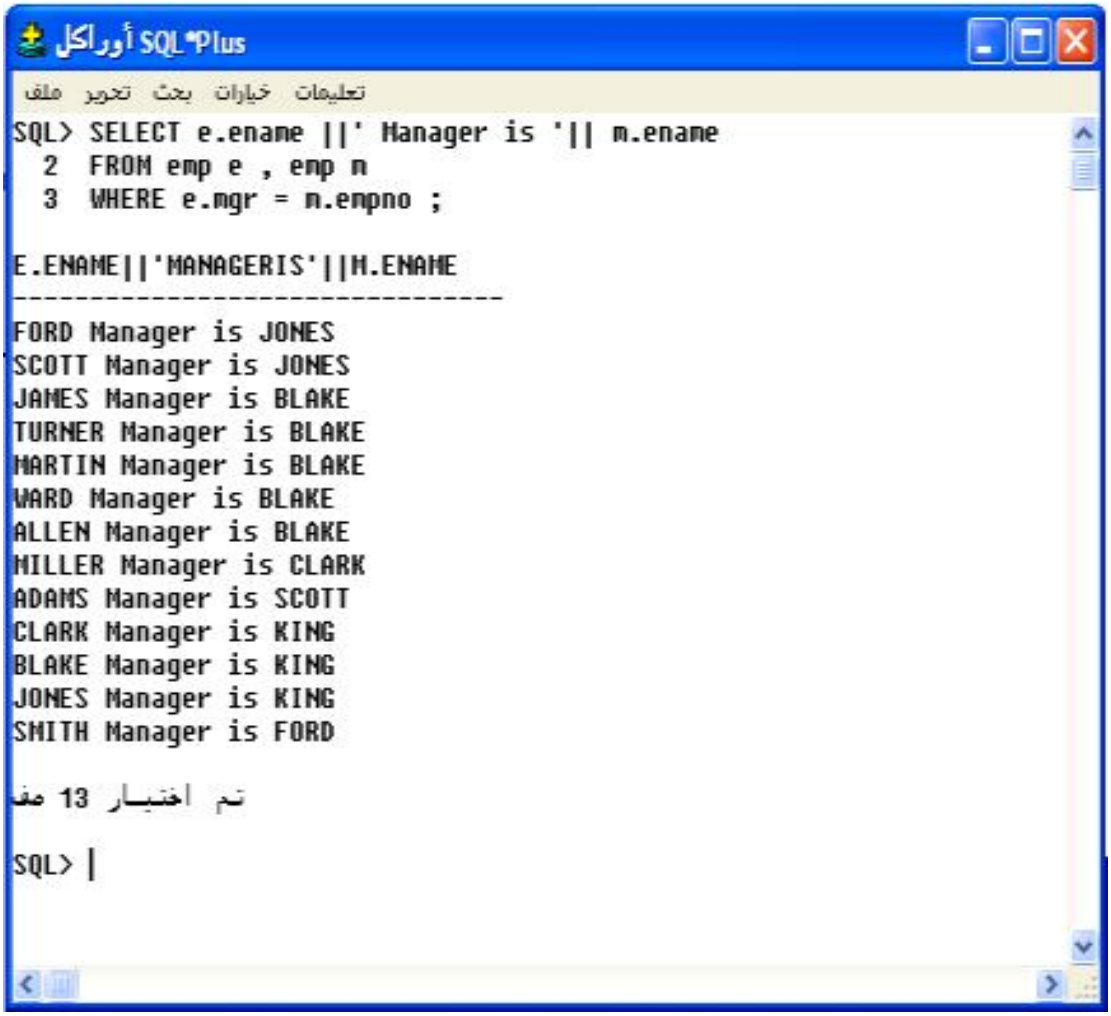
يتيح لنا هذا النوع من الربط قراءة الجدول نفسه عدة مرات و بواســطة نفــس الاســتعلام و المثال التالي يوضح ذلك.

مثال(28)

اكتب استعلاماً يوضح اسم الموظف و اسم المدير.

الحل:

نجد أن كل المعلومات المطلوبة موجودة في جدول الموظفين، و لذلك سيتم فتح هــذا الملــف مرتين.



أسئلة تقويم ذاتي

1/ في الاستعلام بشرط الربط بين الجداول هنالك عدة أنواع يمكن أن تستخدم لعمل ربط بين الجداول التي ترد في الاستعلام الواحد أذكرها.

2/ اكتب الصيغة العامة للربط بين جداول الاستعلام الواحد.

4.1.2 دوال المجموعات Group Function

عزيزي الدارس، يهدف هذا الفصل إلى وصف عمل دوال المجموعات و كيفية عملها و كيفية استخدام العبارة Group By و كيفية عمل الشرط علـى مسـتوى المجموعـة باسـتخدام Having.

و الدوال التي يمكن استخدمها على مجموعة من السجلات توصف كما يلي:

- AVG ( ): تستخدم لايجاد المتوسط الحسابي للحقل العددي، و تتجاهل الحقول التـــي تحمل القيمة NULL.
- COUNT ( ) : تستخدم لمعرفة عدد السجلات، و عند استخدام COUNT(*) فذلك يعني استخدام كل السجلات و لو كانت مكررة أو تحتوي على عبارة NULL.
- MAX( ) : تستخدم لإيجاد أكبر قيمة عددية و تتجاهل NULL.
- MIN( ) : تستخدم لإيجاد أصغر قيمة عددية و تتجاهل NULL.
- SUM( ): تستخدم لجمع مجموعة من الأعداد و تتجاهل NULL.

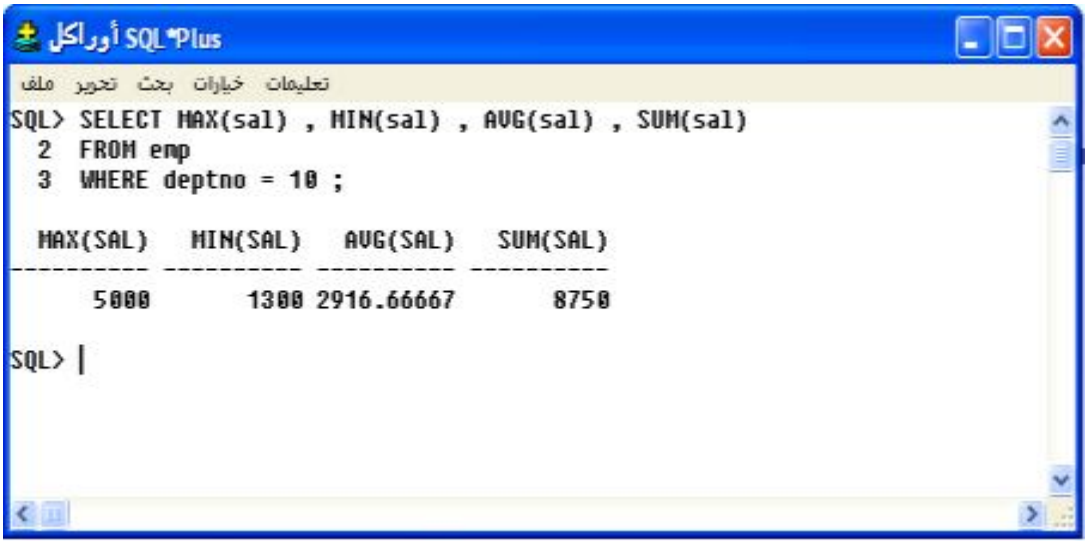
و الصيغة العامة للتعامل مع هذه الدوال هي:

```
SELECT        [column,] group_function(column), ...
FROM          table
[WHERE        condition]
[GROUP BY     column]
[ORDER BY     column];
```

مثال(29)

اكتب جملة استعلام لإيجاد أكبر مرتب و أصغر مرتب و متوسط مرتبات و مجموع المرتبات الخاصة بالموظفين في الإدارة رقم 10:

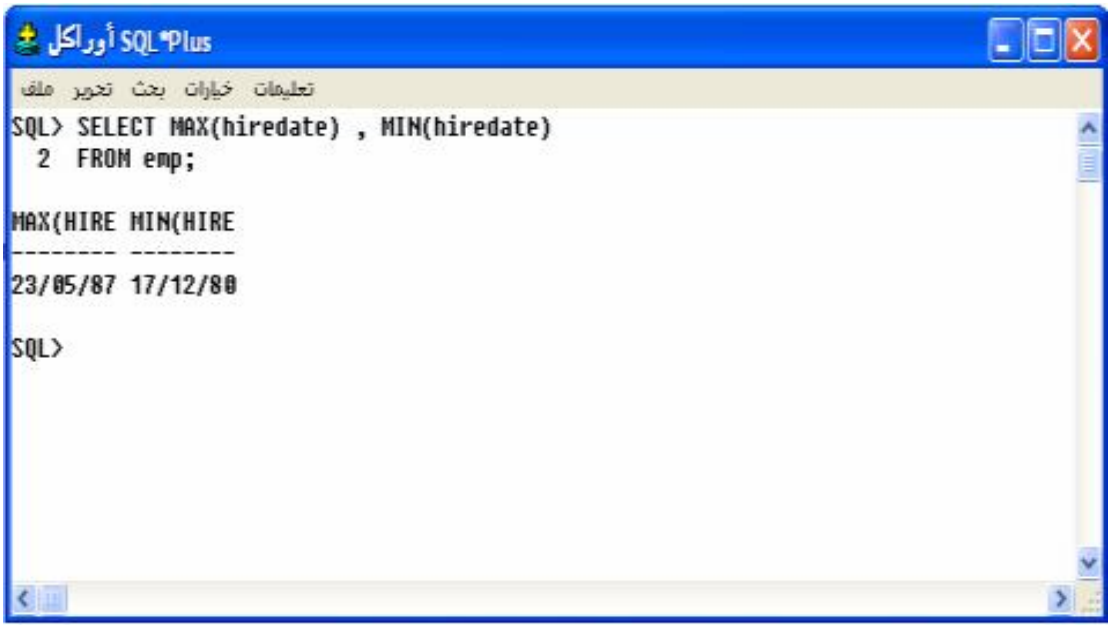
الحل:



مثال(30)

اكتب استعلام يسترجع أحدث تاريخ تعين و أقدم تاريخ موظفين من جدول الموظفين:

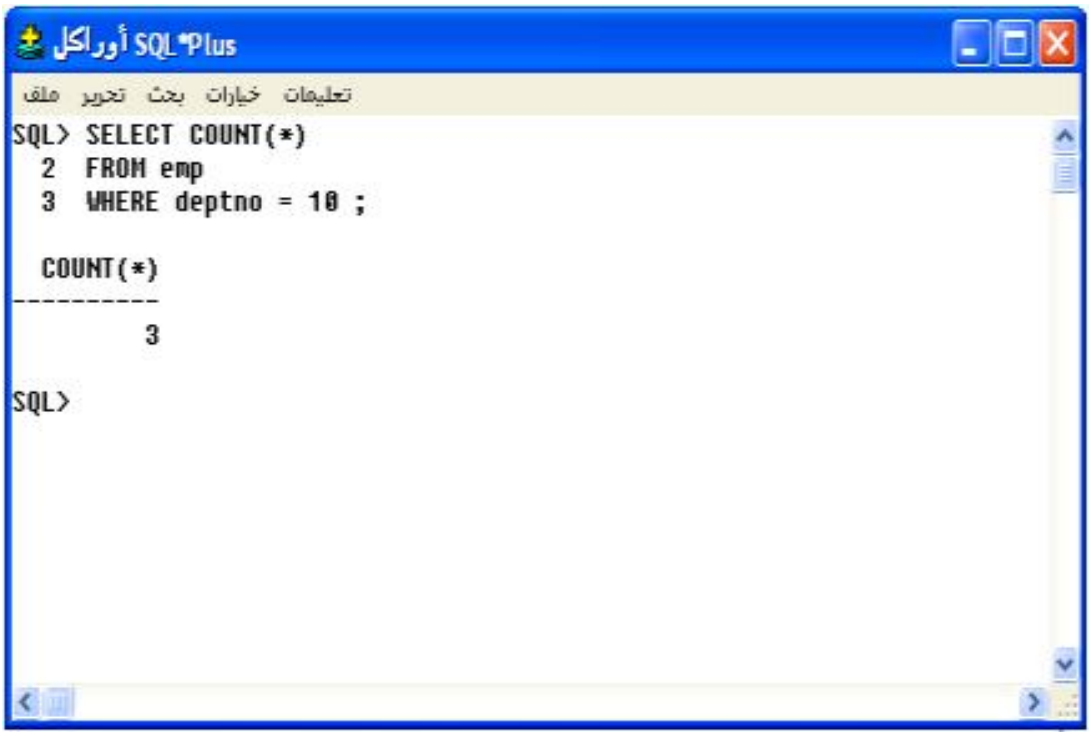
الحل:



مثال(31)

اكتب جملة استعلام لمعرفة عدد الموظفين في الإدارة رقم 10:

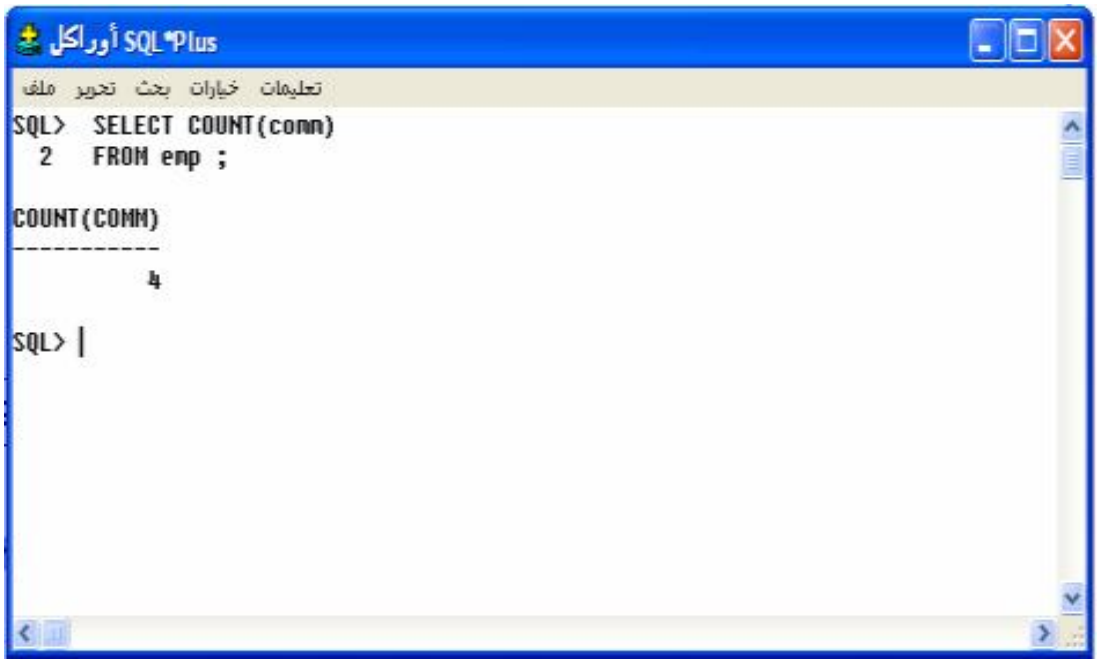
الحل:



مثال(32)

اكتب جملة استعلام لمعرف عدد الموظفين الذين لهم حوافز.

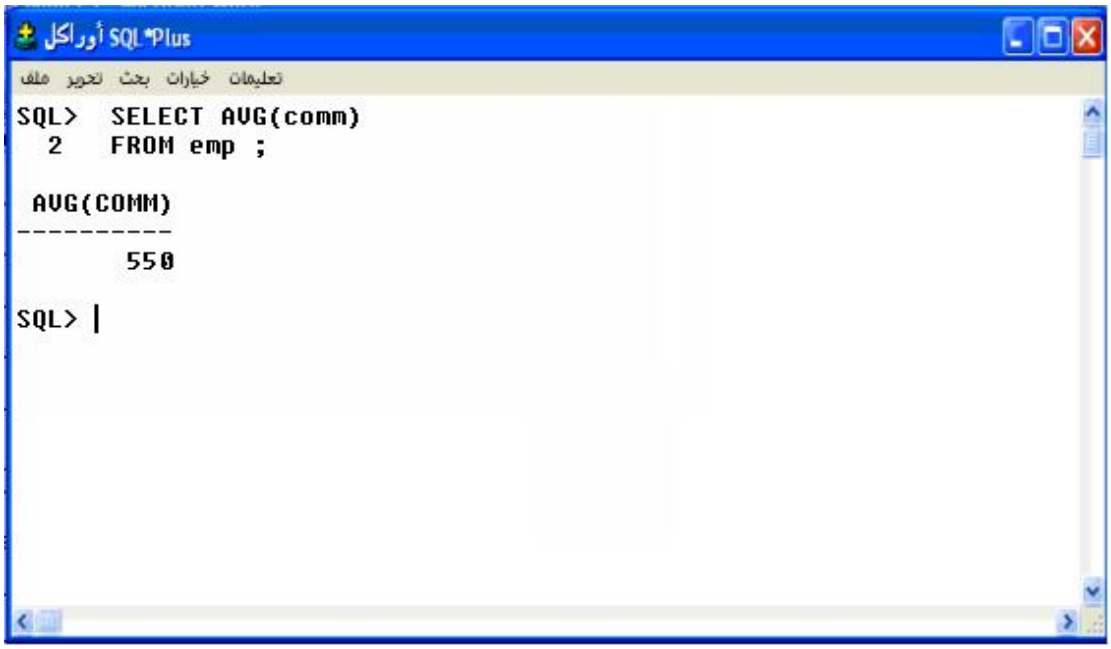
الحل:



مثال(33)

اكتب جملة استعلام لمعرفة متوسط حوافز الموظفين مع استبعاد الموظفين الذين لـــيس لهـــم حوافز:

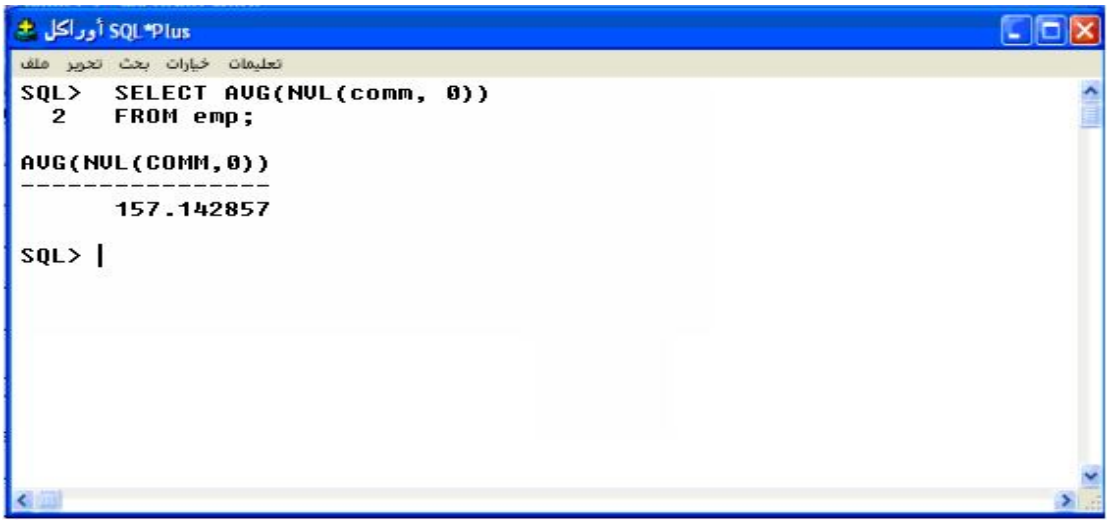
الحل:



مثال(34)

اكتب جملة استعلام لمعرفة متوسط حوافز الموظفين بدون استبعاد الموظفين اللذين ليس لهـــم حوافز:

الحل:



نلاحظ في هذا المثال، أنه تم وضع القيمة صفراً لأي موظف ليس له حوافز.

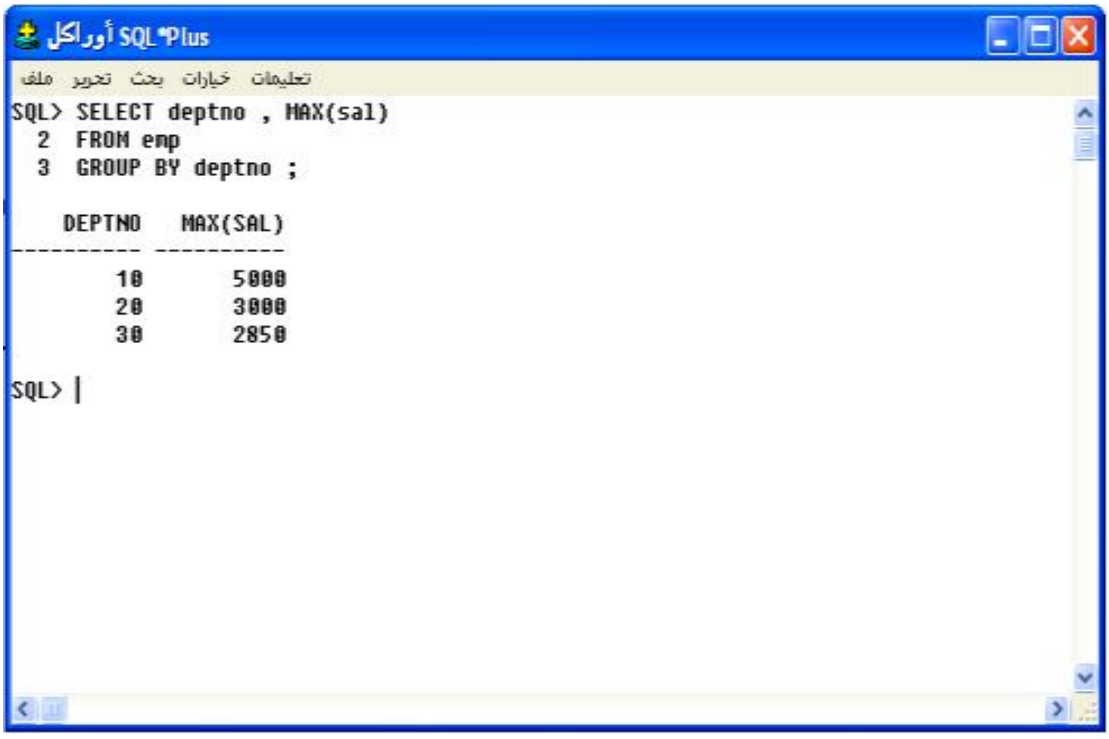
❐ العبارة GROUP BY

تستخدم هذه العبارة لتقسيم السجلات في شكل مجموعات على أن تطبــق دوال المجموعــات على كل مجموعة كل على حدة و بشكل مستقل. مع العلم أن عملية التقسيم يمكن أن تتم فــي عدة مستويات.

مثال(35)

اكتب جملة استعلام لمعرفة أكبر مرتب في كل إدارة.

الحل:



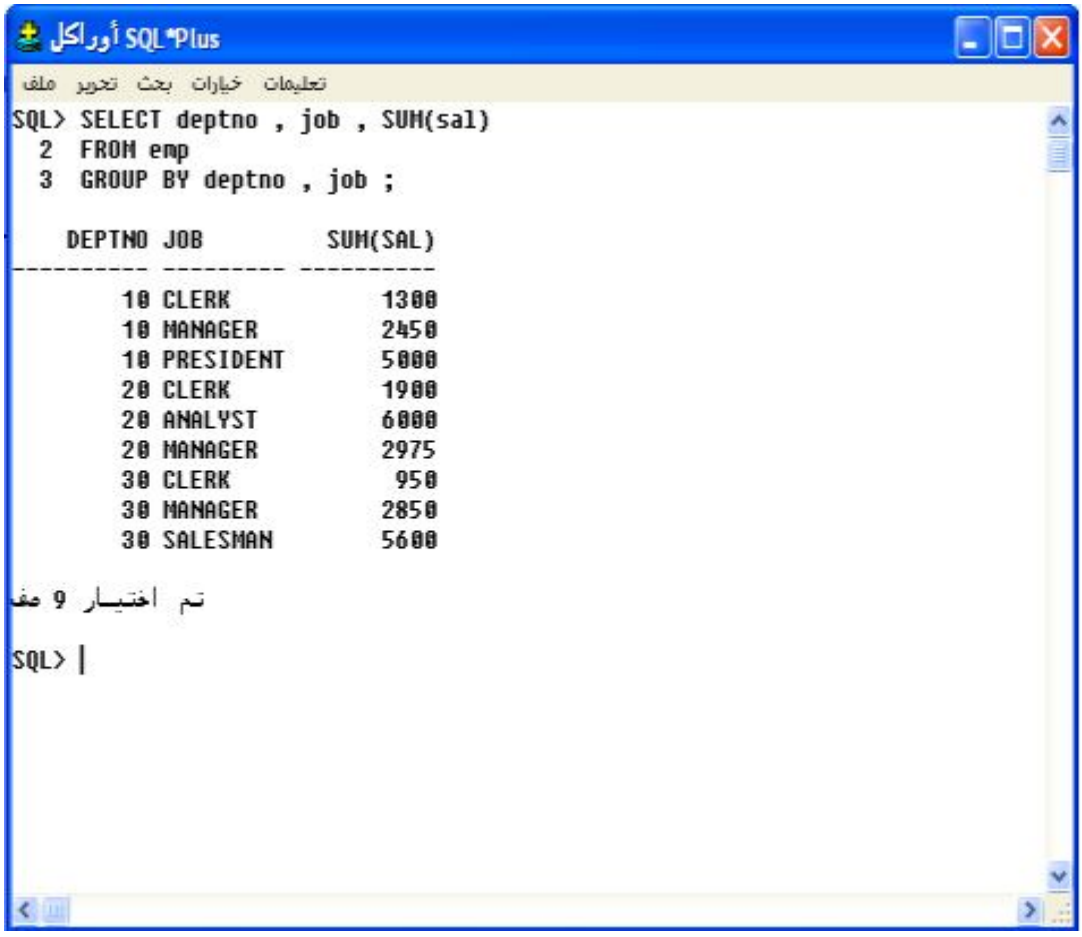
تدريب (9)

اكتب جملة استعلام لمعرفة عدد الموظفين في كل وظيفة.

مثال(36)

اكتب جملة استعلام لإيجاد مجموع مرتبات الموظفين في كل وظيفة علـــى مســـتوى الإدارات المختلفة:

الحل:

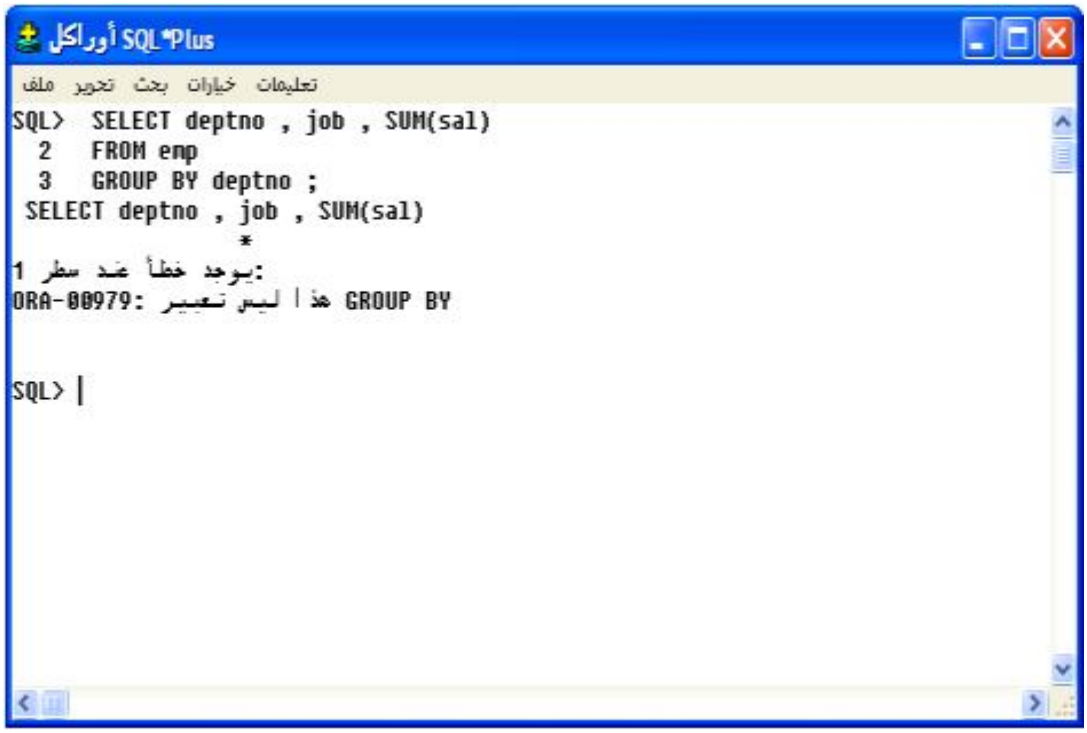


ملاحظة مهمة(1)

أي حقل يظهر مع جملة SELECT تلقائياً لابد من ظهور الحقـل مع Group By . و لا يحدث خطأ في الجملة.

المثال التالي يوضح ذلك:

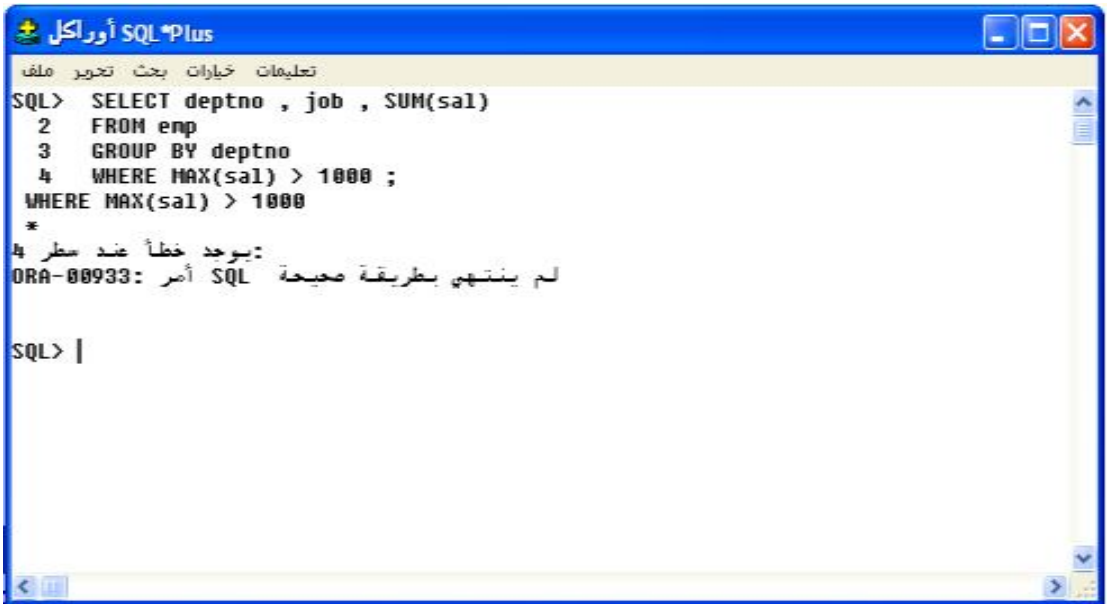
مثال(37)



ملاحظة مهمة(2)

لا يمكن أن نستخدم دوال المجموعات مع عبارة WHERE و المثال التالي يوضح ذلك:

مثال(38)



❐ العبارة HAVING

تستخدم هذه العبارة لعمل شرط على كل مجموعات الحل و تقابل Where و التي تعمل علـــى مستوى السجلات.

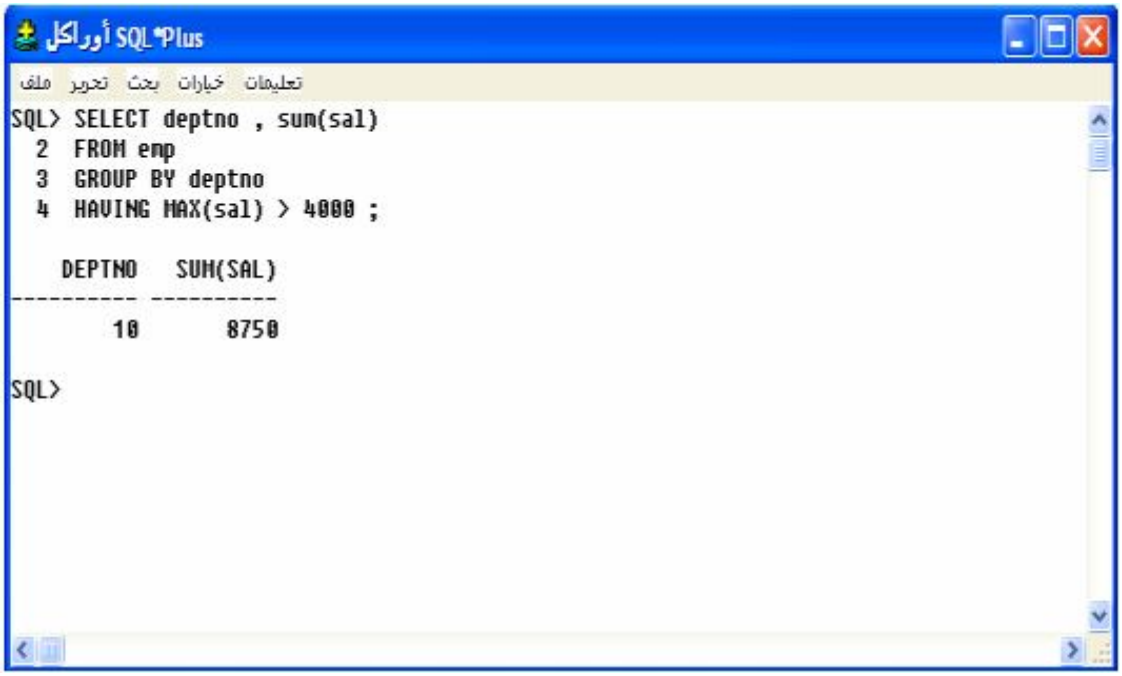
و الصيغة العامة للجملة بعد إضافة HAVING ستكون:

```
SELECT        column, group_function
FROM          table
[WHERE        condition]
[GROUP BY     group_by_expression]
[HAVING       group_condition]
[ORDER BY     column];
```

مثال(39)

اكتب جملة استعلام لعرض مجموع المرتبات في كل قسم على أن يكون أكبر مرتب في كـــل قسم يزيد عن 4000 جنية:

الحل:



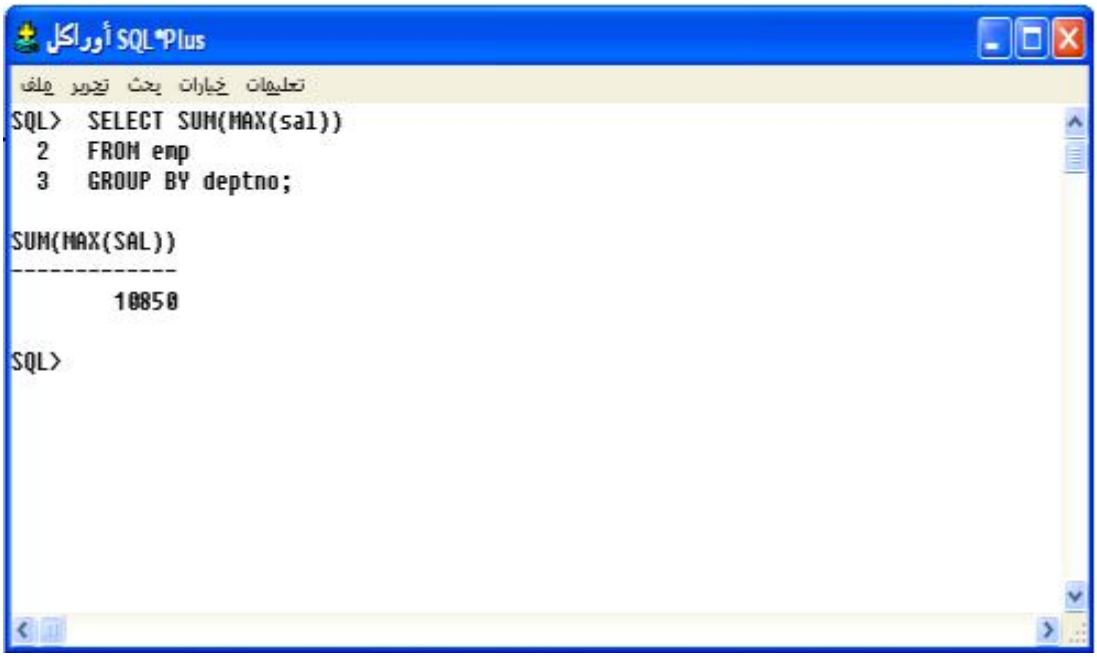
تم استبعاد كل الإدارات عدا الادارة رقم 10 لأنها تحقق الشرط.

يمكن استخدام دوال المجموعات بشكل متداخل على أن تنفذ الدوال الداخلية أولا ثـــم الـــدوال الخارجية.

مثال(40)

اكتب جملة استعلام لاسترجاع مجموع لأكبر المرتبات في كل إدارة:

الحل:



في هذا المثال استخدمنا الدالة MAX داخل الدالة SUM

أسئلة تقويم ذاتي

1/ عدّد الدوال التي يمكن استخدمها على مجموعة من السجلات، مع وصف استخدام كل دالة.

2/ أكتب الصيغة العامة للتعامل مع دوال المجموعات.

3/ وضح استخدام كلٍ من العبارة GROUP BY و العبارة HAVING .

5.1.2 الاستعلامات الفرعية Subqueries

عزيزي الدارس، في هذا الفرع من القسم سوف نتعرف على أنواع المسائل التي يمكن حلها بواسطة الاستعلام الفرعي. و كيفية استخدام الاستعلام الفرعي، و ماهي أنواع الاستعلام الفرعي.

فعلى سبيل المثال لو أردنا أن نسترجع بيانات الموظف الذي يحصل علي أعلى مرتب لابد أولا أن نعرف ما هو أعلى مرتب، ثم بعد ذلك نسترجع بيانات الموظف الذي يتقاضى ذلك المرتب. و لكي ننجز ذلك لابد من وجود استعلامين، الأول لإيجاد أعلى مرتب و الثاني

لاسترجاع بيانات الموظف. و نجد أن الاستعلام الأول يعتمد و بشكل كبير علـــى الاســـتعلام الثاني.

الصيغة العامة للاستعلامات الفرعية هي:

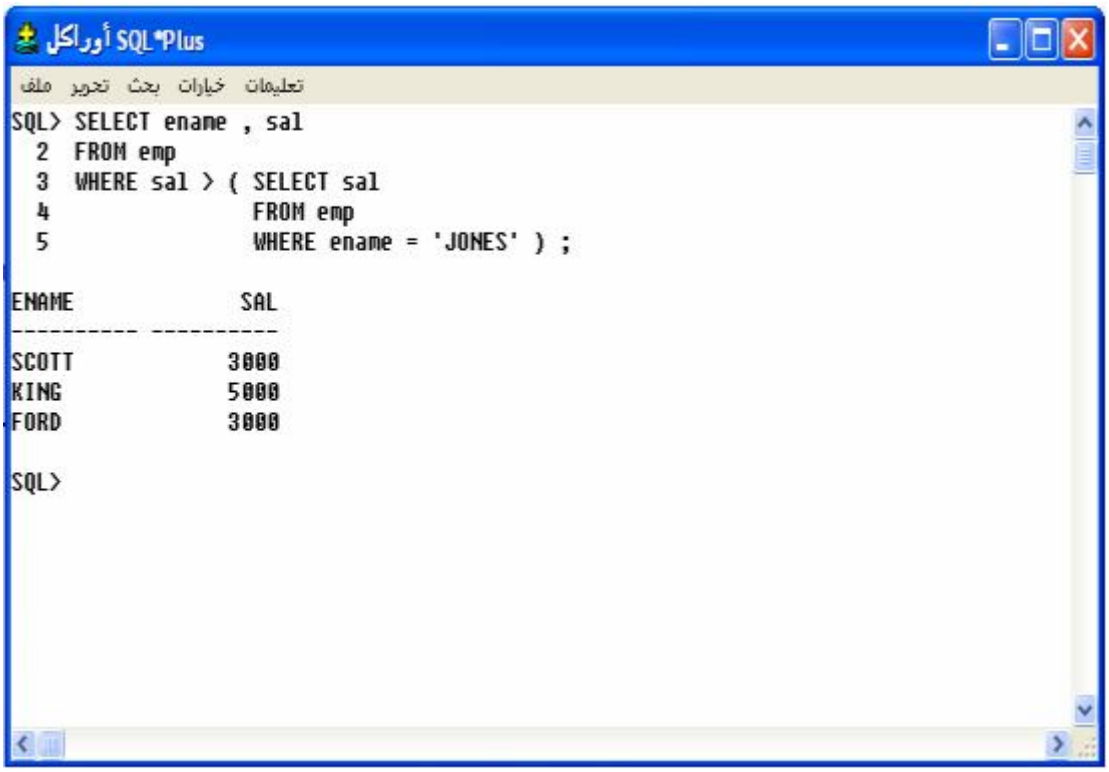
```
SELECT    select_list
FROM      table
WHERE     expr operator
                    (SELECT      select_list
                    FROM         table);
```

الاستعلام الفرعي

مثال(41)

اكتب استعلاماً يقوم باسترجاع أسماء و مرتبات الموظفين الذين يحصلون على نفس المرتـــب الذي يحصل علية الموظف JONES.

الحل:



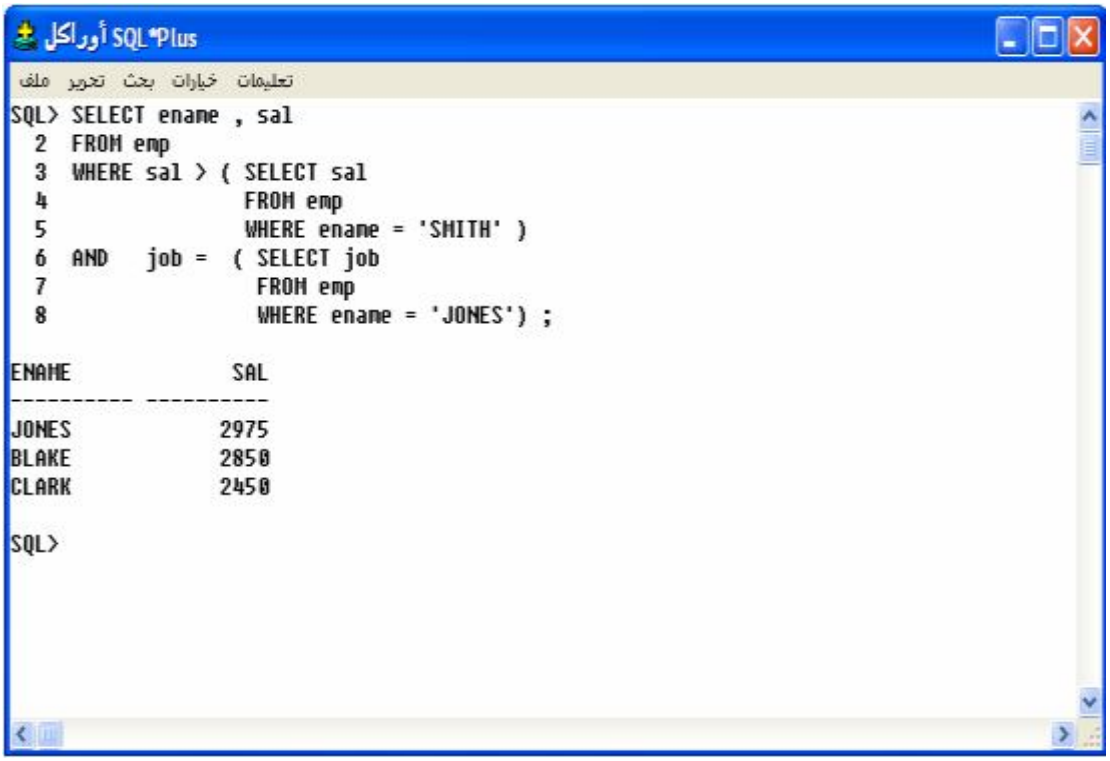
ملاحظة

يمكن أن نستخدم أكثر من استعلام فرعي واحد في الجملة.

مثال(42)

اكتب استعلاماً يقوم باسترجاع أسماء و مرتبات الموظفين الذين يحصلون على نفس المرتـــب الذي يحصل عليه الموظف SMITH و وظيفة JONES.

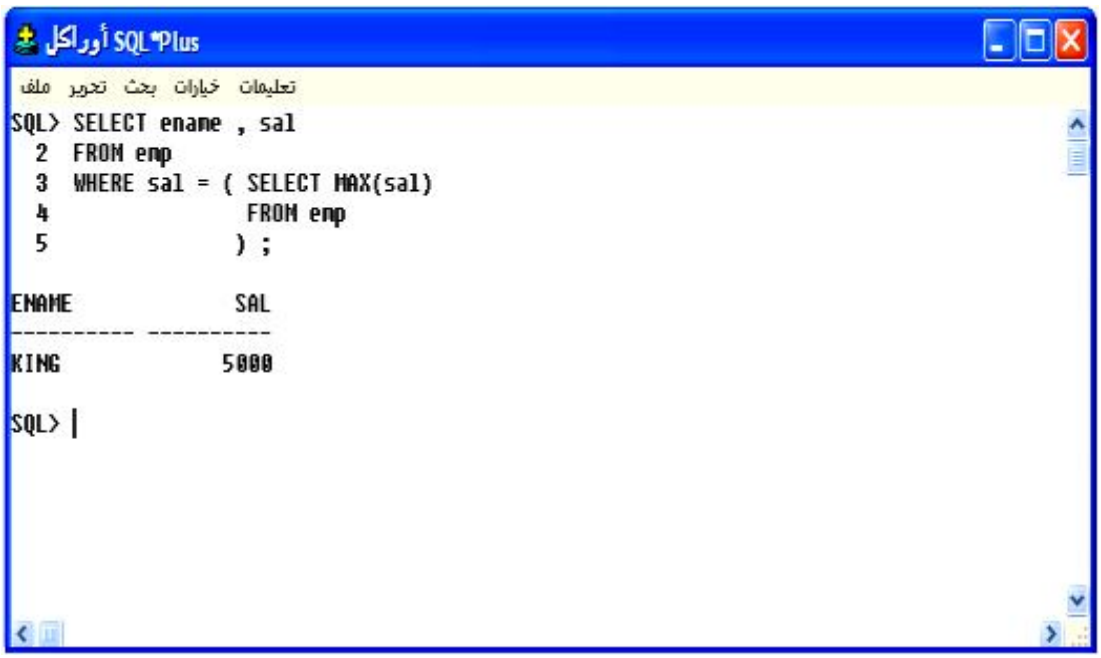
الحل:



مثال(43)

اكتب استعلاماً يقوم باسترجاع أسماء و مرتبات الموظفين الذين يحصلون على أعلى مرتب:

الحل:

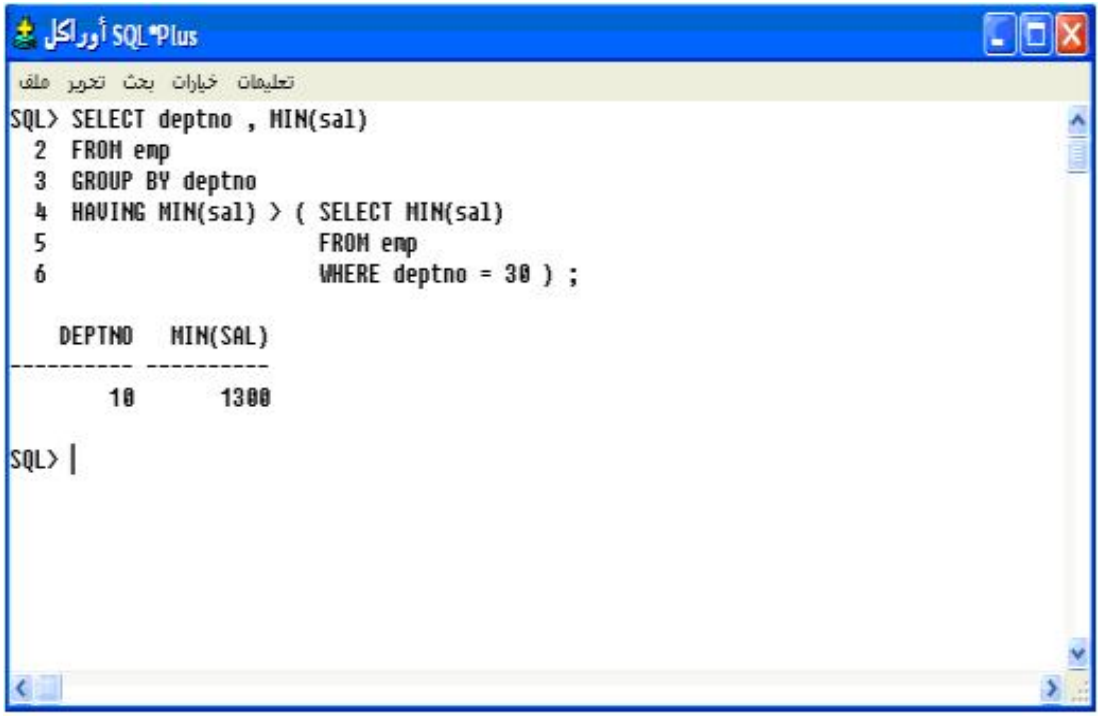


يمكن أن يظهر لنا أكثر من سجل واحد إذا كان هنالك أثر من موظفين يحصلون على أعلى مرتب.

مثال(44)

اكتب استعلاماً يقوم باسترجاع أقل مرتب في كل إدارة. بحيث يكون أقل مرتب أكبر من أقل مرتب في الادارة رقم 10.

الحل:



✳ ملاحظات مهمة

- لا يمكن أن نستخدم Order By في الاستعلام الفرعي .
- يمكن استخدام عمليات المقارنة > و < و == و .... بين الاستعلام الفرعي و الاستعلام الرئيسي، على أن لايسترجع الاستعلام الفرعي قيمة واحدة فقط و إلا فسيحدث خطأ، و المثال (45) التالي يوضح ذلك:

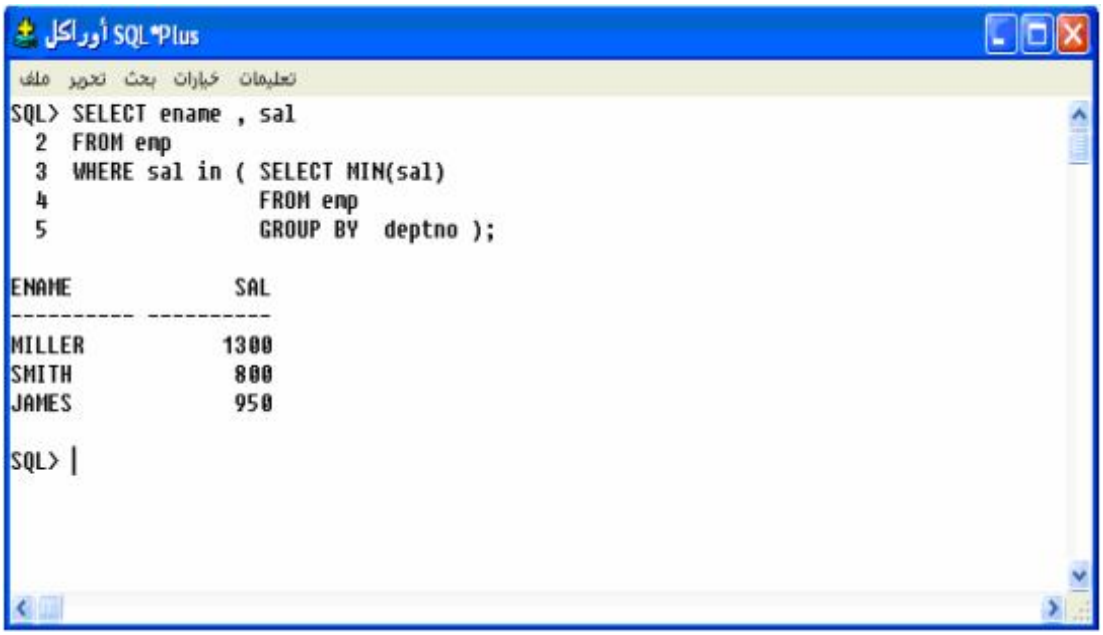
مثال (45)



- يمكن أن أيضا أن نستخدم العمليات التالية بين الاستعلام الفرعي و الرئيسي:

أ- IN و تستخدم لمعرفة أن ناتج الاستعلام الرئيسي موجود داخل القائمة و المثـــال (46) التالي يوضح ذلك.

مثال (46):



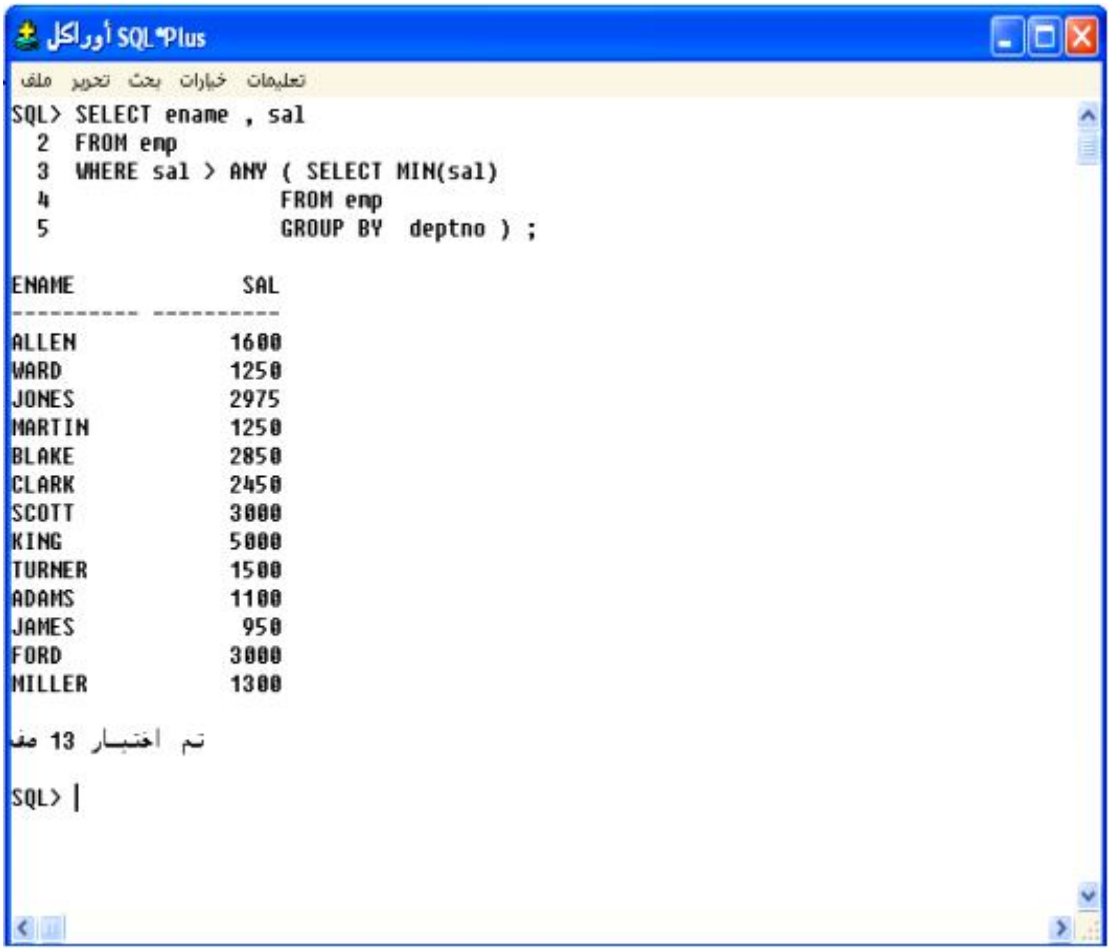
ملاحظة

نلاحظ أن الاستعلام الفرعي يمكن أن ينتج منه أكثر من قيمـــة واحدة.

ب- ANY تستخدم في المقارنة و تكون صحيحة لأي قيمة واحدة محققه الشـــرط، و تستخدم مع عمليات المقارنة.

كما يوضح المثال (47) التالي:

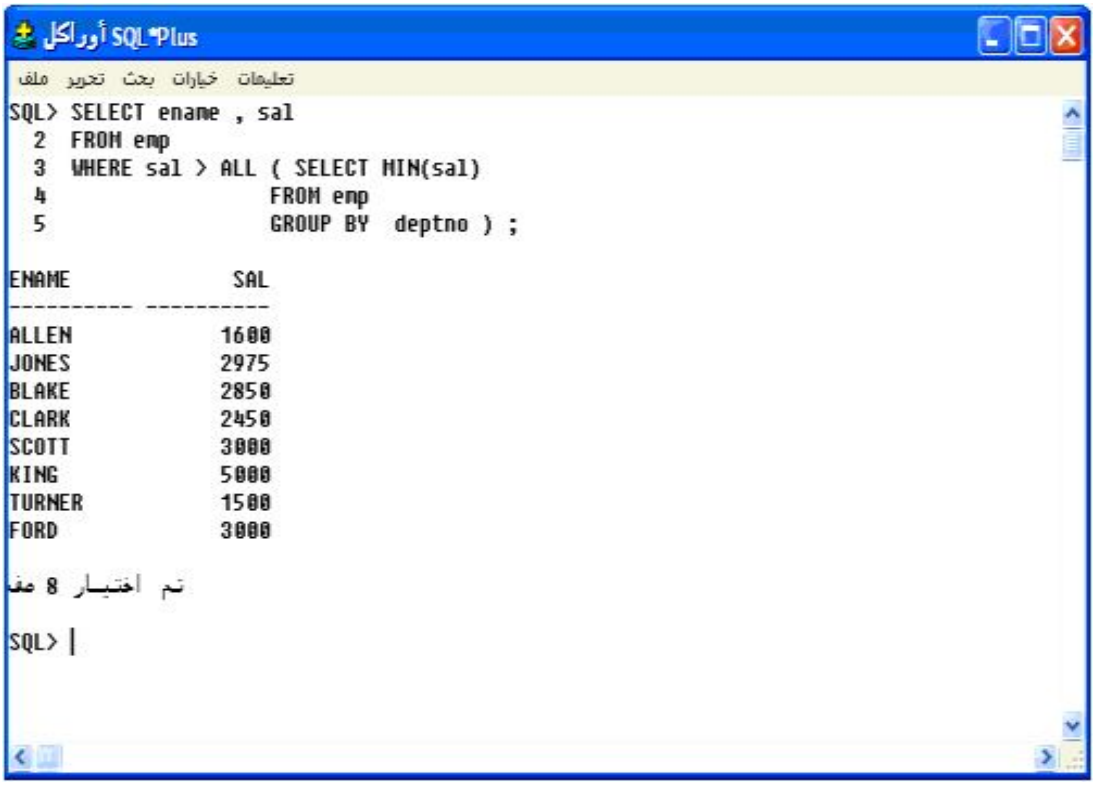
مثال (47):



ت- ALL : في المقارنة و تكون صحيحة لكل القيم محققه الشرط، و تســتخدم مع عمليات المقارنة.

و المثال (48) التالي يوضح ذلك:

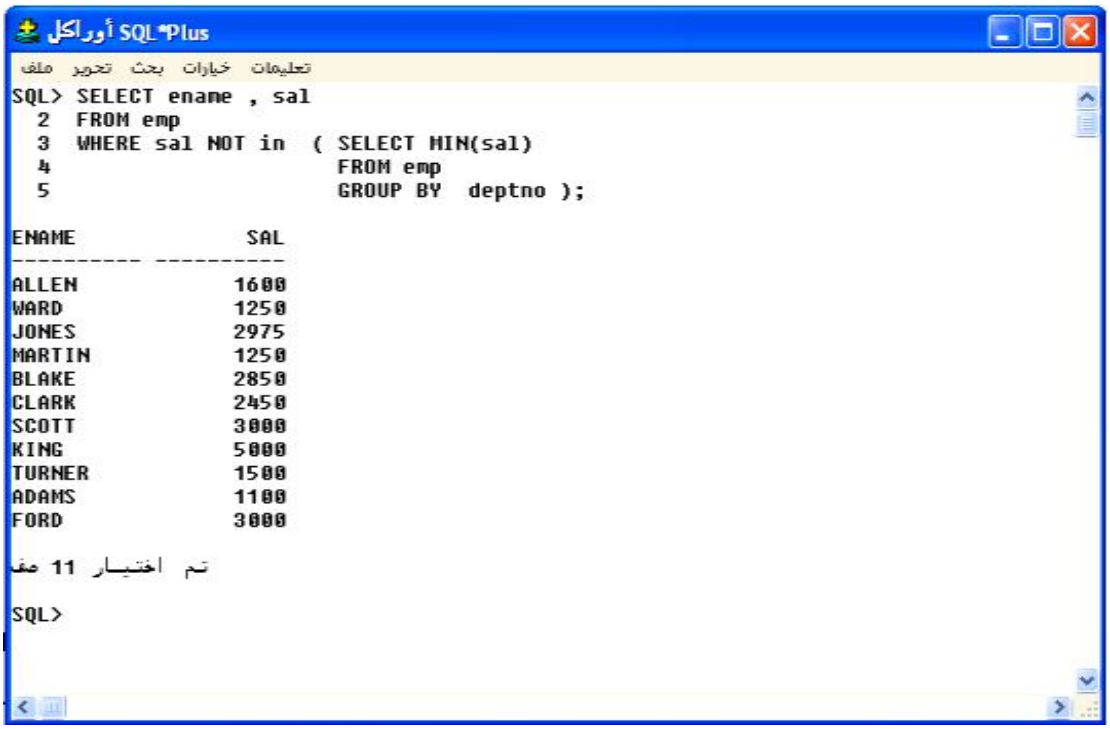
مثال (48)



ملاحظة

يمكن استخدام Not مع ALL , ANY , IN و المثال (49) التالي يوضح ذلك.

مثال (49)



أسئلة تقويم ذاتي

1/ اكتب الصيغة العامة للاستعلامات الفرعية.

2.2 لغة التعامل مع البيانات Data Manipulation Language

عزيزي الدارس، في هذا الفرع من القسم سوف نتعامل مع البيانات من حيث إدخالها أو تعديلها أو حذفها بواسطة جُمل الـSQL.

1.2.2 جملة الإدخال INSERT

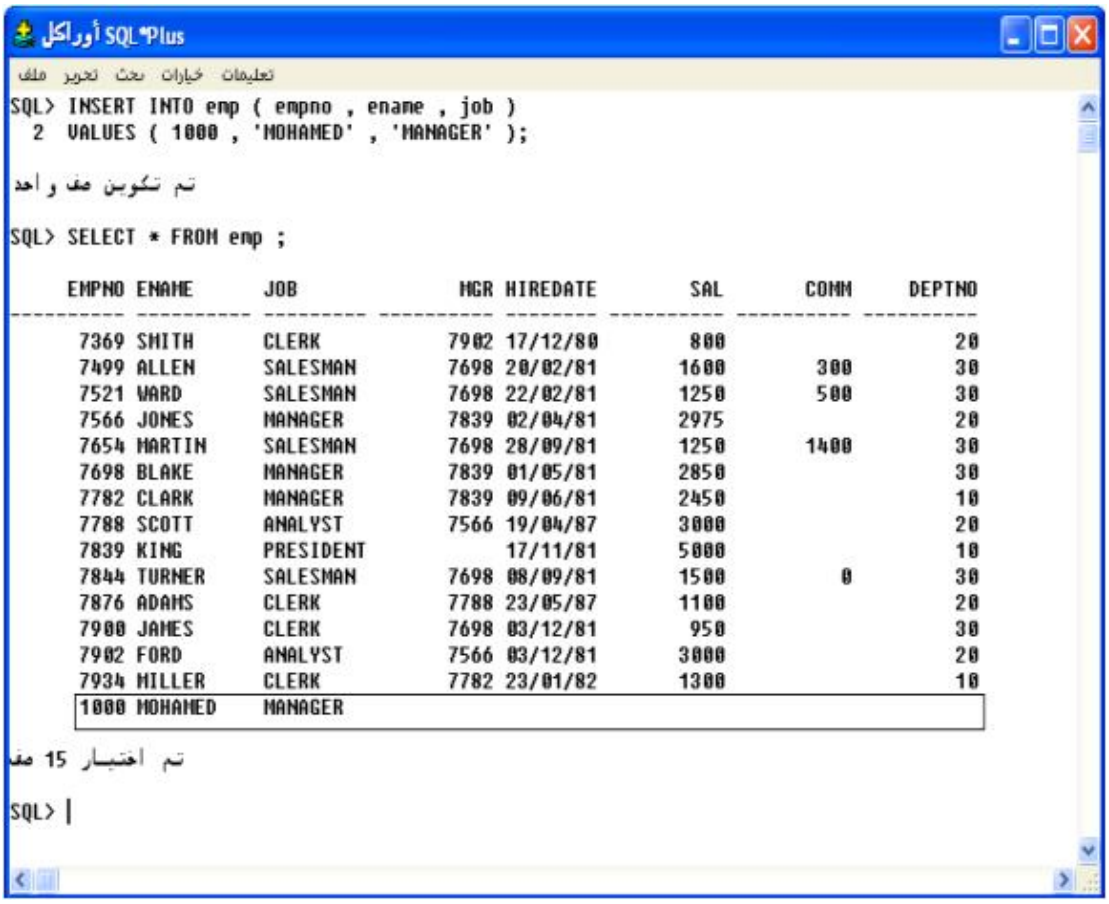
تستخدم هذه الجملة لإدخال السجلات داخل قاعدة البيانات. و الصيغة العامة لهذه الجملة:

```
INSERT INTO   table [(column [, column...])]
VALUES        (value [, value...]);
```

مثال(50)

اكتب جملة SQL لإدخال بيانات الموظف محمد ورقمه الوظيفي 1000 و يعمل كمدير:

الحل:



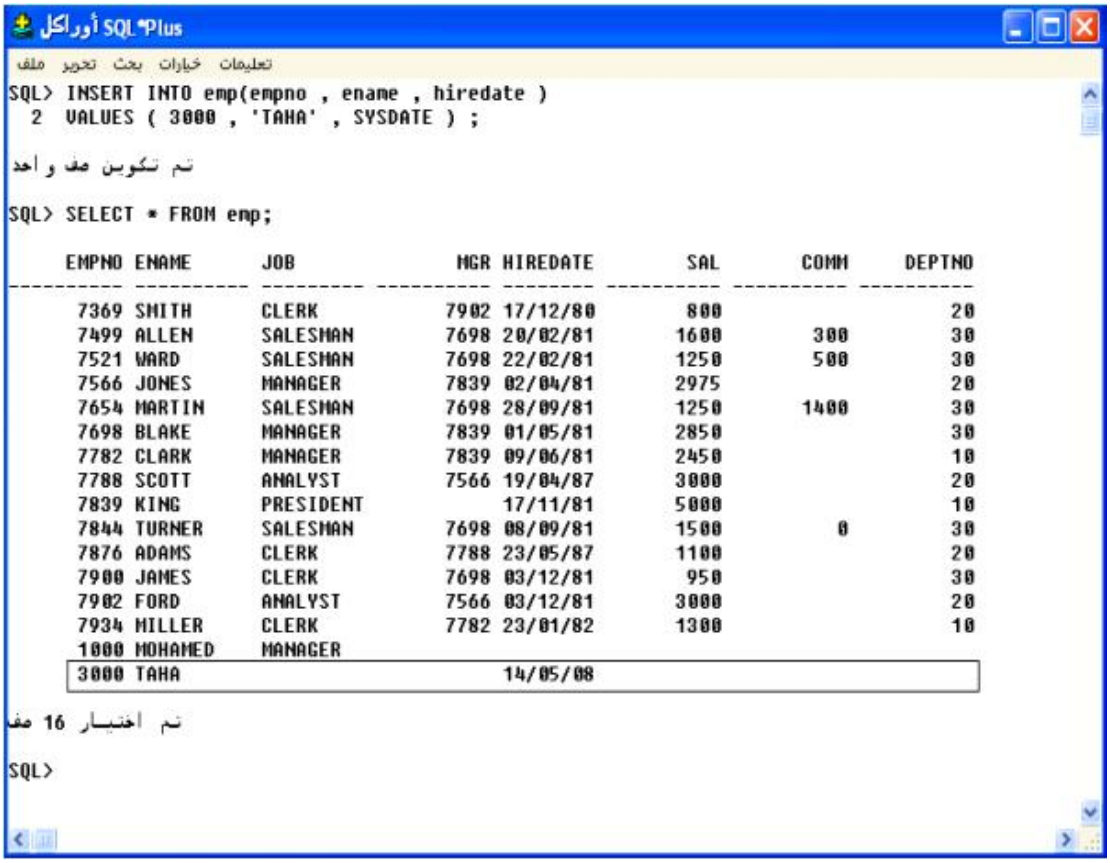
ملاحظة

نلاحظ أن الحقول المدخلة تقابل الحقول التي وردت في الاستعلام.

مثال(51)

اكتب جملة SQL لإدخال بيانات الموظف طه ورقمه الوظيفي 3000 و تاريخ التعيين هو تاريخ الجهاز :

الحل:



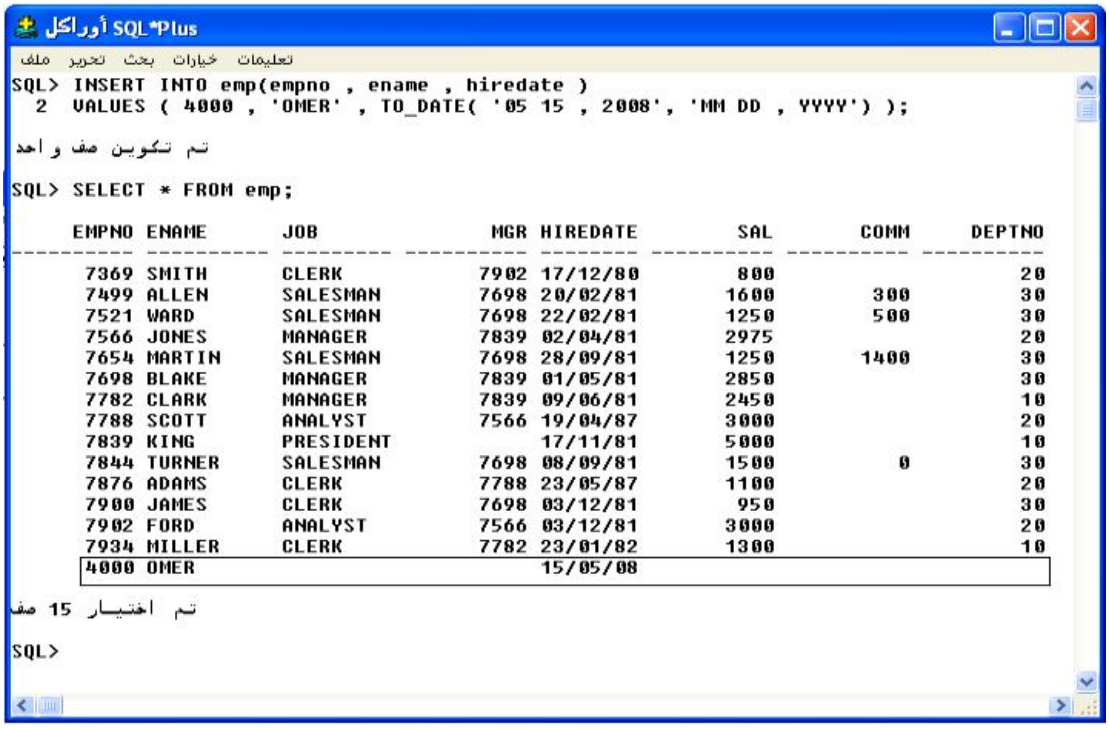
تدريب (10)

اكتب جملة SQL لإدخال بيانات الموظف عمر ورقمه الوظيفي 4000 و تاريخ التعيين هو 2008/05/15 .

مثال(52)

اكتب جملة SQL لإدخال بيانات الموظف عمر ورقمه الوظيفي 4000 و تاريخ التعيين هــو 2008/05/15 :

الحل:



```
أوراكل SQL*Plus
ملف تحرير بحث خيارات تعليمات
SQL> INSERT INTO emp(empno , ename , hiredate )
  2  VALUES ( 4000 , 'OMER' , TO_DATE( '05 15 , 2008', 'MM DD , YYYY') );

تم تكوين صف واحد

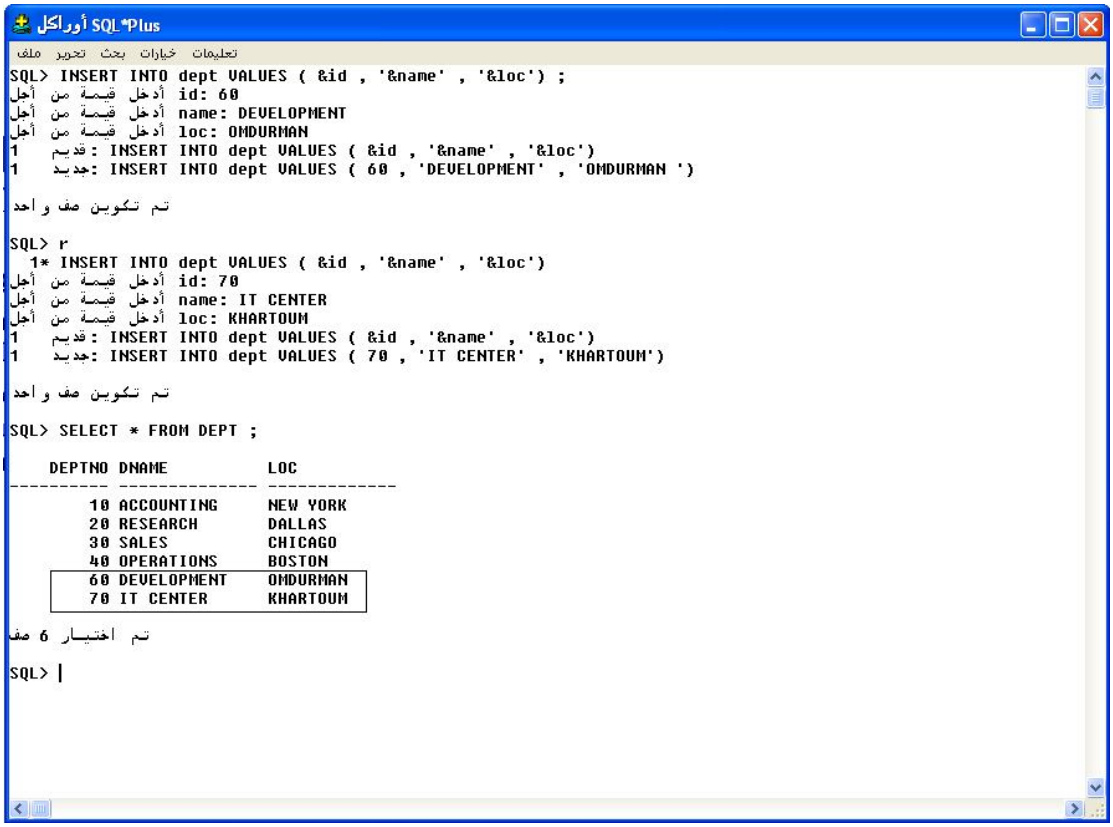
SQL> SELECT * FROM emp;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- -------- ---------- ---------- ----------
      7369 SMITH      CLERK           7902 17/12/80        800                    20
      7499 ALLEN      SALESMAN        7698 20/02/81       1600        300         30
      7521 WARD       SALESMAN        7698 22/02/81       1250        500         30
      7566 JONES      MANAGER         7839 02/04/81       2975                    20
      7654 MARTIN     SALESMAN        7698 28/09/81       1250       1400         30
      7698 BLAKE      MANAGER         7839 01/05/81       2850                    30
      7782 CLARK      MANAGER         7839 09/06/81       2450                    10
      7788 SCOTT      ANALYST         7566 19/04/87       3000                    20
      7839 KING       PRESIDENT            17/11/81       5000                    10
      7844 TURNER     SALESMAN        7698 08/09/81       1500          0         30
      7876 ADAMS      CLERK           7788 23/05/87       1100                    20
      7900 JAMES      CLERK           7698 03/12/81        950                    30
      7902 FORD       ANALYST         7566 03/12/81       3000                    20
      7934 MILLER     CLERK           7782 23/01/82       1300                    10
      4000 OMER                            15/05/08                               

تم اختيار 15 صف

SQL>
```

يمكن أن ندخل عدداً من السجلات بجملة Insert واحدة. و المثال التالي يوضح ذلك.



```
أوراكل SQL*Plus
ملف تحرير بحث خيارات تعليمات
SQL> INSERT INTO dept VALUES ( &id , '&name' , '&loc') ;
أدخل قيمة من أجل id: 60
أدخل قيمة من أجل name: DEVELOPMENT
أدخل قيمة من أجل loc: OMDURMAN
1   قديم: INSERT INTO dept VALUES ( &id , '&name' , '&loc')
1   جديد: INSERT INTO dept VALUES ( 60 , 'DEVELOPMENT' , 'OMDURMAN ')

تم تكوين صف واحد

SQL> r
  1* INSERT INTO dept VALUES ( &id , '&name' , '&loc')
أدخل قيمة من أجل id: 70
أدخل قيمة من أجل name: IT CENTER
أدخل قيمة من أجل loc: KHARTOUM
1   قديم: INSERT INTO dept VALUES ( &id , '&name' , '&loc')
1   جديد: INSERT INTO dept VALUES ( 70 , 'IT CENTER' , 'KHARTOUM')

تم تكوين صف واحد

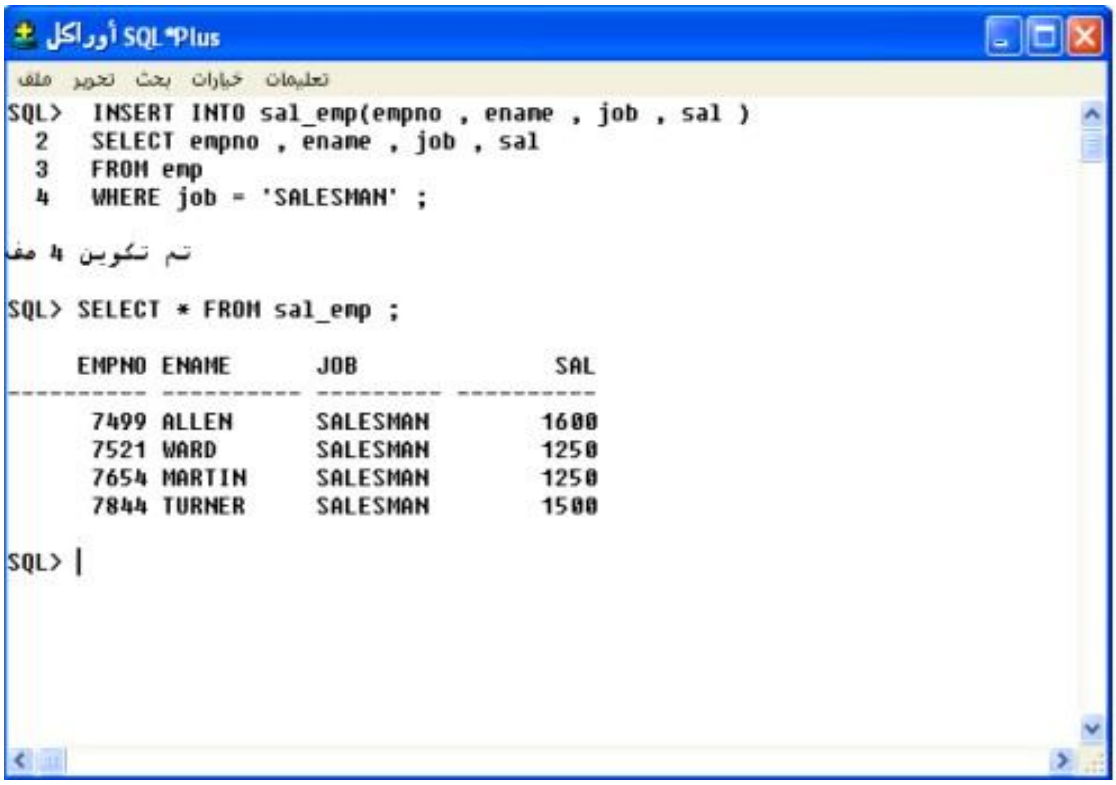
SQL> SELECT * FROM DEPT ;

    DEPTNO DNAME          LOC
---------- -------------- -------------
        10 ACCOUNTING     NEW YORK
        20 RESEARCH       DALLAS
        30 SALES          CHICAGO
        40 OPERATIONS     BOSTON
        60 DEVELOPMENT    OMDURMAN
        70 IT CENTER      KHARTOUM

تم اختيار 6 صف

SQL>
```

يمكن أن ندخل بيانات موجودة في جدول آخر و المثال التالي يوضح قـراءة مجموعـة مـن السجلات موجودة في جدول EMP و سيتم تخزينها في جدول SAL_EM بشرط أن يكـون الموظف يعمل كـ Salesman:



أسئلة تقويم ذاتي



1/ وضح استخدام الجملة INSERT.

2/ اكتب الصيغة العامة لجملة INSERT .

2.2.2 جملة تعديل البيانات UPDATE

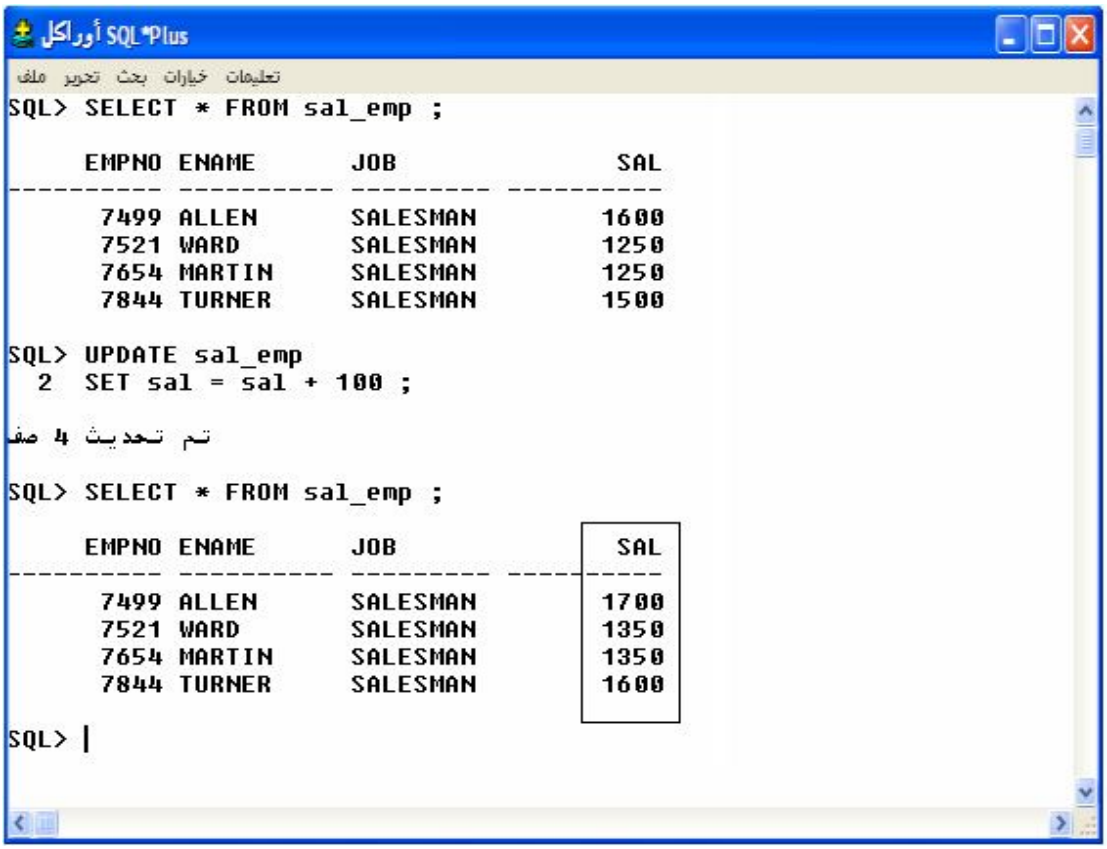
تستخدم هذه الجملة لتعديل بيانات السجلات على أن يتم التعـديل علـى مسـتوي الحقـول و الصيغة العامة لهذه الجملة:

```
UPDATE      table
SET         column = value [, column = value, ...]
[WHERE      condition];
```

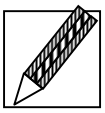
مثال(53):

اكتبجملة SQL لزيادة رواتب الموظفين في جدول Sal _EMP بحيث يزيد مرتب كل واحد عن 100 جنيه :

الحل:



تدريب (10)

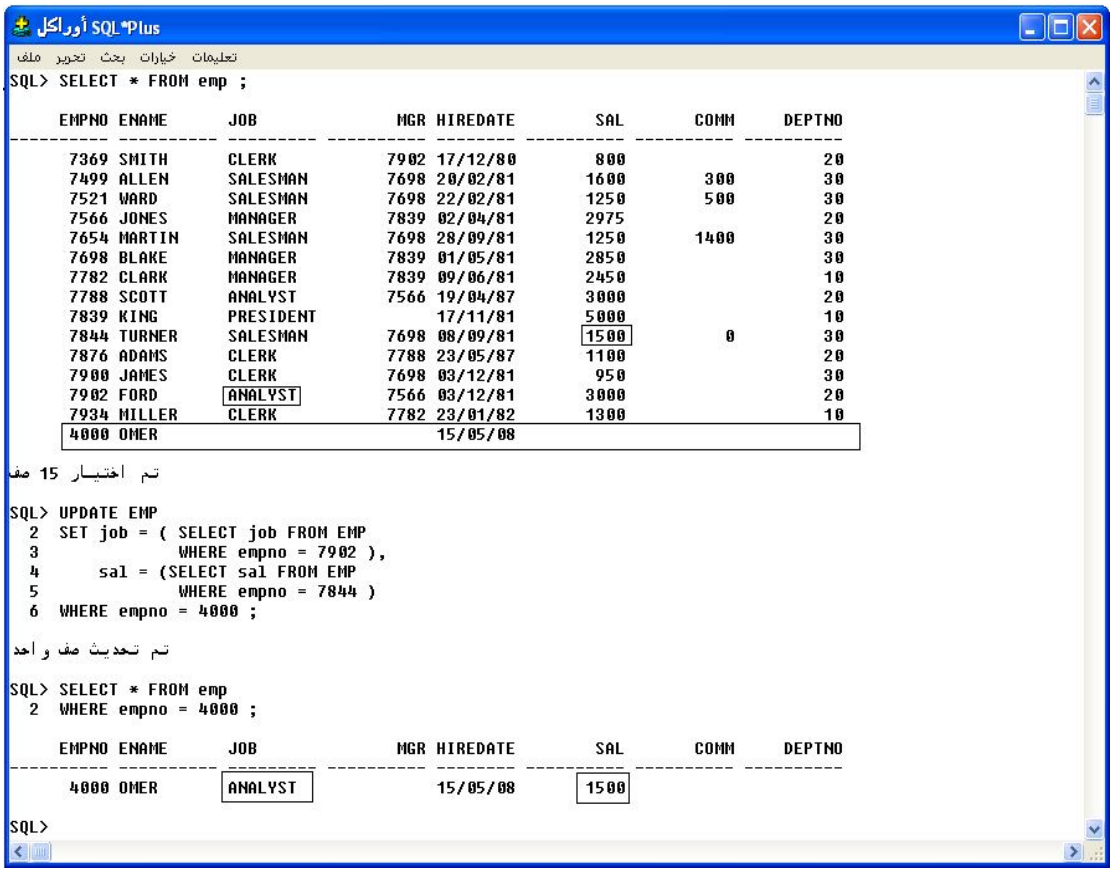


اكتب جملة SQL لزيادة رواتب الموظفين في جـــدول EMP بحيـــث يزيـــد مرتب كل واحد عن 50 جنيهاً بشرط أن تكون رواتبهم تزيد عن 1500جنيـــه

مثال(54)

اكتب جملة SQL لإضافة بيانات غير موجوده للموظف رقم 4000 بحيث تصبح وظيفتـــه مثل وظيفة الموظف رقم 7902 و مرتبه مثل مرتب الموظف رقم 7844.

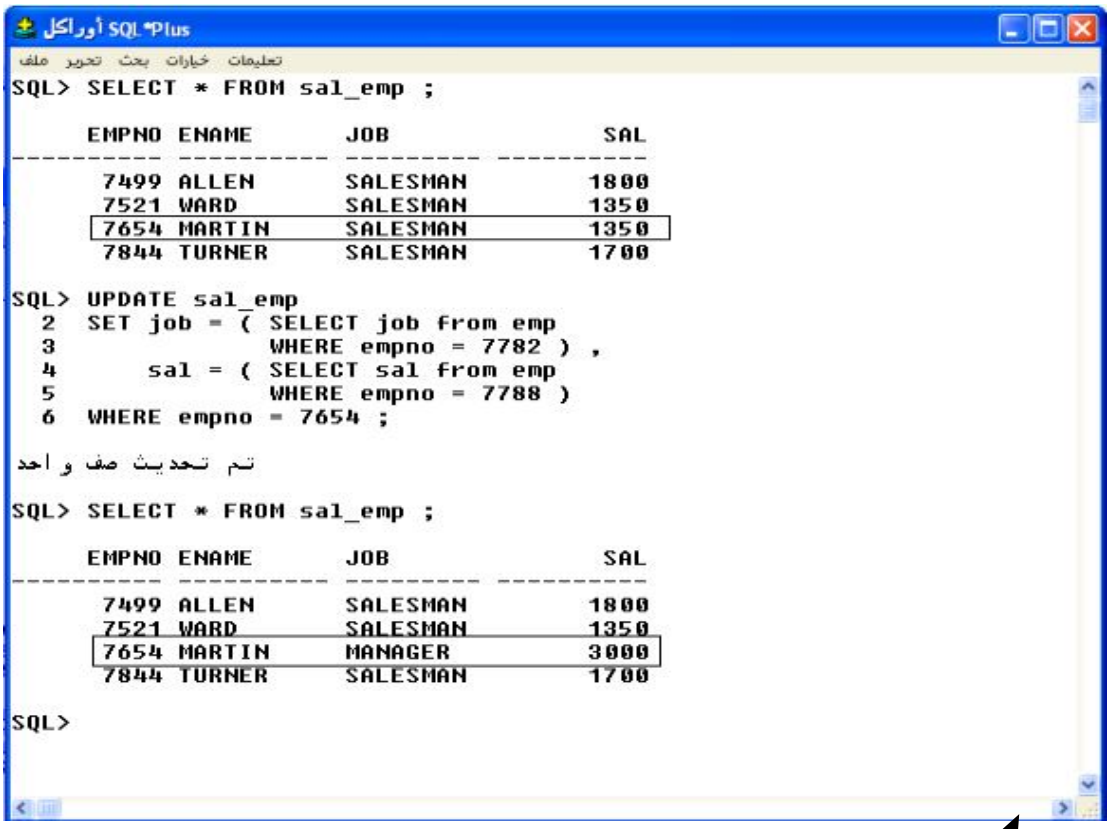
الحل:



مثال(55):

اكتب جملة SQL لتعديل بيانات الموظف رقم 7654 ، بحيث تصبح وظيفته مثل وظيفة الموظف رقم 7782 ، و مرتبه مثل مرتب الموظف رقم 7788

الحل:



ملاحظة

إذا كانت هنالك علاقة مرجعيـــة بواســـطة Foreign Key بـــين جدولين في هذا القيد يتم التحقق منه قبل تعديل الســـجل و المثـــال التالي يوضح ذلك.

مثال(56)

اكتب جملة SQL لتعديل رقم إدارة الموظف رقم 7369 لكي تصبح 80 :

الحل:



نلاحظ حدوث خطأ، وذلك لأنه لا توجد ادارة رقم 80 في جـــدول الإدارات، و تـــم رفـــض التعديل.

أسئلة تقويم ذاتي

1/ وضح استخدام الجملة UPDATE.

2/ أكتب الصيغة العامة لجملة UPDATE .

3.2.2 جملة الحذف DELETE

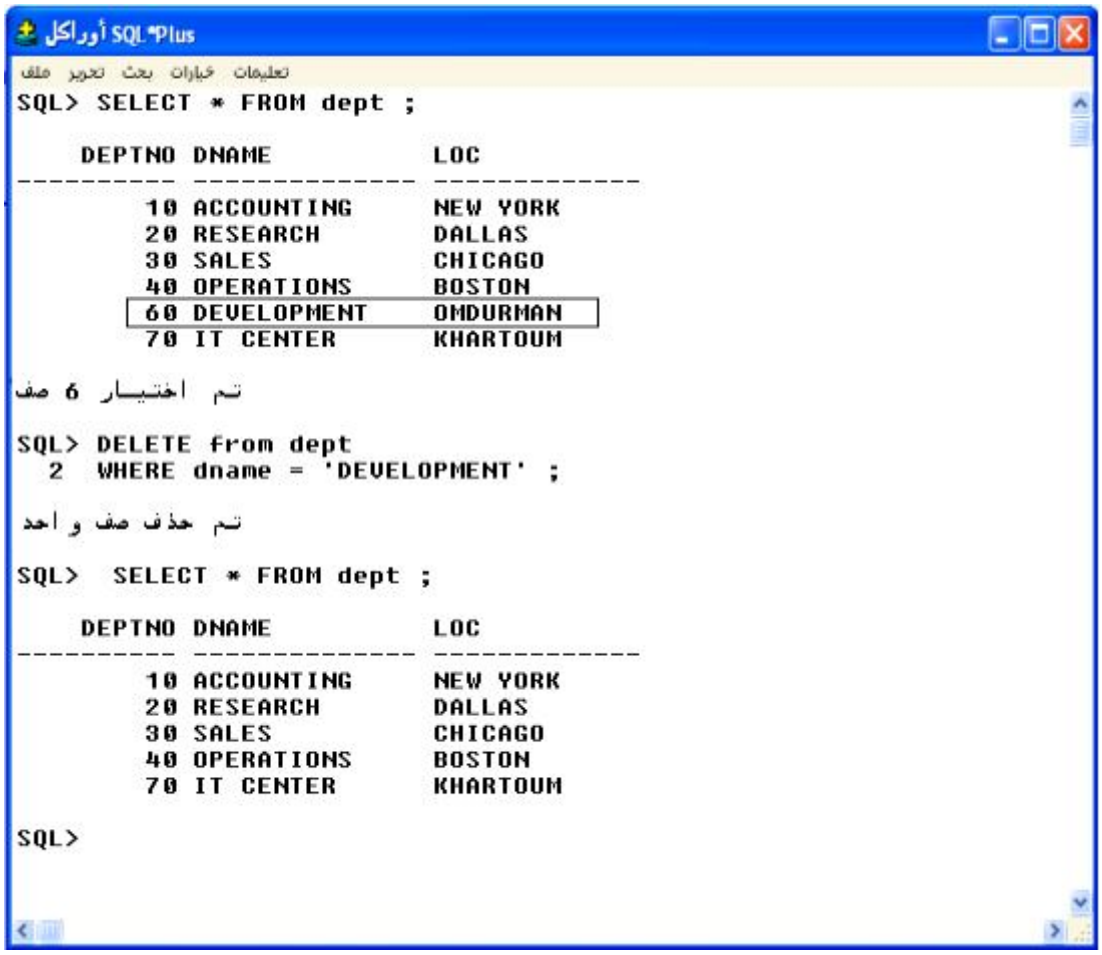
تستخدم جملة DELETE لحذف السجلات و الصيغة العامة هي:

```
DELETE [FROM]   table
[WHERE          condition];
```

مثال(57)

اكتب جملة SQL لحذف بيانات الإدارة DEVELOPMENT

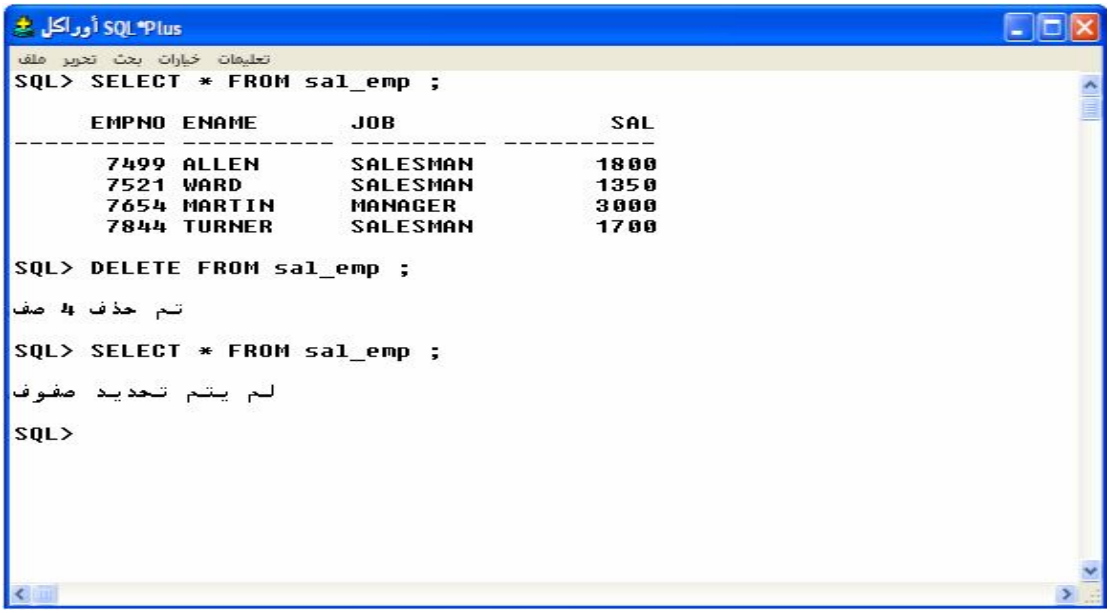
الحل:



مثال(58)

اكتب جملة SQL لحذف بيانات الجدول SAL_EMP

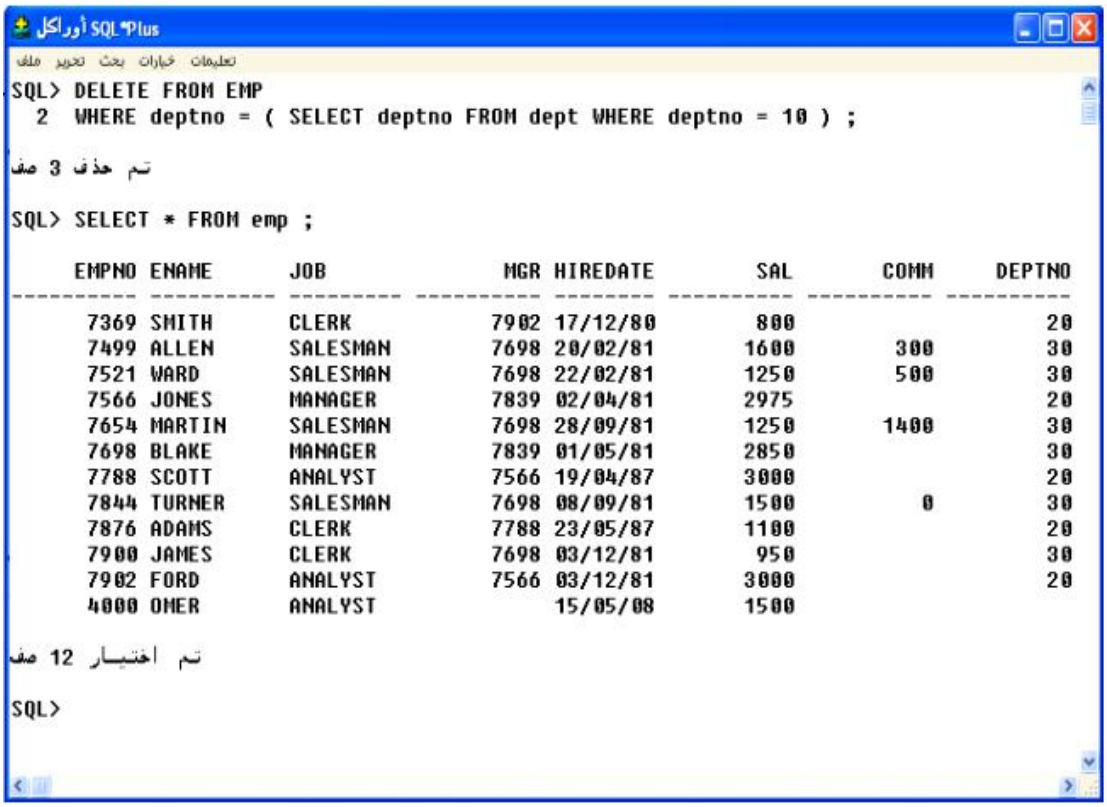
الحل:



مثال(59)

اكتب جملة SQL لحذف بيانات الموظفين الذين يعملون في الادارة رقم 10 .

الحل:



ملاحظة

إذا كانت هنالك علاقة مرجعية بواسطة Foreign Key بين جدولين في هذا القيد يتم التحقق منه قبل حـــذف الســـجل و المثال(60) التالي يوضح ذلك.

مثال(60)

اكتبجملة SQL لحذف بيانات الإدارة رقم 10

الحل:



ملاحظة

نلاحظ أنه تم رفض الحذف لوجود سجلات مرتبطة مع الإدارة رقم 10 .

أسئلة تقويم ذاتي

1/ وضح استخدام الجملة DELETE.

2/ اكتب الصيغة العامة لجملة DELETE .

3.2 إدارة و إنشاء الجداول Creating and Managing Tables

عزيزي الدارس، في هذا الفرع من القسم سوف نتناول كيفية إنشاء الجداول و أنواع البيانـــات التي يمكن استخدامها و أيضا نتناول عملية تعديل الجدول و كذلك حذف الجدول.

1.3.2 إنشاء الجدول Creating Table

لإنشاء جدول لابد من تحديد اسم الجدول و كذلك الحقول التي ســوف يحتويهــا و نســتخدم الصيغة التالية لعمل ذلك:

```
CREATE TABLE [schema.]table
            (column datatype [DEFAULT expr][, ...]);
```

حيث إن :

Schema توضح المستخدم الذي يمتلك الجدول و هو اختياري.

Table اسم الجدول و يجب أن يبدأ بحرف و لايسمح بتكرار اسم الجدول.

Column اسم الحقل.

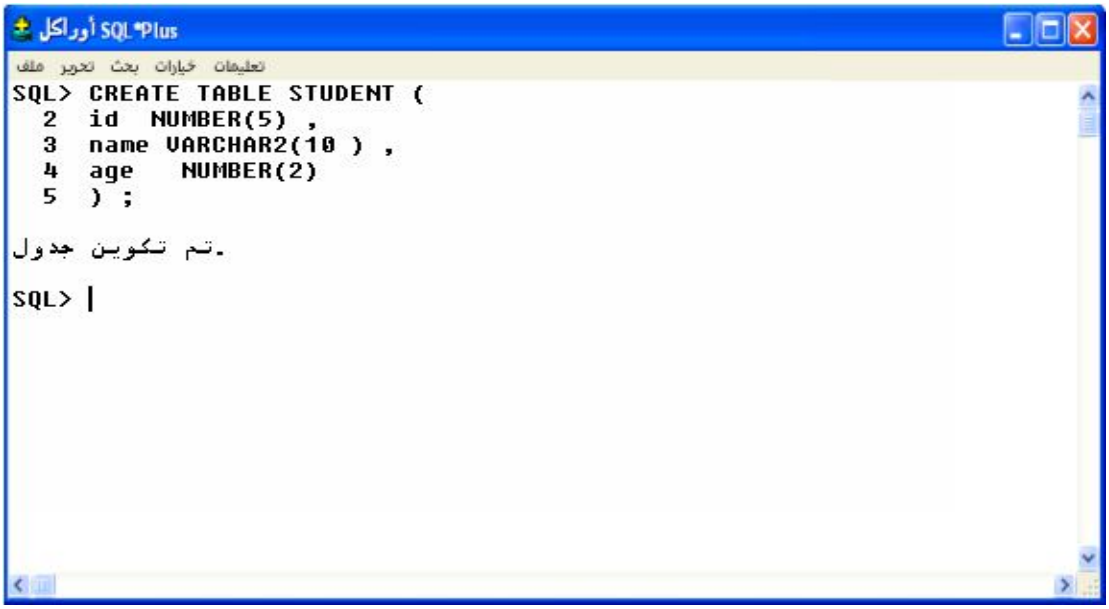
Data Type نوع الحقل.

Default تستخدم بصورة اختيارية لتحديد قيمة افتراضية للحقل عند إنشاء السجلات.

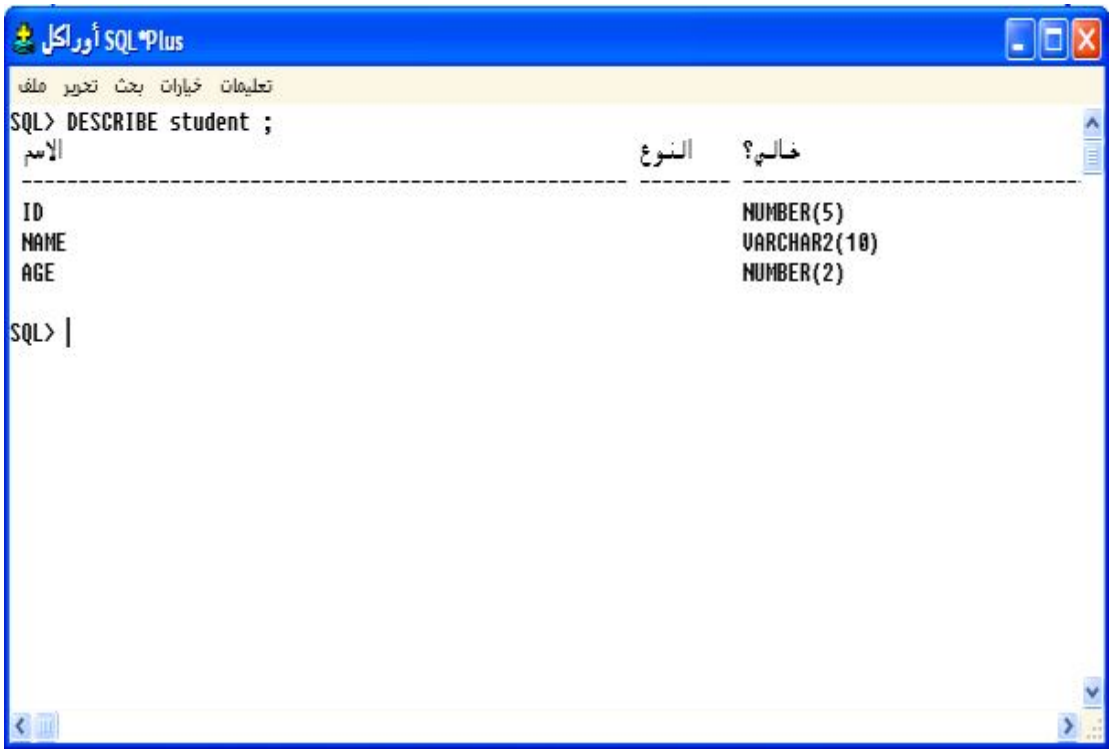
مثال(61)

اكتب جملة SQL لإنشاء جدول للطلاب يحتوي على رقم الطالب الأكاديمي واســم الطالــب والعمر.

الحل:



❐ **لعرض شكل الجدول**



❐ **أنواع بيانات الحقول Column Data Type**

الجدول التالي يوضح بعضاً من أنواع بيانات الحقول:

أنواع البيانات	وصف أنواع البيانات
VARCHAR2 (*size*)	البيانات الحرفية ذات الطول المتغير
CHAR[(*size*)]	البيانات الحرفية ذات الطول الثابت
NUMBER[(*p,s*)]	بيانات عددية ذات طول متغير
DATE	بيانات الوقت و التاريخ
LONG	بيانات حرفية تصل إلى 2 جيجا بايت

❐ **إنشاء الجدول بواسطة استعلام**

يمكن أن ننشئ جدولاً من جدول آخر باستخدم استعلام. وعملية الإنشاء تتضمن نقل البيانـــات كذلك و يكون شكل الأمر كما يلي :

```
CREATE TABLE table
        [(column, column...)]
AS subquery;
```

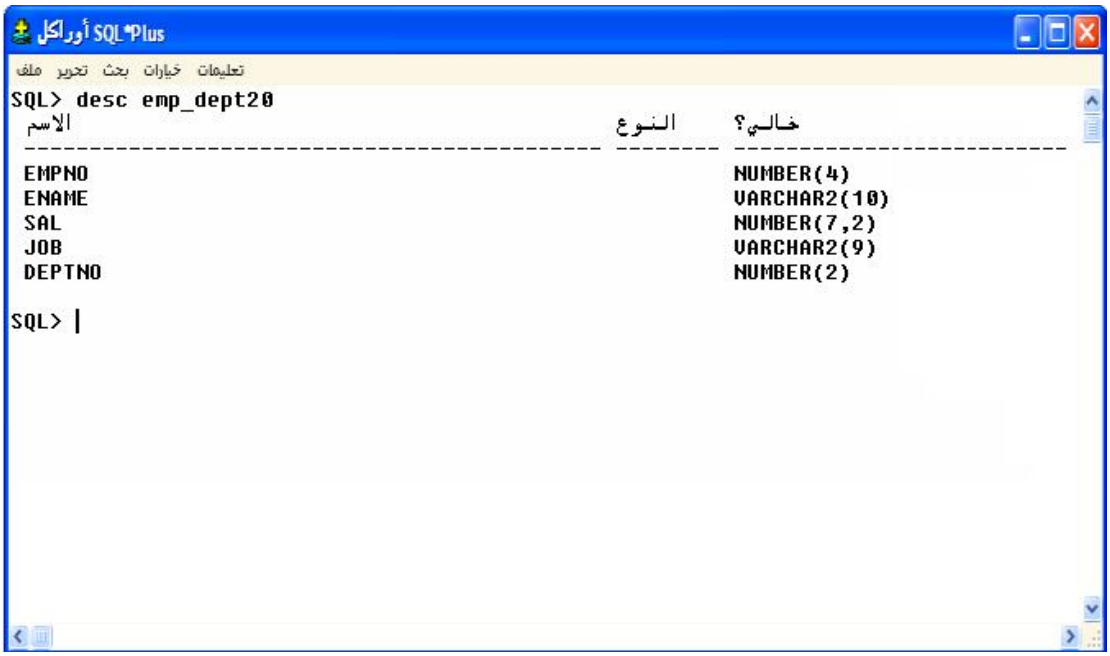
مثال(62)

اكتب جملة SQL تقوم بإنشاء جدول للموظفين الذين يعملون في الإدارة رقم 20:

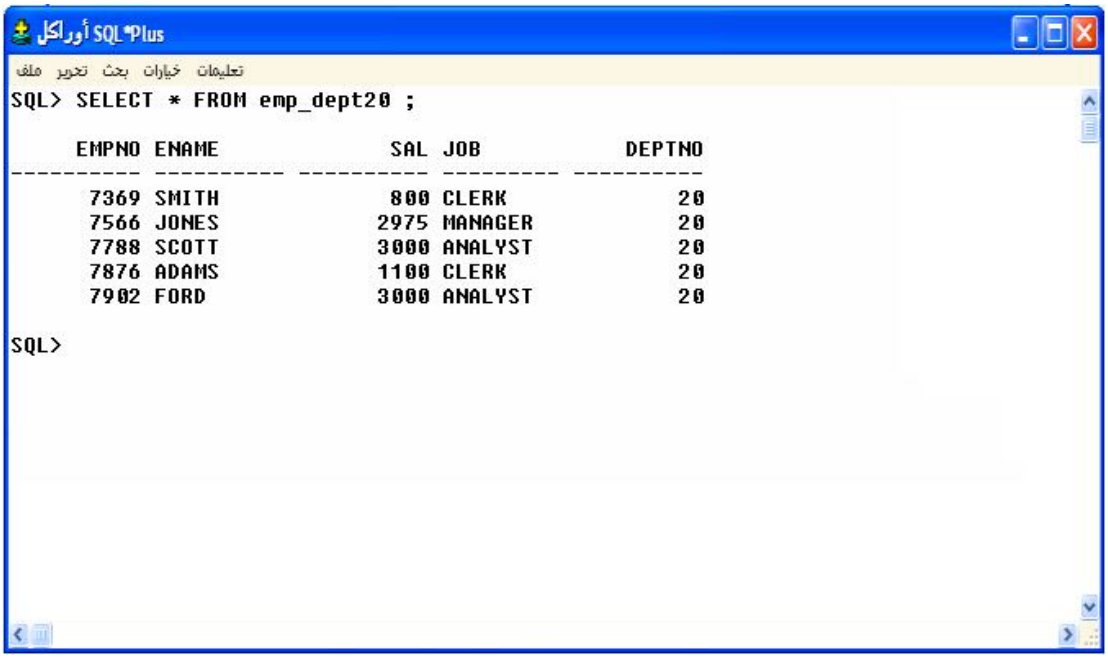
الحل:



❐ **لعرض شكل الجدول**



❐ لعرض محتوى الجدول من بيانات



2.3.2 تعديل الجدول Altering Table

يمكن أن يتم تعديل الجدول بعد الإنشاء و تشمل عملية التعديل العمليات التالية:

أ- إضافة حقول جديدة.

ب- تعديل مواصفات حقل موجود.

ت- حذف حقل.

و الصيغ التالية تستخدم في عملية التعديل:

```
ALTER TABLE table
ADD         (column datatype [DEFAULT expr]
            [, column datatype]...);
```

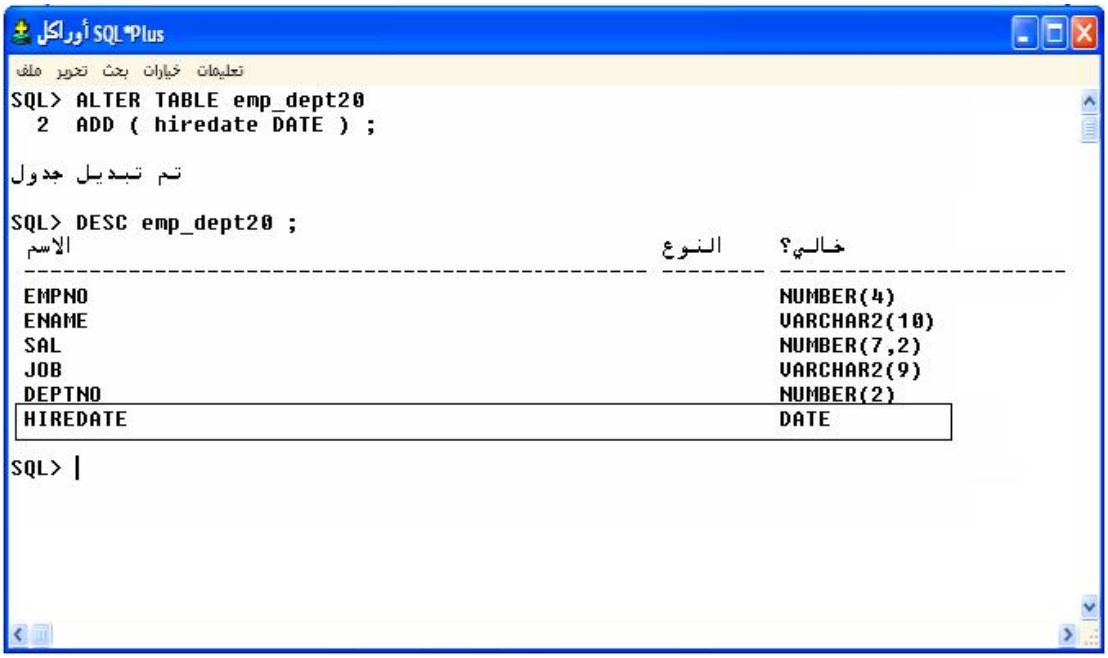
```
ALTER TABLE table
MODIFY      (column datatype [DEFAULT expr]
            [, column datatype]...);
```

```
ALTER TABLE table
DROP        (column);
```

مثال(63):

اكتبجملة SQL تقوم بإضافة حقل باسم hiredate من نوع Date في جدول emp_dept20

الحل:



تدريب (11)

اكتب جملة SQL تقوم بتعديل حقل Ename ليصبح طوله 20 بدلا من 10 .

تدريب (12)

اكتب جملة SQL تقوم بحذف hiredate من جدول emp_dept20 .

أسئلة تقويم ذاتي

1/ لإنشاء جدول لابد من تحديد اسم الجدول، و كذلك الحقول التي سوف يحتويها ، أكتب الصيغة العامة لعمل ذلك.

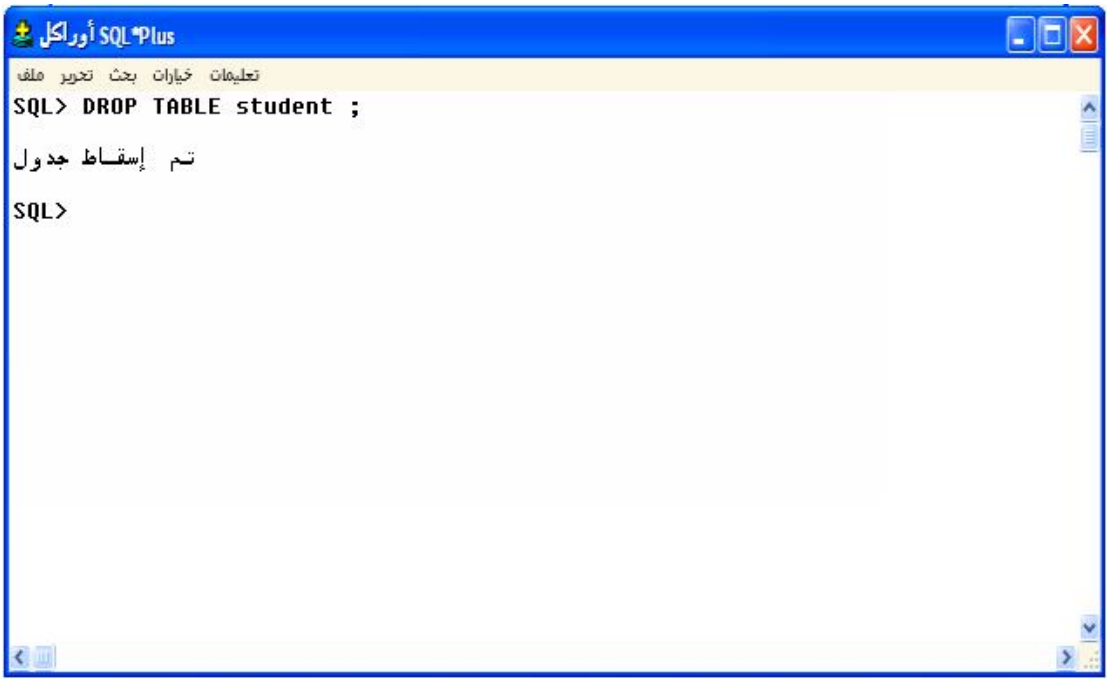
2/ عدد بعض أنواع البيانات.

3/ يمكن أن ننشئ جدول من جدول آخر باستخدم استعلام و عملية الإنشاء تتضمن نقل البيانات كذلك وضح كيف يكون شكل الأمر.

4/ تشتمل عملية تعديل الجدول بعد الإنشاء على عدد من العمليات اذكرها، مع توضيح الصيَّغ التي تستخدم في عملية التعديل.

3.3.2 حذف الجدول Drop Table

يمكن أن نحذف الجدول باستخدام الأمر Drop كما هو موضح في الشكل التالي:



3. قيود السلامة Integrity Constraints

عزيزي الدارس، تستخدم قيود السلامة على مستوى الجداول للتحقق من سلامة العمليات التي تجري على الجداول و هنالك عدة أنواع من قيود السلامة متمثلة في التالي :

أ- NOT NULL: وتعني أن يكون الحقل غير فارغ.

ب- UNIQUE: و تعني أن يكون الحقل أو مجموعة من الحقول غير مكررة.

ت- Primary Key : لعمل مفتاح أساسي من حقل أو مجموعة من الحقول.

ث- Foreign Key : لعمل المفتاح الأجنبي.

ج- CHECK للتحقق من أن قيمة حقل محققة معينة.

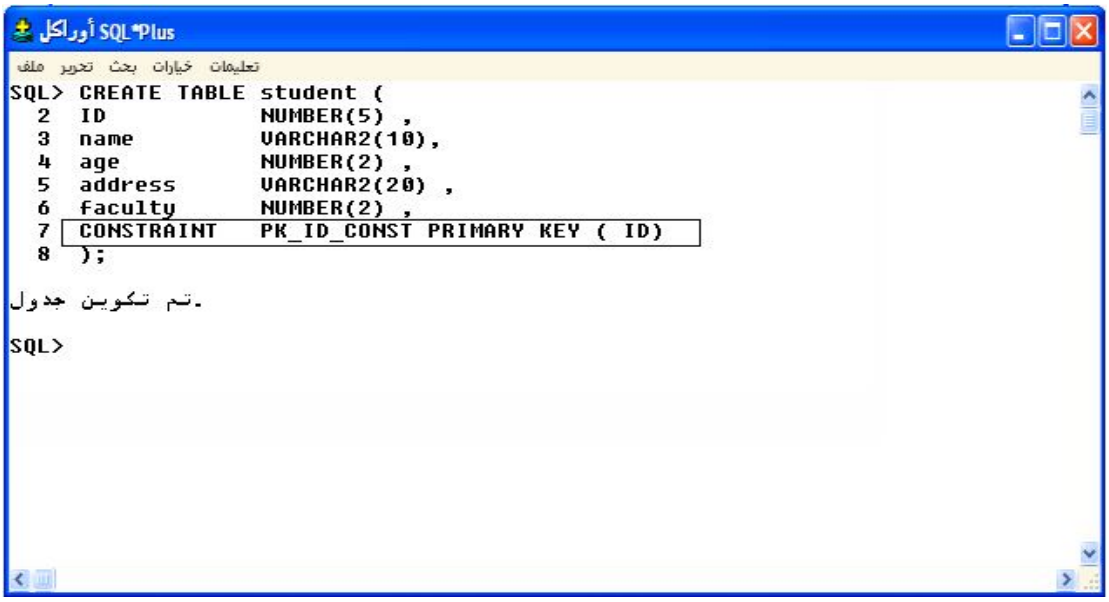
و لعمل قيد عند إنشاء الجدول تكون صيغة الجملة كما يلي:

```
CREATE TABLE [schema.]table
            (column datatype [DEFAULT expr]
            [column_constraint],
            ...
            [table_constraint][,...]);
```

مثال(64)

اكتب جملة SQL تقوم بإنشاء جدول للطلاب يحتوي على رقم الطالب واسم الطالب و عمـــره و العنوان و الكلية، على أن يكون رقم الطالب مفتاحاً اساسياً.

الحل:



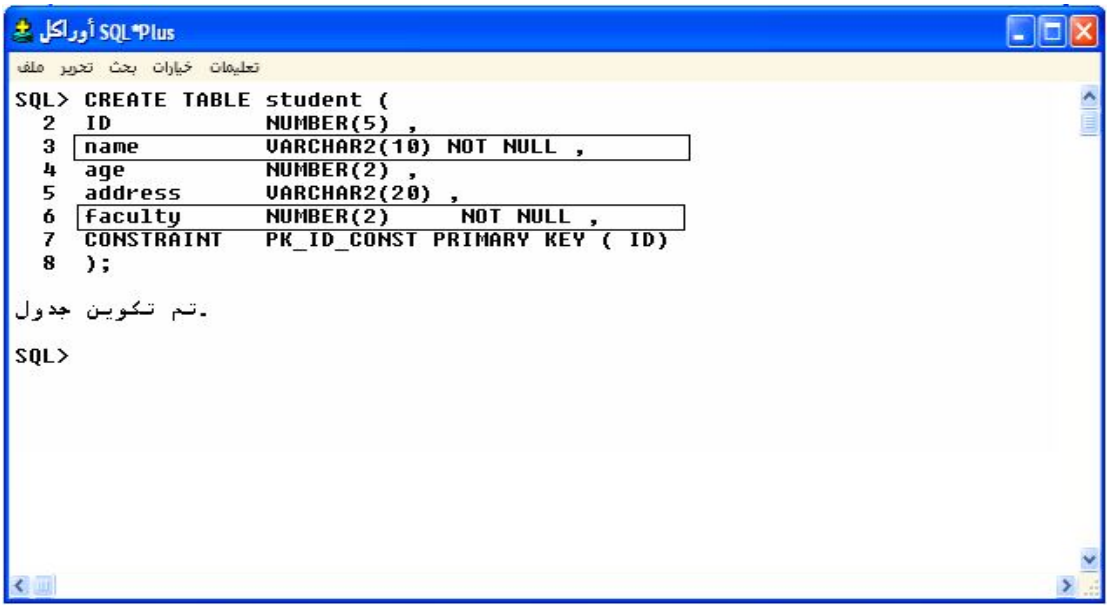
يمكن أن نعرف القيد على مستوى الحقول. و تكون الصيغة العامة في هذه الحالة:

```
column [CONSTRAINT constraint_name] constraint_type,
```

أو

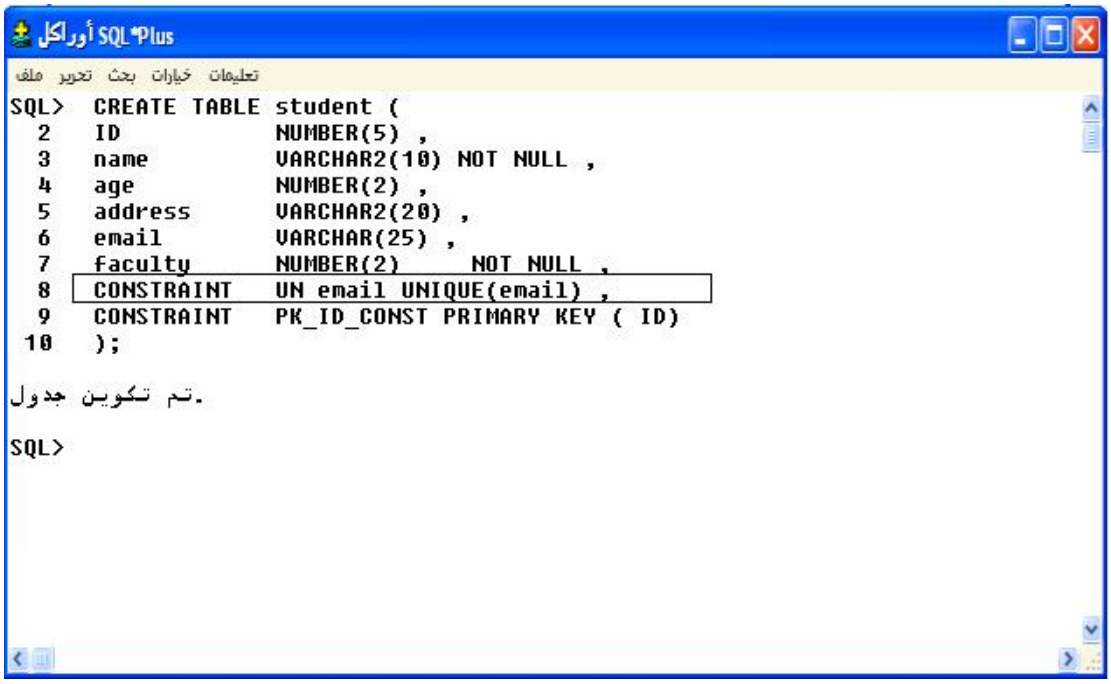
```
column,...
  [CONSTRAINT constraint_name] constraint_type
  (column, ...),
```

و المثال التالي يوضح ذلك:



في هذا المثال تم عمل قيدين على مستوى الحقول name و Faculty

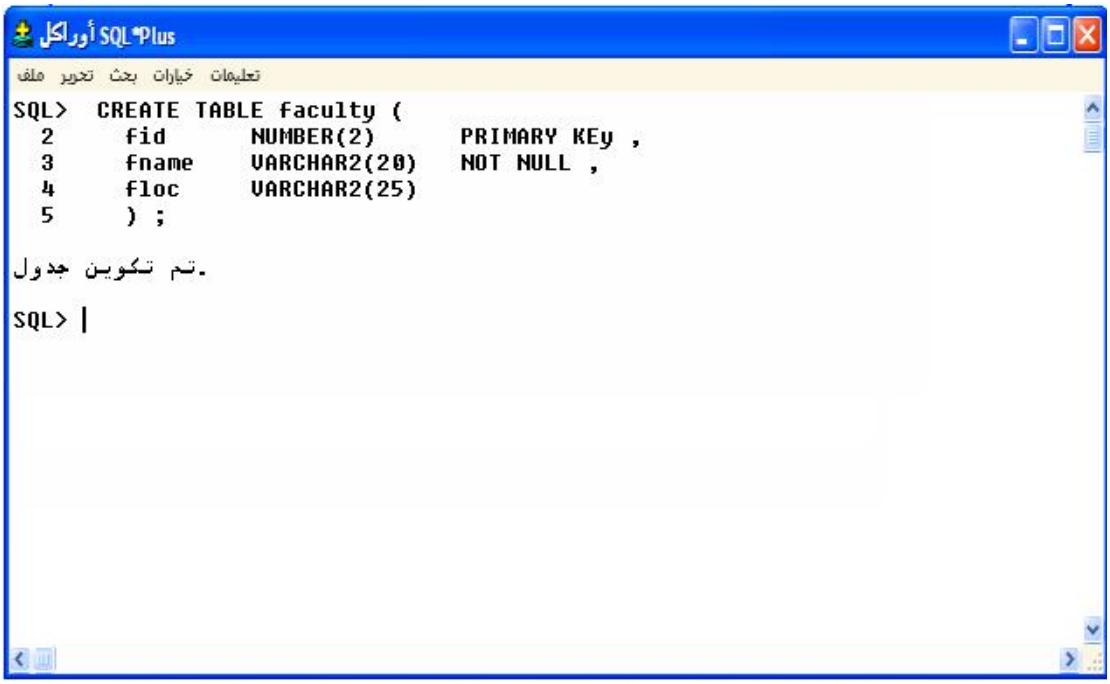
و المثال التالي ايضا عمل القيد UNIQUE



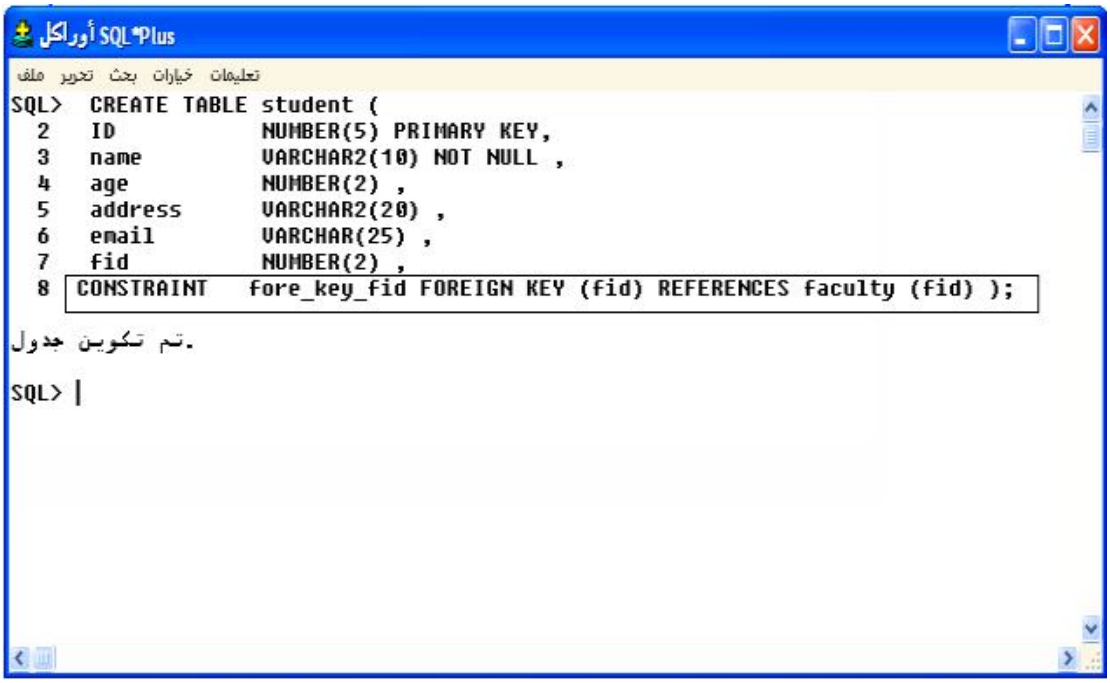
مثال(65)

المثال التالي يشرح كيفية عمل علاقة بين الجداول من خلال المفتاح الأجنبي Foreign Key

و سوف ننشئ جدولاً جديداً للكليات، و سوف يتم ربطه بواسطة جدول الطلاب:

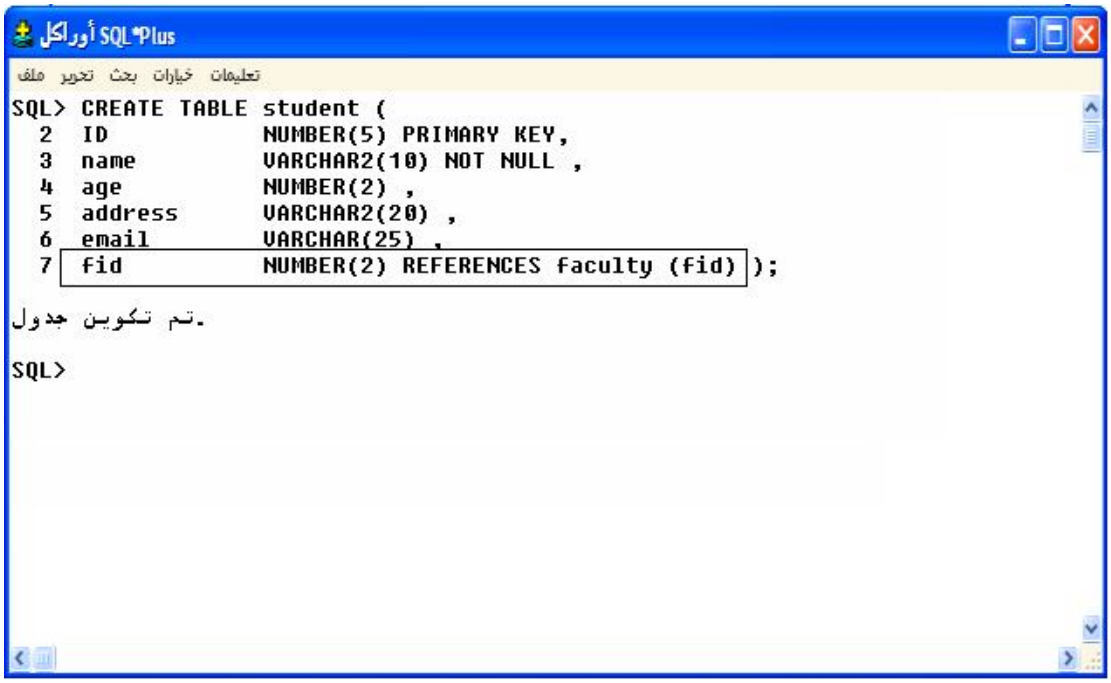


إنشاء الجدول

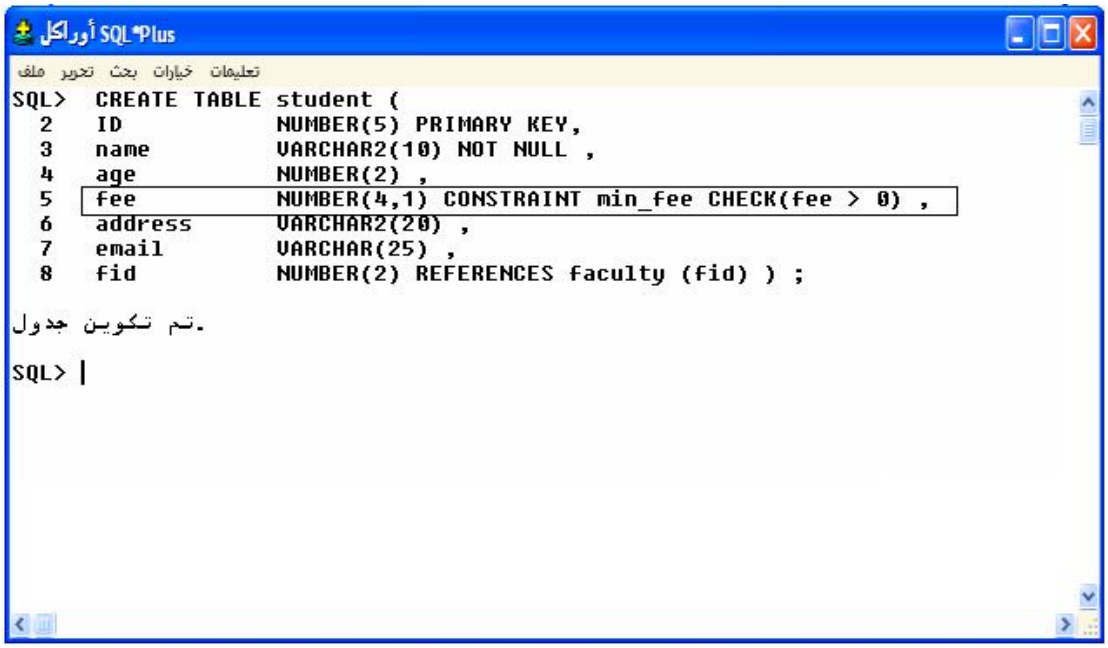


تم عمل قيد المفتاح الأجنبي.

و أيضا يمكن أن يكون هذا القيد على مستوى الحقل.



القيد Check يستخدم للتأكد من أن قيمة الحقل تحقق قيمة معينة قبل الإدخال. و المثال التالي يوضح أن قيمة الحقل Fee و يعني الرسوم يجب أن تكون أكبر من 0



أسئلة تقويم ذاتي

1/ عدد أنواع قيود السلامة.

2/ اكتب صيغة الجملة لعمل قيد عند إنشاء الجدول.

3/ اكتب الصيغ العامة لتعريف القيد على مستوي الحقول.

الخلاصة

عزيزي الدارس، لقد عرضت لك في هذه الوحدة الثانية من مقرر "تطبيقات قواعد البيانات" الموضوعات الأساسية التالية في دراسة التطبيق بواسطة لغة SQL .

- بينت الوحدة أن جملة SELECT تستخدم لاسترجاع البيانات من قاعدة البيانات.

كما بينت أنه من خلال جملة Select نستطيع أن ننفذ عدداً من العمليات. و هي كما يلي:
الإسقاط Projection – الاختيار Selection – الربط Joining .

- أوضحت الوحدة أنه يمكن في جمل الاستعلام استخدام عمليات حسابية على الحقول الرقمية. و العمليات التي يمكن استخدامها هي : + لعملية الجمع ، و – لعملية الطرح و * لعملية الضرب ، و / لعملية القسمة . كما بينت أنه عند تنفيذ هذه العمليات نجد أن الضرب و القسمة لها أولوية على الجمع و الطرح في التنفيذ .
- أوضحت الوحدة أن القيمة Null تعني أن هذه القيمة غير معرفة أو غير متوفرة و هي في أي حال من الأحوال لا تعني العدد صفراً في الحقول الرقمية و فراغ space في الحقول الحرفية. وبينت أنه يمكن تعريف أي حقل على أساس أنه null value ما لم يكن هنالك قيد سلامة Integrity Constraint معرف على هذا الحقل سواء أكان NOT NULL أو أن الحقل معرف كمفتاح أساسي .

وأوضحت الوحدة أنه عند إجراء أي عملية حسابية على حقل يحتوي على Null فإن ناتج التنفيذ سوف يكون أيضا Null .

- بينت الوحدة أنه من خلال جملة الاستعلام يمكن أن نغير اسم الحقل في عرض ناتج الاستعلام مع العلم أن اسم الحقل يظل كما هو في قاعدة البيانات.
- أوضحت الوحدة أنه في جملة الاستعلام يمكن أن نعرض مجموعة من بيانات الحقول كنص واحد، و يمكن أن يتخلل ذلك النص نصوص ثابتة، و نستخدم لتمثيل العملية الإلحاقية الرمز || .
- بينت الوحدة أن العبارة DISTINCT تستخدم بصورة اختيارية لمنع تكرار ظهور السجل بصورة متكررة كلياً و لا تمنع تكرار الحقول جزئياً.
- بينت الوحدة أن عبارة Where تستخدم مع جملة الاستعلام لإضافة شرط يجب أن يتحقق مقابل السجلات حتى تكون مؤهلة أن تكون في مجموعة حل الاستعلام.
- ذكرت الوحدة أن عبارة LIKE تستخدم لعمل مقارنات بين القيم و الحقول الحرفية .
- أوضحت الوحدة أنه يمكن أن نرتب البيانات الناتجة من الاستعلام باستخدام حقل و أكثر تصاعديا أو تنازلياً، و نستخدم في ذلك عبارة Order by و للترتيب التصاعدي نستخدم

الكلمة ASC و للترتيب التنازلي نستخدم DESC . و في حالة عدم استخدام أي من الكلمتين فإن الوضع الافتراضي هو ASC .

- بينت الوحدة أن هنالك نوعين من دوال السجل هما: Single-Row Functions و Multi-Row Function .

بينت الوحدة أن الدالة ( ) LOWER تستخدم لتحويل حالة الأحرف إلى أحرف صغيرة.

- بينت الوحدة أن دوال التعامل مع الأحرف تشمل الدوال التالية : CONCAT - SUBSTR – INSTR - LPAD , RPAD – TRIM .
- بينت الوحدة أن الدالة ROUND تستخدم لتقريب الأعداد.
- بينت الوحدة أن الدالة TRUNC تقوم باستقطاع جزء من العدد دون التقريب.
- بينت الوحدة أن الدالة MOD تستخدم لحساب باقي القسمة .
- بينت الوحدة أن الدالة SYSDATE تستخدم هذه الدالة لاسترجاع التاريخ الحالي للجهاز.
- بينت الوحدة أن الدالة ( )TO_CHAR تستخدم هذه الدالة لتحويل البيانات من نوع تاريخ أو نوع عددي إلى نوع نصي.
- بينت الوحدة أن الدالة ( )NVLتستخدم هذه الدالة لاستبدال قيمة معينة في حالة أن قيمة الحقل غير معرفة.
- هنالك عدة أنواع يمكن أن تستخدم لعمل ربط بين الجداول التي ترد في الاستعلام الواحد، و هي كما يلي:

الربط بالتساوي– الربط بعدم التساوي– الربط الخارجي– الربط الذاتي ،الصيغة العامة للربط بين جداول الاستعلام الواحد:

```
SELECT   table1.column, table2.column
FROM     table1, table2
WHERE    table1.column1 = table2.column2;
```

وصفت الوحدة الدوال التي يمكن استخدمها على مجموعة من السجلات كما يلي:

( ) AVG: تستخدم لإيجاد المتوسط الحسابي للحقل العددي و تتجاهل الحقول التي تحمل القيمة NULL.

( ) COUNT : تستخدم لمعرفة عدد السجلات و عند استخدام COUNT(*) فذلك يعني استخدام كل السجلات و لو كانت مكررة أوكانت تحتوي على عبارة NULL.

( )MAX : تستخدم لإيجاد أكبر قيمة عددية و تتجاهل NULL.

MIN( ) : تستخدم لإيجاد أصغر قيمة عددية و تتجاهل NULL.

SUM( ): تستخدم لجمع مجموعة من الأعداد و تتجاهل NULL.

و الصيغة العامة للتعامل مع هذه الدوال هي:

```
SELECT        [column,] group_function(column), ...
FROM          table
[WHERE        condition]
[GROUP BY     column]
[ORDER BY     column];
```

وضحت الوحدة أن العبارة GROUP BY تستخدم لتقسيم السجلات في شـــكل مجموعـــات، على أن تطبق دوال المجموعات على كل مجموعة كل على حدة و بشكل مستقل.

كذلك وضحت الوحدة أن العبارة HAVING تستخدم لعمل شرط على كل مجموعات الحل و تقابل Where و التي تعمل على مستوى السجلات.

الصيغة العامة للاستعلامات الفرعية هي:

```
SELECT   select_list
FROM     table
WHERE    expr operator
                    (SELECT      select_list
                    FROM         table);
```

الاستعلام الفرعي

الصيغة العامة لجملة الإدخال INSERT:

```
INSERT INTO   table [(column [, column...])]
VALUES        (value [, value...]);
```

الصيغة العامة لجملة تعديل البيانات UPDATE:

```
UPDATE        table
SET           column = value [, column = value, ...]
[WHERE        condition];
```

الصيغة العامة لجملة DELETE:

```
DELETE [FROM]   table
[WHERE          condition];
```

الصيغة العامة لإنشاء جدول:

```
CREATE TABLE [schema.]table
            (column datatype [DEFAULT expr][, ...]);
```

بينت الوحدة أنه يمكن أن يتم تعديل الجدول بعد الإنشاء وتشمل عملية التعديل العمليات التالية:

إضافة حقول جديدة، تعديل مواصفات حقل موجود، وحذف حقل.

عرضت الوحدة أنواعاً من قيود السلامة:

NOT NULL: وتعني أن يكون الحقل غير فارغ.

UNIQUE: و تعني أن يكون الحقل أو مجموعة من الحقول غير مكرره.

Primary Key : لعمل مفتاح أساسي من حقل أو مجموعة من الحقول.

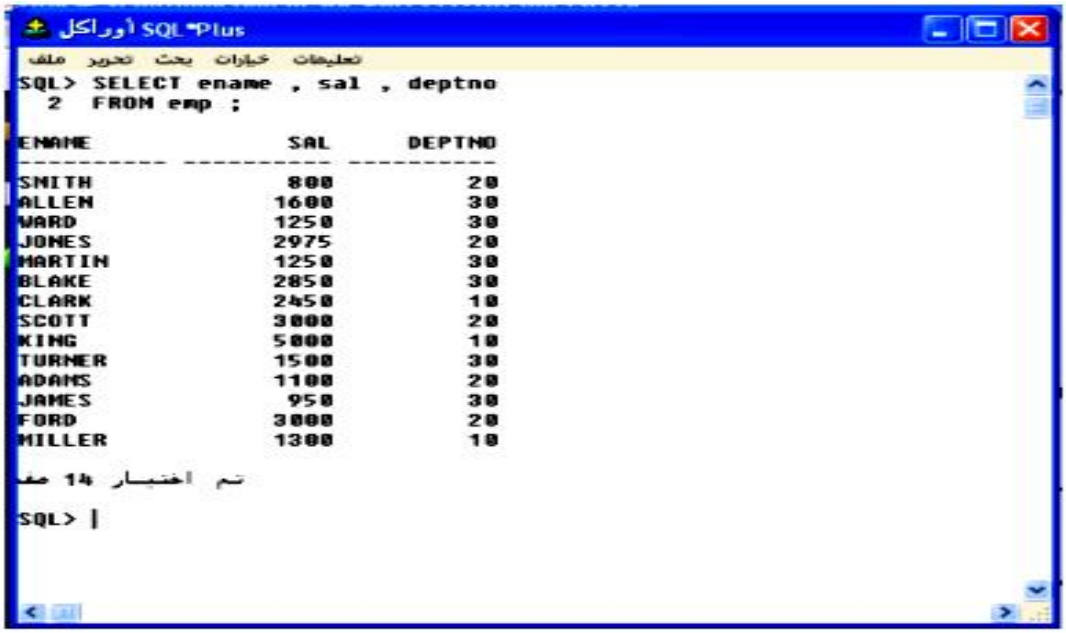
Foreign Key : لعمل المفتاح الأجنبي.

CHECK للتحقق من أن قيمة حقل محققة معينة.

إجابات التدريبات

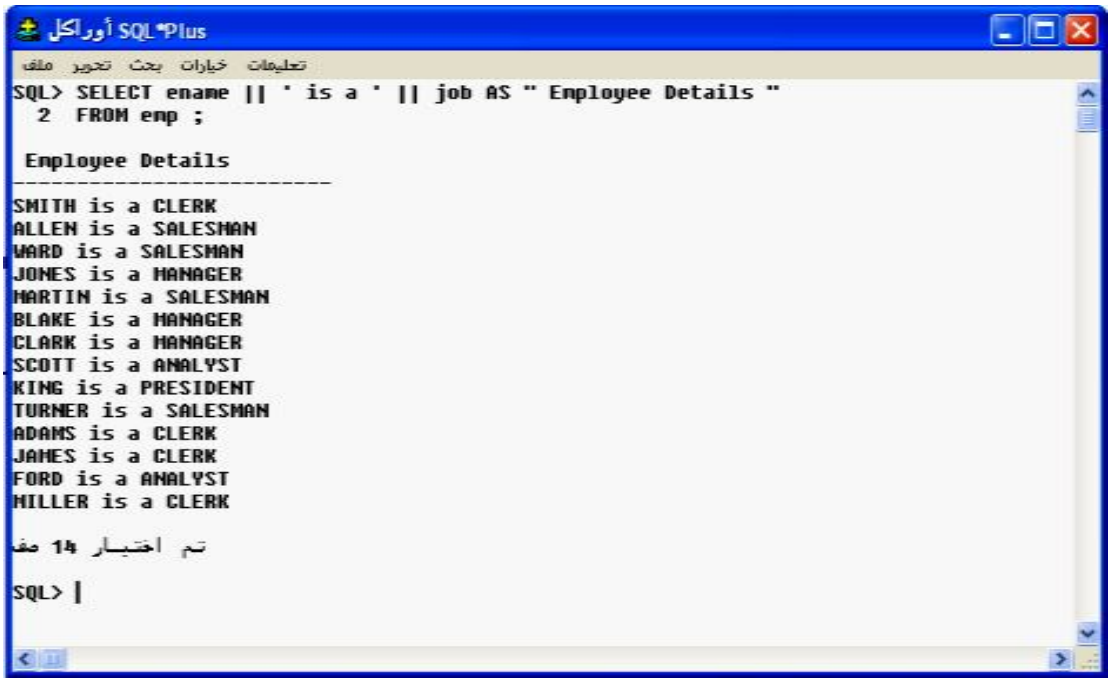
تدريب (1)

الحل:



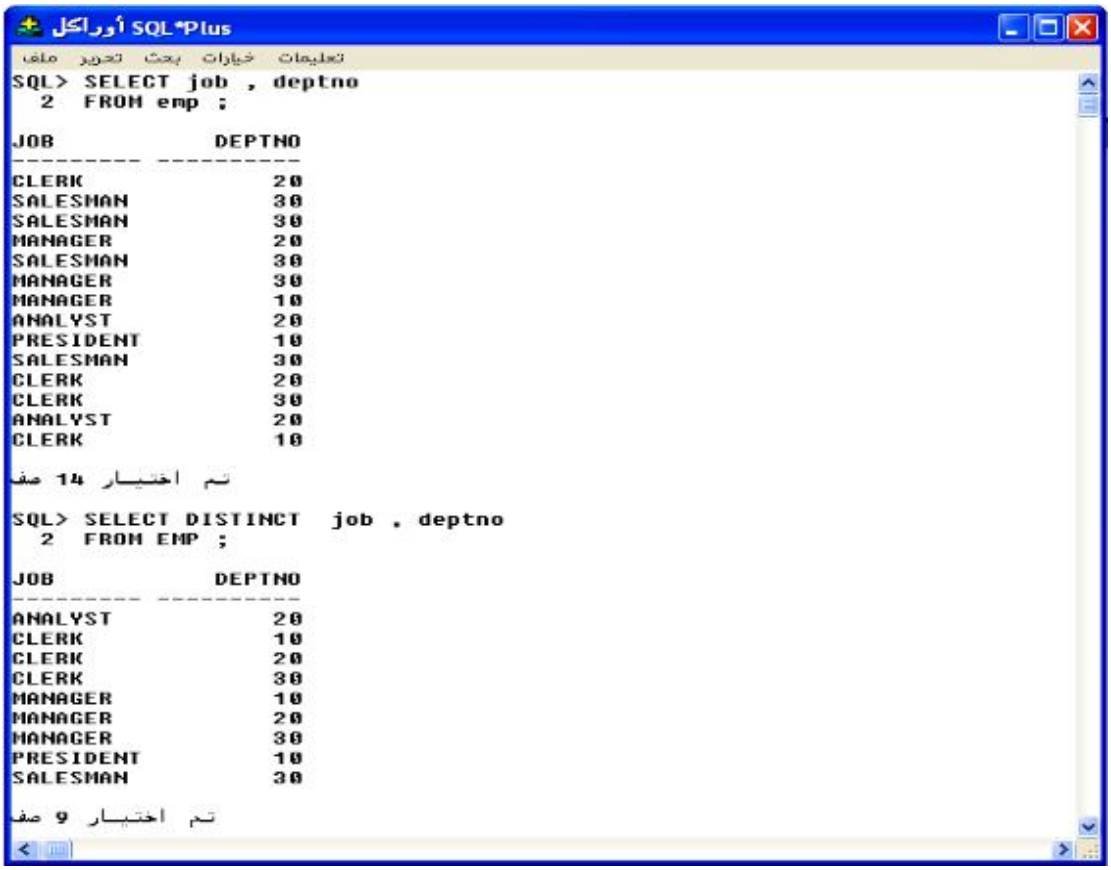
تدريب (2)

الحل:



تدريب (3)

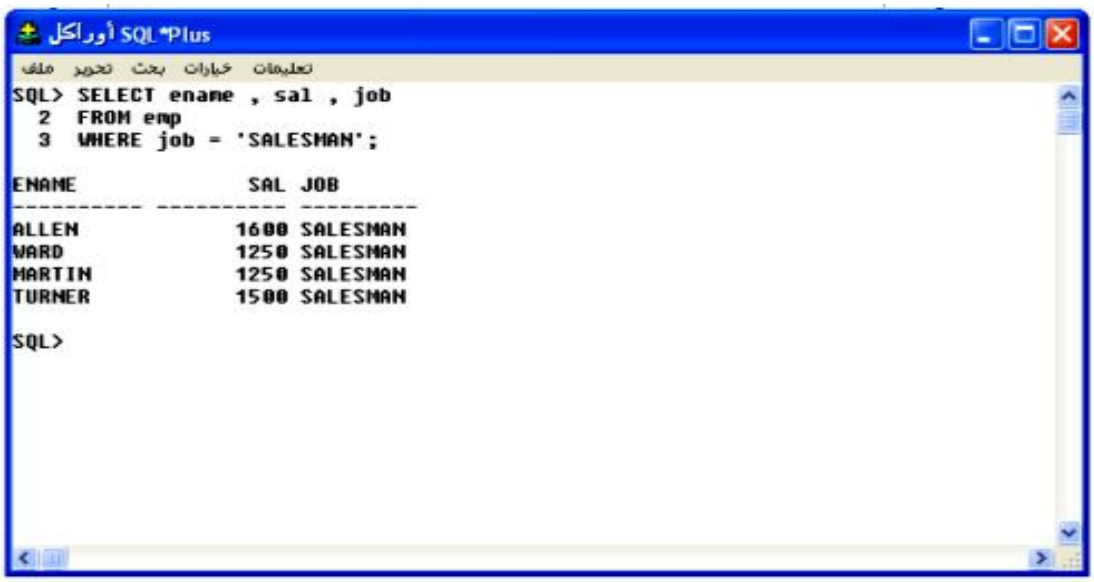
الحل:



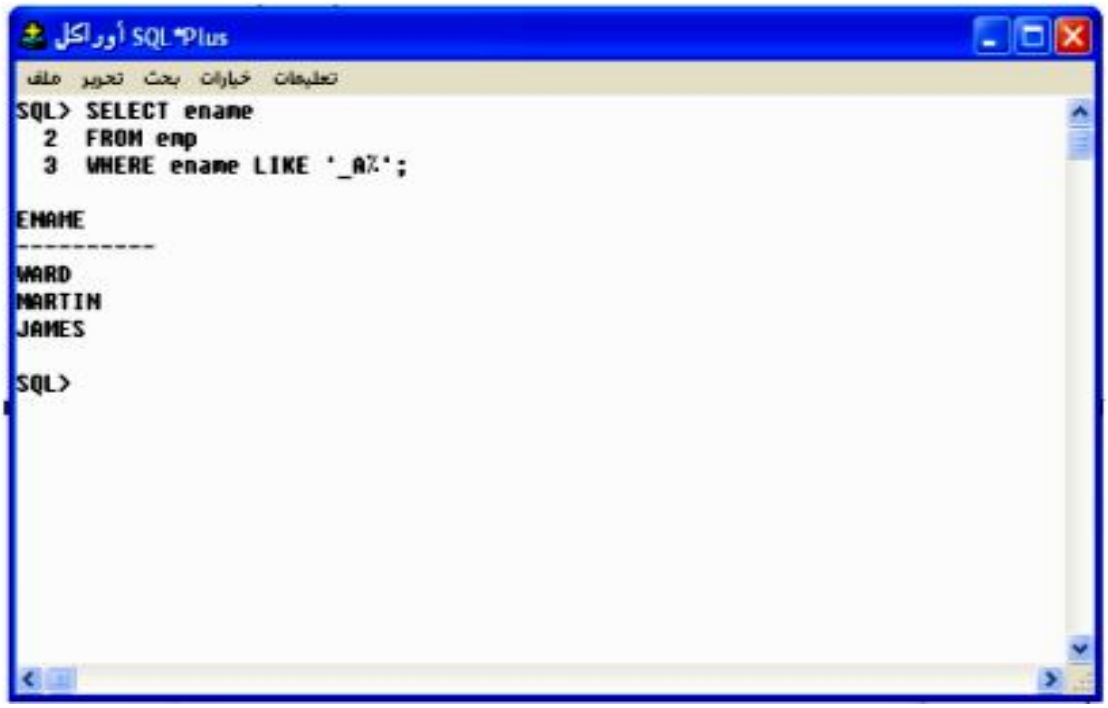
في الجملة الأولى ظهر لنا تكرار لبعض السجلات، و في المرة الثانية أزيل التكرار على مستوى السجلات و لكنه بقي على مستوى الحقول.

تدريب (4)

الحل:

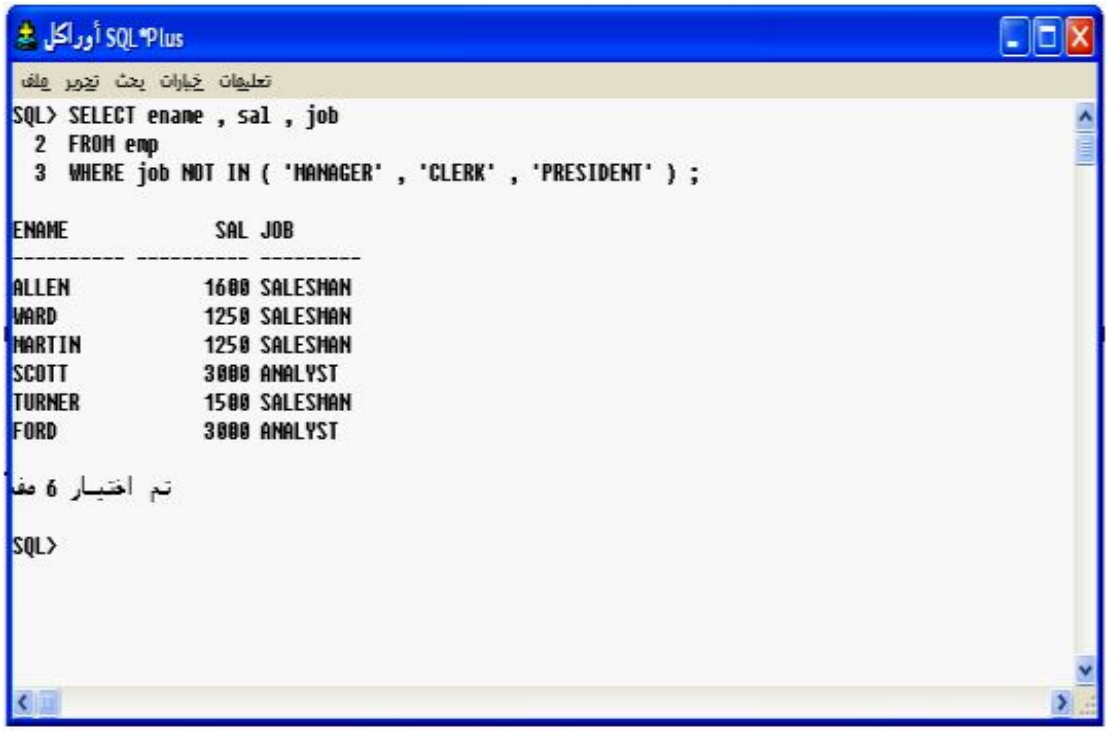


تدريب (5)



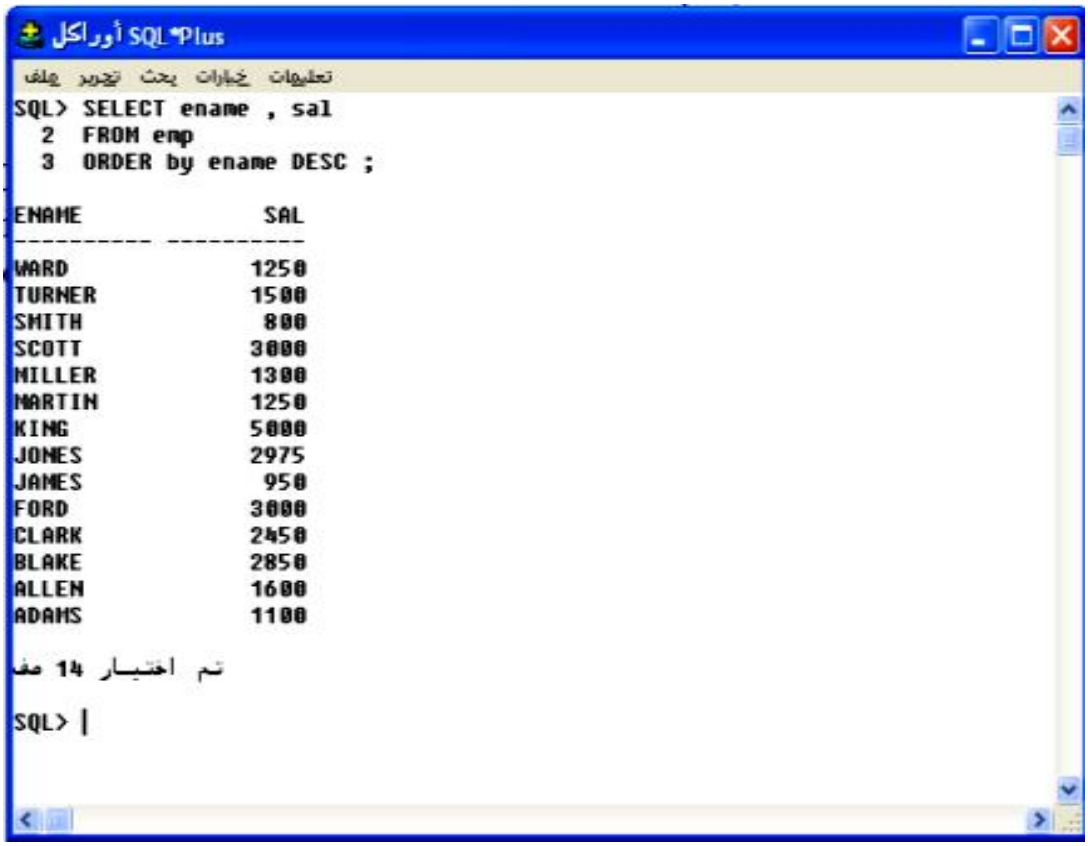
تدريب (6)

الحل:



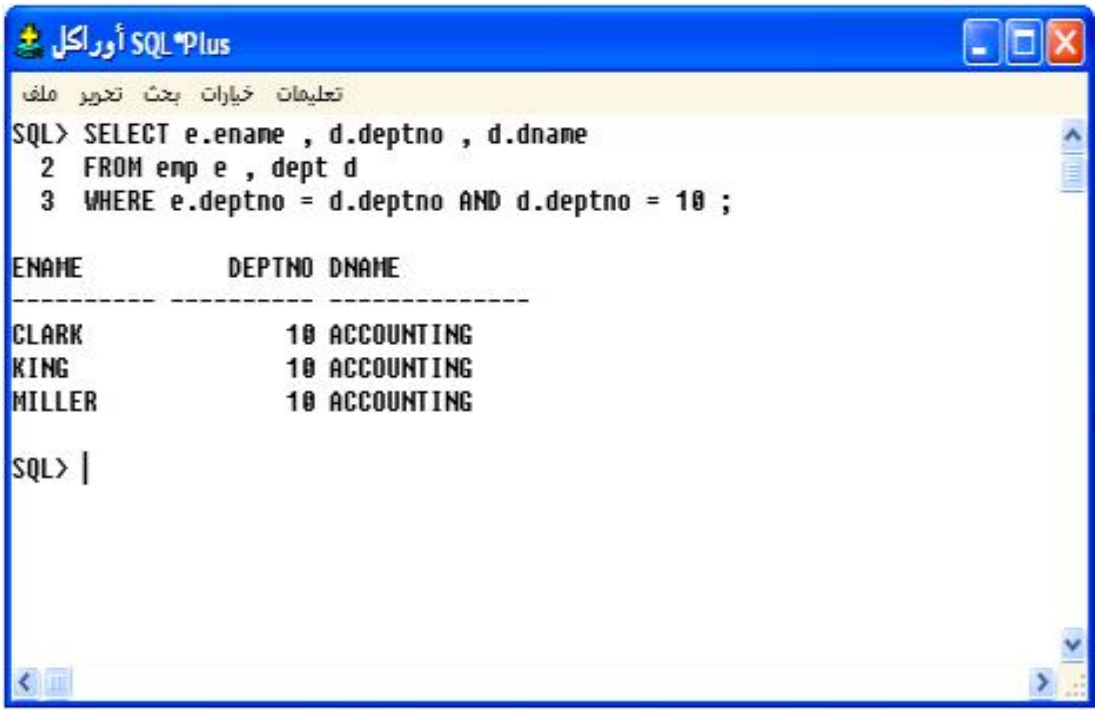
تدريب (7)

الحل:



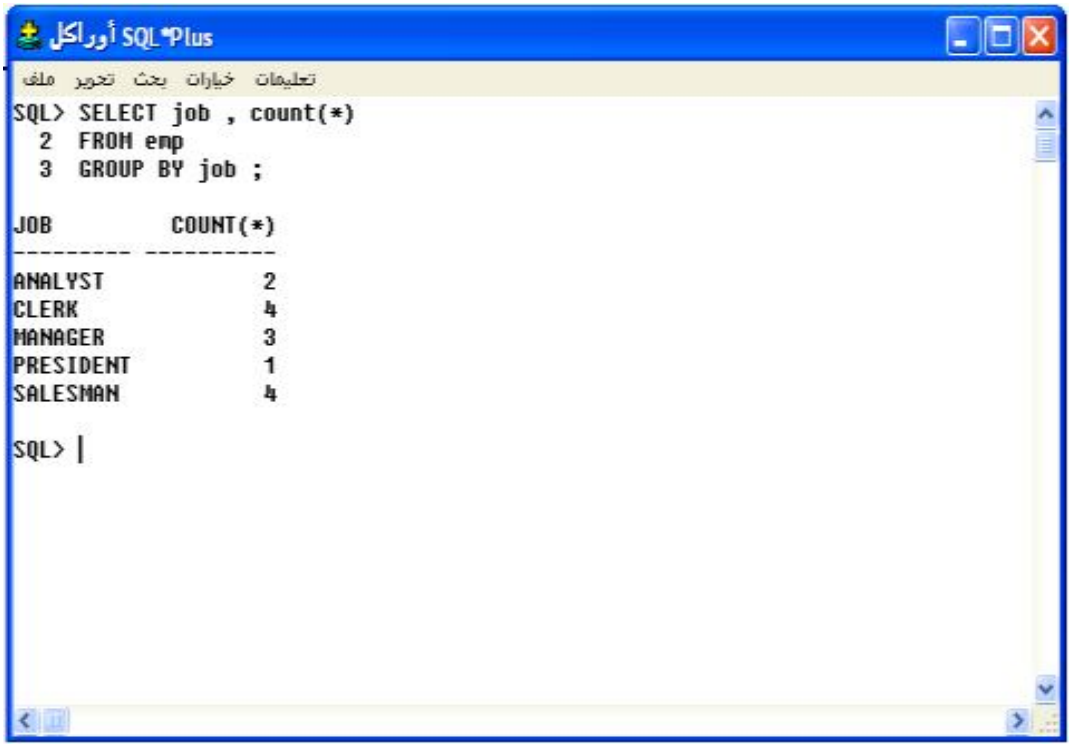
تدريب (8)

الحل:



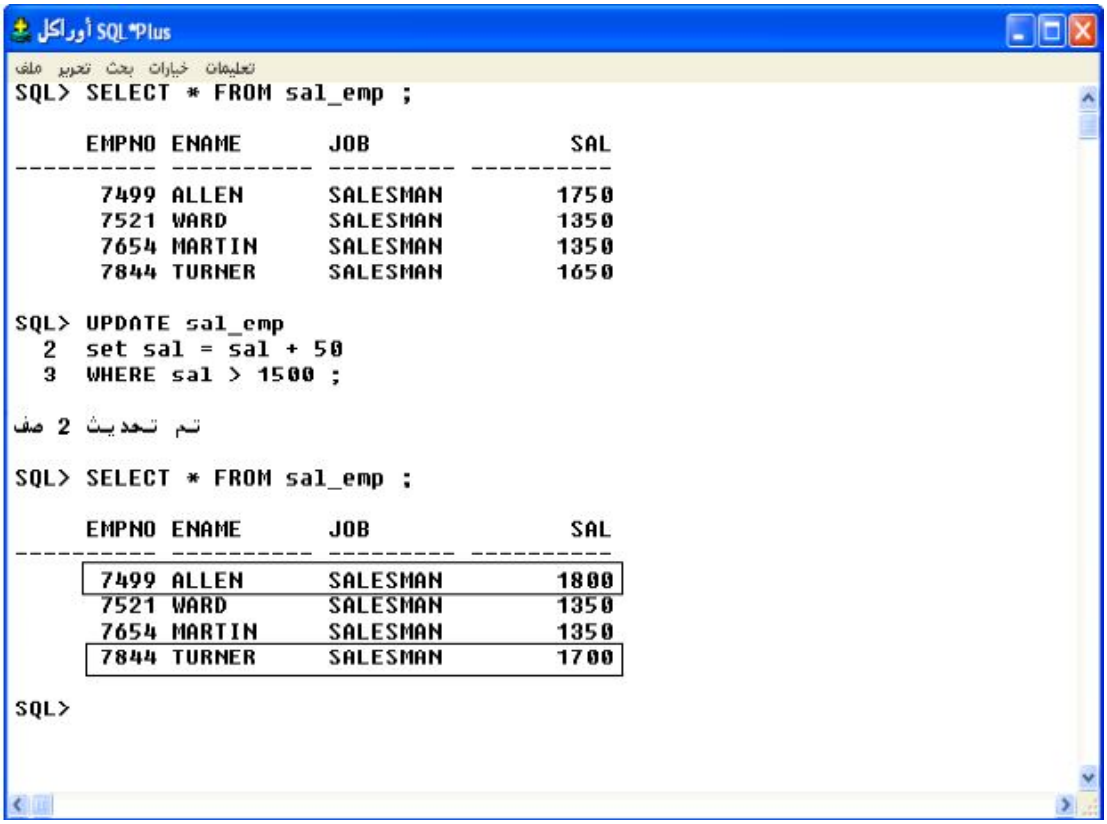
تدريب (9)

الحل:



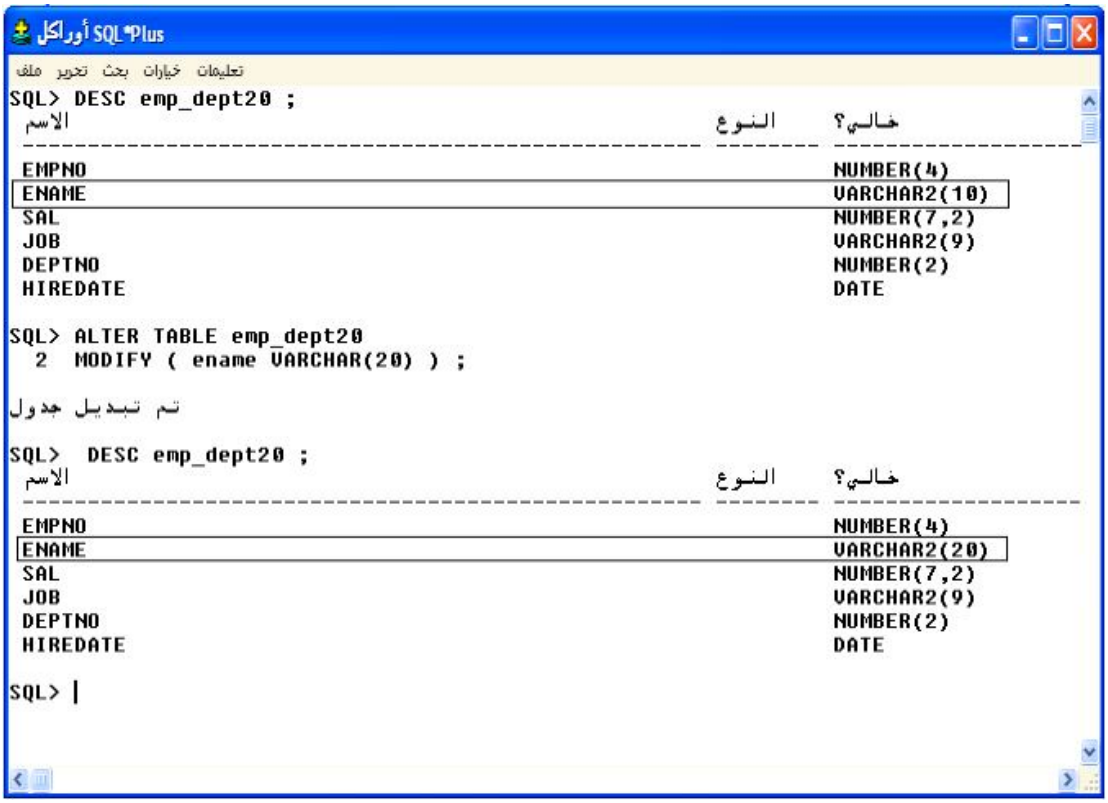
تدريب (10)

الحل:



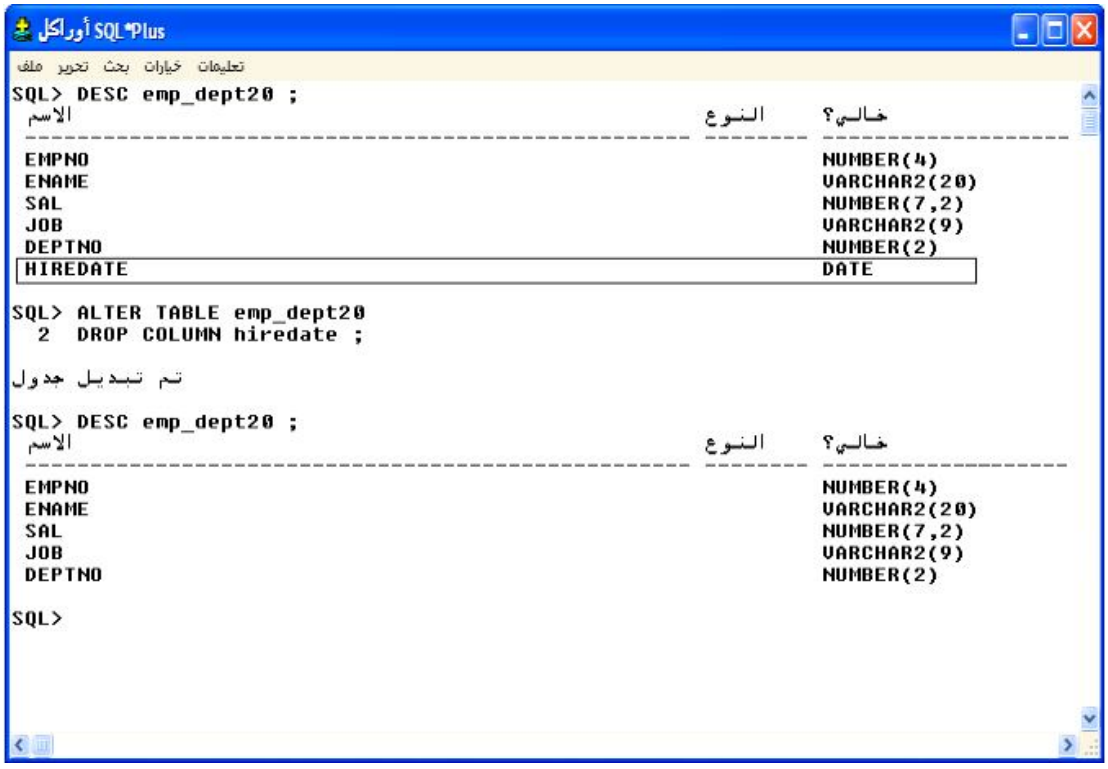
تدريب (11)

الحل:



تدريب (12)

الحل:



مسرد المصطلحات

جملة SELECT

إحدى جمل لغة SQL وتستخدم هذه الجملة لاسترجاع البيانات من قاعدة البيانات.

جملة الإدخال INSERT

إحدى جمل لغة SQL وتستخدم هذه الجملة لإدخال السجلات داخل قاعدة البيانات.

جملة تعديل البيانات UPDATE

إحدى جمل لغة SQL وتستخدم هذه الجملة لتعديل بيانات السجلات على أن يتم التعديل على مستوى الحقول.

جملة الحذف DELETE

إحدى جمل لغة SQL وتستخدم جملة DELETE لحذف السجلات .

الإسقاط Projection

هذه العملية تمكننا من عرض حقول معينة في الجدول و إمكانية إسقاط الحقول غير المطلوبة.

الاختيار Selection

هذه العملية تمكننا من عمل فرز لبعض السجلات من الجدول و التي تحقق شرطاً معيناً يتم تحديده في الجملة.

الربط Joining

تتيح هذه العملية إسترجاع البيانات من عدة جداول.

العبارة DISTINCT

تستخدم هذه العبارة بصورة اختيارية لمنع تكرار ظهور السجل بصورة متكررة كلياً، و لاتمنع تكرار الحقول جزئياً.

عبارة Where

تستخدم هذه العبارة مع جملة الاستعلام لإضافة شرط يجب أن يتحقق مقابل السجلات حتى تكون مؤهلة أن تكون في مجموعة حل الاستعلام.

دوال التحويل Conversion Functions

يستخدم هذا النوع من الدوال لتحويل بيانات الحقل من نوع إلى نوع آخر.

العبارة GROUP BY

تستخدم هذه العبارة لتقسيم السجلات في شكل مجموعات على أن تطبق دوال المجموعات على كل مجموعة كل على حدة و بشكل مستقل.

العبارة HAVING

تستخدم هذه العبارة لعمل شرط على كل مجموعات الحل و تقابل Where و التي تعمل علـــى مستوى السجلات.

قيود السلامة Integrity Constraints

تستخدم قيود السلامة على مستوى الجداول للتحقق من سلامة العمليات التي تجرى على الجداول. و هنالك عدة أنواع من قيود السلامة .

References المراجع

1. Database Management Systems (Third Edition)

, 2002.Raghu Ramakrishnan and Johannes Gehrke

2. Fundamentals of Database System , (Third Edition)

Ramez Elmasri and Shamkant B. Navatlac , 2000

3. Database System , Design , Implementation & Management (Fifth Edition)

Rob Coronel 2002

4. Introduction to Oracle9i , Oracle Education, Oracle Company