

CORE 401 – Oracle

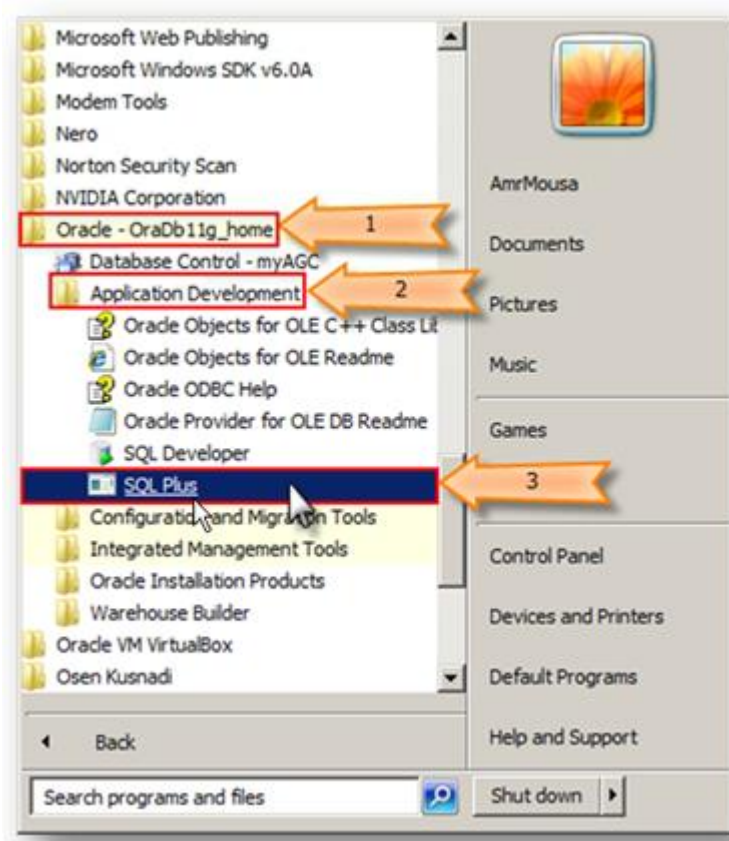


يحتوى اوراق على العديد من المميزات ومن اهمها امكانية تشغيله على كافة انظمة التشغيل (ويندوز – لينكس - ...) وقوة قواعد البيانات الخاصة به ويعيبه ارتفاع اسعار كثيرا مقارنة ب اس كيو ال سيرفر كما ان اوركمل يحتوى على الاوركمل فورم التى يمكن تصنيع البرامج بواسطتها ولكنها اضعف كثيرا من الفورمز الخاصة بلغات البرمجة.

Using Sql Plus



هو برنامج يمكن من خلاله الدخول والتعامل مع قواعد بيانات اوراكل وذلك من خلال الخطوات التالية





ويتميز اوركل بمستويات الامان العالية جدا من خلال استخدام صلاحيات المستخدمين والتي تحدد لكل مستوى النطاق الذى يمكن التعامل فيه وقواعد البيانات الخاصة به ومدى قدرته على التعامل معها



SQL Plus window showing the execution of the query `SELECT * FROM TAB;`. The output displays a table with columns `TNAME`, `TABTYPE`, and `CLUSTERID`. The results show four rows: `BONUS`, `DEPT`, `EMP`, and `SALGRADE`, all of type `TABLE`. An orange callout box points to the query and contains the text: "تم ارسال الامر لقاعدة البيانات بعرض جميع الجداول المتاحة للمستخدم scott".

```
SQL*Plus: Release 11.2.0.1.0 Production on 11/01/2010
Copyright (c) 1982, 2010, Oracle. All rights reserved.
Enter user-name: SCOTT
Enter password:
Connected to:
Oracle Database 11g Release 11.2.0.1.0 - Production
SQL> SELECT * FROM TAB;
```

TNAME	TABTYPE	CLUSTERID
BONUS	TABLE	
DEPT	TABLE	
EMP	TABLE	
SALGRADE	TABLE	

```
SQL>
```

كما يمكن استخدام الاوامر الخاصة بعرض الجداول والتعامل مع قاعدة البيانات من خلال هذا البرنامج



SQL Plus window showing the execution of the `CONNECT` command. The user `SYSTEM` is connected. An orange callout box points to the `CONNECT` command and contains the text: "تم الدخول بالمستخدم الأدمن SYSTEM".

```
7902 FORD ANALYST 3000
7934 MILLER CLERK 1300
14 rows selected.
SQL> CONNECT
Enter user-name: SYSTEM
Enter password:
Connected.
```

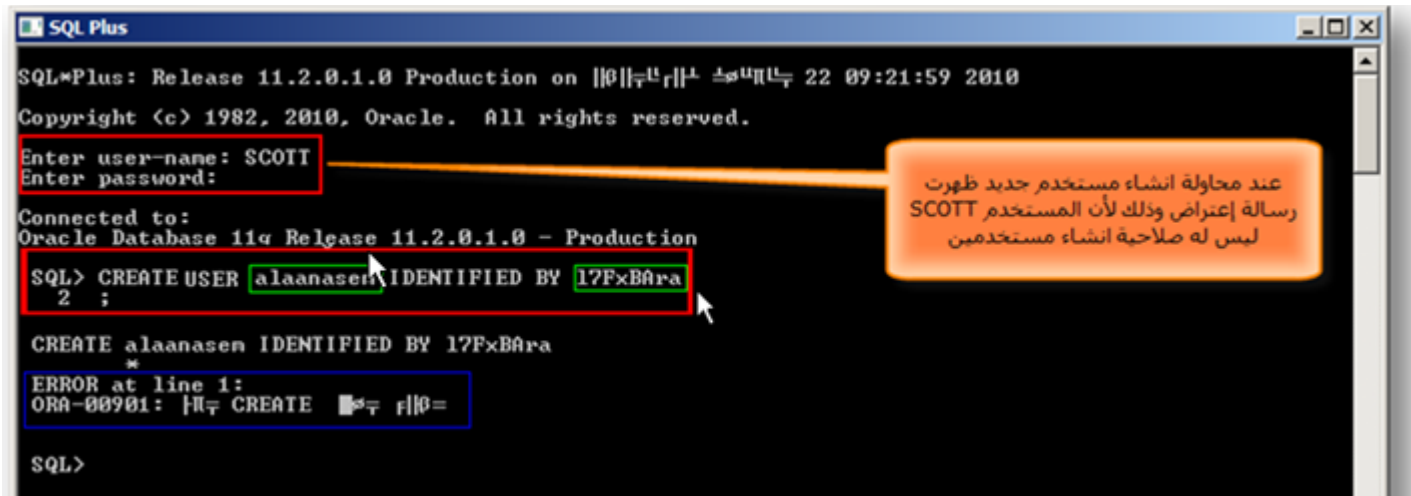
ويمكن تغيير المستخدم من خلال الامر connect

الوامر المستخدمة للتعامل مع البيانات والجداول وقواعد البيانات

القسم	الامر	وصف الامر
Data Retrieval	SELECT	أمر استرجاع البيانات من جدول أو كائن
(DML) Data Manipulation Language	INSERT	أمر إضافة بيانات إلى جدول أو كائن
	UPDATE	أمر التعديل في بيانات جدول أو كائن
	DELETE	أمر حذف بيانات جدول أو كائن
(DDL) Data Definition Language	CREATE	أمر إنشاء جدول أو كائن
	Alter	أمر التعديل في جدول أو كائن
	DROP	أمر إلغاء جدول أو كائن
	RENAME	أمر تغيير الاسم جدول أو كائن
	TRUNCATE	إلغاء جزء أو بقر جزء من جدول أو كائن
Transaction Control	COMMIT	تثبيت البيانات في الجدول
	ROLLBACK	الرجوع عن تثبيت البيانات
	SAVEPOINT	الرجوع لنقطة معينة
(DCL) Data Control Language	GRANT	إعطاء الصلاحيات للمستخدمين للدخول على البيانات
	REVOKE	سحب الصلاحيات من المستخدمين

إنشاء مستخدم جديد

```
CREATE USER user_name  
IDENTIFIED BY password;
```



The screenshot shows a SQL Plus window with the following text:

```
SQL*Plus: Release 11.2.0.1.0 Production on 22 09:21:59 2010  
Copyright (c) 1982, 2010, Oracle. All rights reserved.  
Enter user-name: SCOTT  
Enter password:  
Connected to:  
Oracle Database 11g Release 11.2.0.1.0 - Production  
SQL> CREATE USER alaanasen IDENTIFIED BY 17FxB0ra  
2 ;  
CREATE alaanasen IDENTIFIED BY 17FxB0ra  
*  
ERROR at line 1:  
ORA-00901: cannot create user  
SQL>
```

An orange callout box on the right contains the following Arabic text:

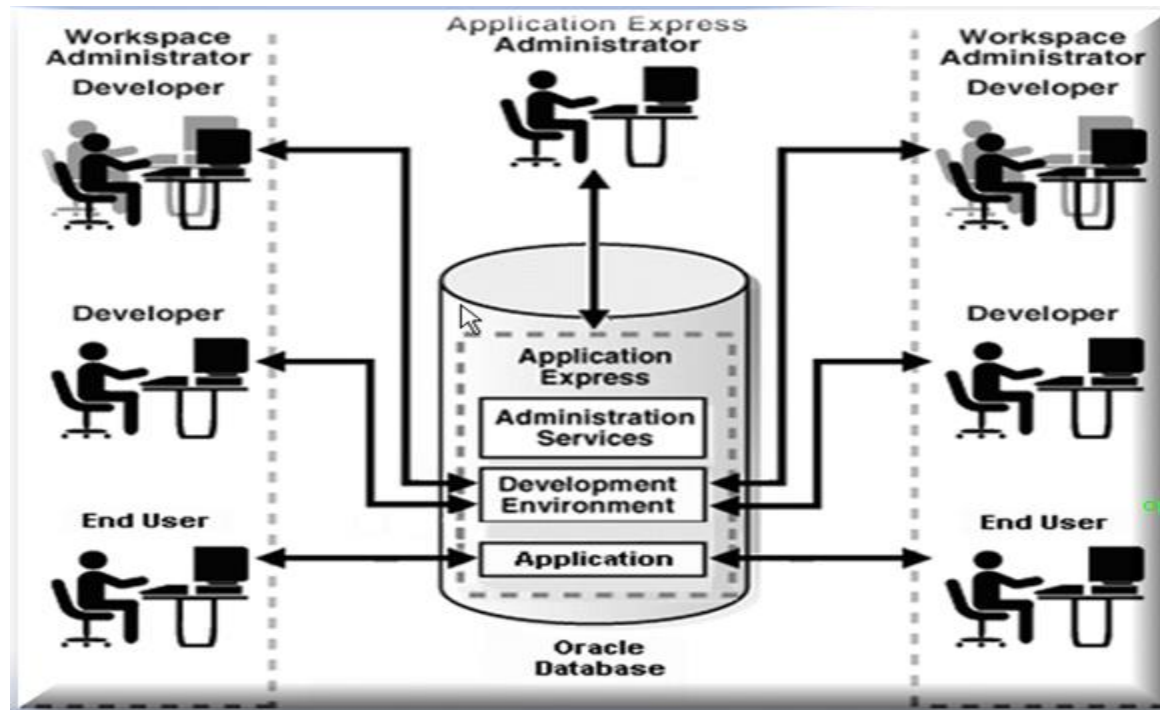
عند محاولة إنشاء مستخدم جديد ظهرت رسالة إعتراض وذلك لأن المستخدم SCOTT ليس له صلاحية إنشاء مستخدمين

يجب لإنشاء مستخدم استخدام الامر السابق ولكن عن طريق مستخدم لديه صلاحية إنشاء مستخدمين جدد

التحكم في صلاحيات المستخدمين

DCL

Data Control Language



اكواد بعض الامتيازات التي يمكن اعطاؤها للمستخدم

امتيازات النظام System Privileges. يوجد هناك أكثر من ٨٠ امتياز للنظام متوفرة للمستخدمين والوظائف Roles	
الامتياز	النشاطات الممكنة أدائها
CREATE USER	إنشاء مستخدم، وإسناده حصة من أي مساحة جدولية Tablespace.
DROP USER	إسقاط أو حذف مستخدم آخر.
ALTER USER	تعديل مستخدم آخر: تغيير كلمة المرور لمستخدم غير المستخدم الحالي .
CREATE ANY TABLE	إنشاء جدول في أي مخطط Schema
DROP ANY TABLE	حذف جدول من أي مخطط
CREATE SEQUENCE	إنشاء تتابع أو سلسلة في المخطط الخاص بالمستخدم
DROP SEQUENCE	حذف التتابع
CREATE TRIGGER	إنشاء زناد في المخطط الخاص بالمستخدم
DROP TRIGGER	حذف الزناد
ALTER TRIGGER	تعديل الزناد: تعطيل، تفعيل أو إعادة ترجمة الزناد
CREATE VIEW	إنشاء جدول وهمي (عرض)
DROP ANY VIEW	حذف عرض من أي مخطط
CREATE ROLE	إنشاء وظيفة
ALTER ANY ROLE	تعديل أي وظيفة في قاعدة البيانات
DROP ANY ROLE	حذف أي وظيفة من قاعدة البيانات
GRANT ANY ROLE	منح أي وظيفة في قاعدة البيانات
CREATE SYNONYM	إنشاء مرادف في قاعدة البيانات الخاصة بالمستخدم
DROP ANY SYNONYM	حذف أي مرادف من قاعدة البيانات الخاصة بالمستخدم

مثلا من اهم الامتيازات الممكن منحها لمطور تطبيقات قواعد البيانات من الجدول أعلاه ما يلي:

- القدرة على الاتصال بقاعدة البيانات CREATE SESSION
- إنشاء جدول CREATE TABLE
- إنشاء تتابع CREATE SEQUENCE
- إنشاء عرض CREATE VIEW
- إنشاء الروتين CREATE PROCEDURE

صيغة كود اعطاء الصلاحية للمستخدم

```
GRANT privilege [, privilege ...]  
TO user_name [, user_name, ...];
```

```
SQL> conn
Enter user-name: alaanasin
Enter password:
ERROR:
ORA-01017: invalid username/password; logon denied
```

محاولة المستخدم علاء الدخول وظهور
اعتراض عليه لعدم وجود صلاحية الدخول

```
SQL> conn
Enter user-name: system
Enter password:
Connected.
```

إعادة الدخول بإسم
المشرف العام

```
SQL> GRANT CREATE SESSION TO alaanasin;
Grant succeeded.
```

أعطاء صلاحية الدخول
للمستخدم علاء

```
SQL> connect alaanasin/12345;
Connected.
SQL> _
```

عند محاولة علاء الدخول مرة أخرى
على قاعدة البيانات تم الدخول بنجاح

SQL*Plus: Release 11.2.0.1.0 Production on ٢٠١٠/١١/٢٢ 16:32:11 2010

Copyright (c) 1982, 2010, Oracle. All rights reserved.

Enter user-name: SYSTEM
Enter password:

Connected to:
Oracle Database 11g Release 11.2.0.1.0 - Production

```
SQL> GRANT CONNECT , RESOURCE TO ALAANASIM;
```

Grant succeeded.

SQL> _

تم اعطاء للمستخدم جميع الصلاحيات
الأساسية التي توفر للمستخدم
الاتصال والتعامل مع قاعدة البيانات

Create Table

```
SQL> CONN ALAANASIM;  
Enter password:  
Connected.  
SQL> CREATE TABLE STUDENTS(  
2  USERNAME VARCHAR(20),  
3  PASSWORD VARCHAR(20),  
4  MOBILE   VARCHAR(20)  
5  );
```

الدخول بإسم المستخدم علاء وإنشاء جدول جديد

Table created.

Create Procedure

```
SQL> CREATE OR REPLACE PROCEDURE NewStu (usr varchar2,pwd varchar2,nbl varchar2)  
2  IS  
3  BEGIN  
4  INSERT INTO STUDENTS(USERNAME,PASSWORD,MOBILE) VALUES (usr,pwd,nbl);  
5  end;  
6  /  
Procedure created.  
SQL> execute NewStu('ArMousa','@34D5!23','@106675159');  
PL/SQL procedure successfully completed.  
Procedure created.
```

Change User Password

```
Connected to:
Oracle Database 11g Release 11.2.0.1.0 - Production
SQL> ALTER USER ALANASIM IDENTIFIED BY 2010 ;
User altered.
SQL>
```

تم تغيير كلمة مرور المستخدم لتصبح 2010

لكتابة هذا الكود يجب ان يحصل المستخدم على صلاحيات ال ادمن لانه الوحيد الذى له صلاحية تغيير كلمة المرور للمستخدم

Create Synonym

يستخدم هذا الامر لعمل مشاركة لجدول بين المستخدمين

```
SQL> connect system
Enter password:
Connected.
SQL> Create Public Synonym Pstu FOR Ahmed.Students;
Synonym created.
```

ويستخدم الامر التالى لحذف المشاركة الموجودة على الجدول

```
SQL> conn system/123;
Connected.
SQL> Drop Public Synonym PStu force;
Synonym dropped.
```

SQL Language Reference

لغة SQL هي لغة لا تحتوى واجهة للمستخدم على وتتكون من مجموعة من الاسكريبتات .

<p>يستخدم هذا الامر فى عرض الهيكل الخاص بالجدول ومواصفات الحقول ويستخدم بالصيغة التالية:</p> <p>DESC + Field Name</p>	<p>DESC</p>
<p>يستخدم هذا الامر لعدم تكرار البيانات اثناء عرضها ويستخدم بالصيغة</p> <pre>SQL> SELECT DISTINCT CUST FROM SALES;</pre> <p>CUST</p> <p>-----</p> <p>ALY KAREN RANY TAME</p> <p>التالية:</p>	<p>DISTINCT</p>
<p>يتم تنفيذ العمليات الحسابية داخل جملة السيكت بشرط ان تكون الحقول رقمية ويكون اولية تنفيذ العمليات الحسابية كالتالى (الاقواس ثم الضرب ثم الجمع ثم الطرح)</p>	<p>العمليات الحسابية</p>

<p>يستخدم لاضافة ارقام الى الحقل فى حالة كان الحقل يحتوى على القيمة نل وصيغته كالتالى :</p> <pre>SQL> SELECT NUL(REVAL+2,2) 2 FROM ITEMS;</pre> <pre>NUL(REVAL+2,2) ----- 12 2 22 7</pre>	<p>NUL</p>
<p>وذلك لتعديل اسم الحقل اثناء العرض ويكون بالصيغة التالية :</p> <pre>SQL> SELECT CUST "CUSTOMER NAME" 2 FROM SALES;</pre> <pre>CUSTOMER ----- TAMER ALY RAMY KAREN RAMY ALY RAMY KAREN 8 rows selected.</pre>	<p>Elias Name</p>
<p>وذلك لتعديل امر مكتوب ثم اعادة تنفيذه ب Oracle-Sql Plus</p>	<p>Edit - R</p>
<p>تستخدم للاستعلام بشرط معين ونستخدم معها الكثير من الدوال منها :</p> <p>Between - like - < > = >= <= <> and or In Isnull</p> <p>Order By</p>	<p>Where</p>
<p>المعاملات المنطقية وتستخدم للمقارنة بين حقلين او اكثر</p>	<p>< - > - <> - <= - >= - =</p>
<p>تستخدم للحصول على قيمة متواجدة فى حقل معين ونستخدم معها الرمز % سواء قبل القيمة او بعدها.</p>	<p>Like</p>

```
SQL> SELECT * FROM SALES
2 WHERE CUST LIKE 'A%';
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
2	KEYBOARD	ALY	25	30	750	01/01/08
6	MOUSE	ALY	35	20	700	06/01/08

تستخدم للحصول على البيانات المذكورة داخل الاقواس

```
SQL> SELECT * FROM
2 SALES WHERE PRIC IN (35 , 80 );
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
3	SPEAKER	RAMY	80	10	800	01/01/08
5	MOUSE	RAMY	35	20	80	04/01/08
6	MOUSE	ALY	35	20	700	06/01/08
7	KEYBOARD	RAMY	80	10	800	08/01/08

In(con1, con2,)

تستخدم للحصول على البيانات المحصورة بين القيمتين

```
SQL> SELECT * FROM
2 SALES WHERE PRIC BETWEEN 10 AND 100;
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
1	MOUSE	TAMER	12	20	240	01/01/08
2	KEYBOARD	ALY	25	30	750	01/01/08
3	SPEAKER	RAMY	80	10	800	01/01/08
5	MOUSE	RAMY	35	20	80	04/01/08
6	MOUSE	ALY	35	20	700	06/01/08
7	KEYBOARD	RAMY	80	10	800	08/01/08
8	MOUSE	KAREM	30	20	600	08/01/08

7 rows selected.

Between con1 and
con2

تستخدم للحصول على الحقول التي تحتوى على قيمة فارغة

Is Null


```
SQL> SELECT * FROM ITEMS
2 WHERE REVAL IS NULL;
```

ITEM_ID	ITEM_NAME	REVAL
2	MONITOR	

تستخدم للربط بين شرطين

```
1 SELECT * FROM SALES
2* WHERE ITEM = 'MOUSE' AND CUST = 'ALY'
SQL> R
1 SELECT * FROM SALES
2* WHERE ITEM = 'MOUSE' AND CUST = 'ALY'
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
6	MOUSE	ALY	35	20	700	06/01/08

```
SQL> SELECT * FROM SALES
2 WHERE ITEM = 'MOUSE' OR CUST = 'ALY';
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
1	MOUSE	TAMER	12	20	240	01/01/08
2	KEYBOARD	ALY	25	30	750	01/01/08
5	MOUSE	RAMY	35	20	80	04/01/08
6	MOUSE	ALY	35	20	700	06/01/08
8	MOUSE	KAREM	30	20	600	08/01/08

And – Or

تستخدم للحصول على البيانات بنفى الشرط الموجود

```
SQL> SELECT * FROM
2 SALES WHERE (PRIC=35 OR CUNT=10) AND CUST NOT LIKE 'RAMY';
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
4	SPEAKER	KAREM	250	10	2500	03/01/08
6	MOUSE	ALY	35	20	700	06/01/08

Not

فى حالة وجود Not – or – and فى جملة واحدة يتم تنفيذهم

بالترتيب التالى

Not (1) – And(2) – Or(3)

تستخدم لترتيب الحقول ترتيبا تصاعديا

Order By

تستخدم لترتيب الحقول ترتيب تنازلى

```
SQL> SELECT * FROM  
2 SALES WHERE CUST = 'RAMY' ORDER BY CUNT;
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
3	SPEAKER	RAMY	80	10	800	01/01/08
7	KEYBOARD	RAMY	80	10	800	08/01/08
5	MOUSE	RAMY	35	20	80	04/01/08

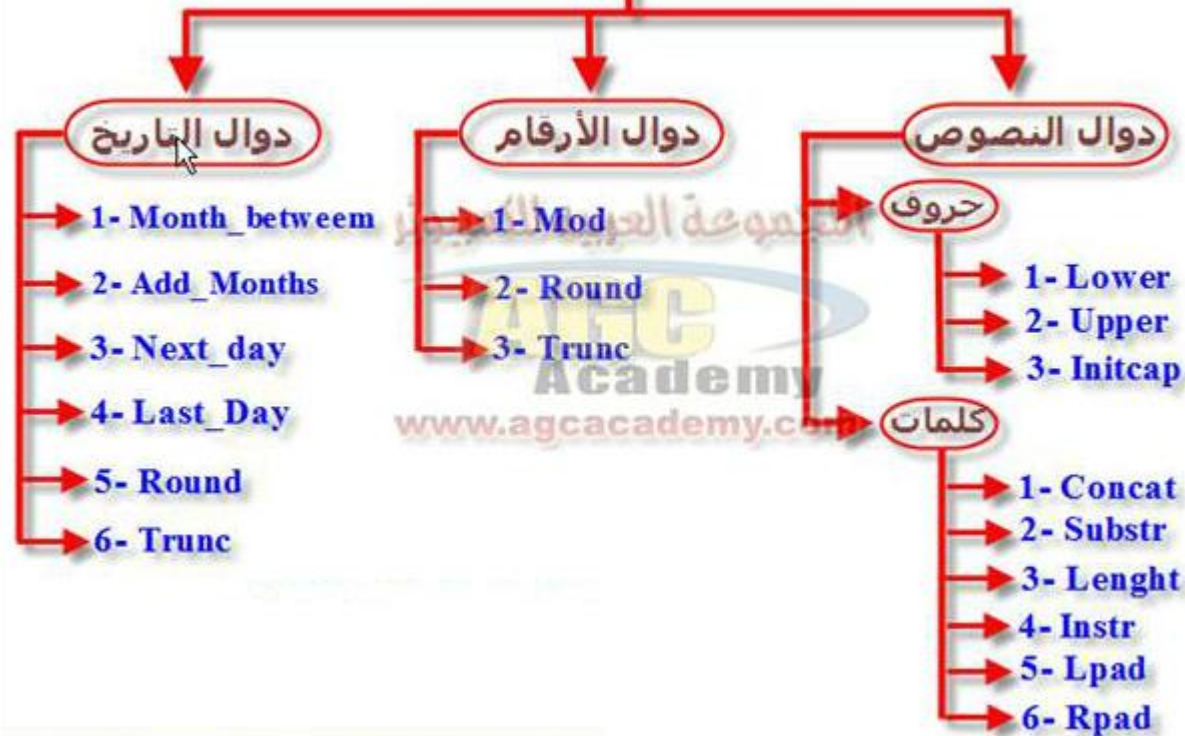
```
SQL> SELECT * FROM  
2 SALES ORDER BY PRIC DESC;
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
4	SPEAKER	KAREM	2500	10	2500	03/01/08
3	SPEAKER	RAMY	80	10	800	01/01/08
7	KEYBOARD	RAMY	80	10	800	08/01/08
5	MOUSE	RAMY	35	20	80	04/01/08
6	MOUSE	ALY	35	20	700	06/01/08
8	MOUSE	KAREM	30	20	600	08/01/08
2	KEYBOARD	ALY	25	30	750	01/01/08
1	MOUSE	TAMER	12	20	240	01/01/08

8 rows selected.

Order By ..field.. desc

SOL Functions



يستخدم لتحويل حالة الاحرف من كابيتال الى سمول

Lower

```
SQL> select lower (cust)
      2  from sales;
```

```
LOWER(CU
```

```
-----
tamer
aly
ramy
karen
ramy
aly
ramy
karen
```

لتحويل حالة الاحرف من اسمول الى كابيتال

```
SQL> select initcap (item) from
      2  sales ;
```

```
INITCAP(IT
```

```
-----
House
Keyboard
Speaker
Speaker
House
House
Keyboard
House
```

```
8 rows selected.
```

لتحويل الحرف الاول الى كابيتال وباقي الاحرف الى اسمول

تقوم بربط قيمة اكثر من حقل

Upper

InitCap

Concat

```
SQL> select concat(S_ID , ITEM)FROM
2 SALES;

CONCAT(S_ID,ITEM)
-----
1HOUSE
2KEYBOARD
3SPEAKER
4SPEAKER
5HOUSE
6HOUSE
7KEYBOARD
8HOUSE

8 rows selected.
```

لمعرفة عدد احرف الكلمات الموجودة داخل حقل معين

```
SQL> SELECT LENGTH(CUST) , CUST FROM
2 SALES;

LENGTH(CUST) CUST
-----
5 TAMER
3 ALY
4 RAYY
5 KAREM
4 RAYY
3 ALY
4 RAYY
5 KAREM

8 rows selected.
```

Length

تقوم باسترجاع عدد من الاحرف من حقل معين

Substr

```
SQL> SELECT SUBSTR ( ITEM,1,3) FROM
2 SALES;
```

SUB

MOU
KEY
SPE
SPE
MOU
MOU
KEY
MOU

تظهر موقع الحرف داخل حقل معين

```
SQL> SELECT INSTR(CUST,'A') FROM
2 SALES;
```

INSTR(CUST,'A')

2
1
2
2
2
1
2
2

8 rows selected.

Instr

يجعل عدد الاحرف المستخدمة متساوية داخل خلايا حقل معين مع
اضافة الاحرف الجديد بدءا من اليسار

Lpad

```
1 SELECT RPAD(ITEM,8,'$') FROM
2* SALES
```

```
RPAD(ITE
```

```
MOUSE$$$
KEYBOARD
SPEAKER$
SPEAKER$
MOUSE$$$
MOUSE$$$
KEYBOARD
MOUSE$$$
```

```
8 rows selected.
```

للتقريب الى اقرب رقم محدد

```
1 SELECT PRIC , ROUND ( PRIC,-1) FROM
2* SALES
```

```
PRIC ROUND(PRIC,-1)
```

```
-----
12          10
25          30
80          80
250        250
35          40
35          40
80          80
30          30
```

```
1 rows selected.
```

Round

لاظهار العدد طبقا لعدد الارقام المحددة دون تقريب

```
SQL> SELECT TRUNC (5869.975,2) FROM
2 DUAL;
```

```
TRUNC(5869.975,2)
-----
5869.97
```

Trunc

تسترجع باقى القيمة الناتجة من قسمة رقمين وليس ناتج القسمة

```
SQL> SELECT MOD (5,2) FROM
2 DUAL;
```

```
MOD(5,2)
```

Mod

تحسب عدد الشهور بين تاريخين

```
1 SELECT * FROM
2* SALES WHERE MONTHS_BETWEEN(SYSDATE,SDAT)<1
```

www.agcacademy.com

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
4	SPEAKER	KAREM	250	10	2500	03/01/08
5	MOUSE	RAMY	35	20	80	04/01/08
6	MOUSE	ALY	35	20	700	06/01/08
7	KEYBOARD	RAMY	80	10	800	08/01/08
8	MOUSE	KAREM	30	20	600	08/01/08

Months_between

لاضافة عدد من الشهور لقيمة حقل معين

```
SQL> SELECT ADD_MONTHS(SDAT,4) FROM
2 SALES;
```

```
ADD_MONT
```

```
01/05/08
01/05/08
01/05/08
03/05/08
04/05/08
06/05/08
08/05/08
08/05/08
```

```
8 rows selected.
```



Add_months

لليوم على تاريخ اليوم المحدد من التاريخ المذكور

Next_day

```

1 SELECT SDAT , NEXT_DAY(SDAT, 'الجمعة')
2* FROM SALES

```

SDAT	NEXT_DAY
------	----------

01/01/08	05/01/08
01/01/08	05/01/08
01/01/08	05/01/08
03/01/08	05/01/08
04/01/08	05/01/08
06/01/08	12/01/08
08/01/08	12/01/08
08/01/08	12/01/08

8 rows selected.

للحصول على اليوم الاخير فى الشهر المحدد

```

SQL> SELECT LAST_DAY(SDAT)
2 FROM SALES;

```

LAST_DAY

31/01/08
31/01/08
31/01/08
31/01/08
31/01/08
31/01/08
31/01/08
31/01/08

8 rows selected.

```

SQL> SELECT LAST_DAY('12/2/2015')FROM
2 DUAL;

```

LAST_DAY

28/02/15

Last_day

SQL Functions



تستخدم لتحويل القيمة الى قيمة نصية

To_char

```
SQL> SELECT TO_CHAR(SDAT, 'MON/YYYY')
2 FROM SALES;
```

```
TO_CHAR(SDA
```

```
٢٠٠٨/ أغسطس
٢٠٠٨/ يناير
٢٠٠٨/ يناير
٢٠٠٨/ يناير
٢٠٠٨/ يناير
٢٠٠٨/ يناير
٢٠٠٨/ يناير
٢٠٠٨/ يناير
```

```
9 rows selected.
```

تستخدم للتعامل مع القيم الفارغة واستبدالها بقيمة معينة

```
1 SELECT ITEM_NAME , NUL(REVAL,4)
2* FROM ITEMS
```

```
ITEM_NAME NUL(REVAL,4)
```

```
RAM 4
100
500 4
10
HOUSE 10
MONITOR 4
KEYBOARD 2
SPEAKER 5
```

```
8 rows selected.
```

Nvl

تستخدم للتعامل مع القيم الفارغة والغير فارغة واستبدال قيمهم بالقيم المحددة

Nval2

```

1 SELECT ITEM_NAME , NUL2(REVAL,111111,000000)
2* FROM ITEMS

ITEM_NAME  NUL2(REVAL,111111,000000)
-----
RAM
500
111111
111111
111111
111111
111111
111111
111111
8 rows selected.

```

تقارن بين قيمتين وفي حالة التساوى يتم وضع القيمة ب نل

```

1 SELECT ITEM_NAME ,REVAL, NULLIF(REVAL,10)
2* FROM ITEMS

ITEM_NAME      REVAL  NULLIF(REVAL,10)
-----
RAM
500
100
10
10
10
20
5
5
8 rows selected.

```

Nullif

التعامل مع القيم الفارغة لاكثر

Coalesce

```
SQL> SELECT COALESCE(REVAL,ITEM_ID,777777)
2 FROM ITEMS;

COALESCE(REVAL,ITEM_ID,777777)
-----
777777
100
777777
10
10
2
20
5

8 rows selected.
```

شبيهة بتركيبة case بلغات البرمجة

```
SQL> SELECT S_ID , ITEM ,PRIC, CASE ITEM WHEN 'MOUSE' THEN PRIC+2
2 WHEN 'SPEAKER' THEN PRIC + 5
3 ELSE PRIC+3 END CASE
4 FROM SALES;
```

S_ID	ITEM	PRIC	CASE
1	MOUSE	12	14
2	KEYBOARD	25	28
3	SPEAKER	80	85
4	SPEAKER	250	255
5	MOUSE	35	37
6	MOUSE	35	37
7	KEYBOARD	80	83
8	MOUSE	30	32

9 rows selected.

Case

نفس الدالة Case ولكن بصيغة مختلفة

Decode

```
SQL> SELECT S_ID , ITEM ,PRIC, DECODE (ITEM , 'MOUSE' ,PRIC+2
2  , 'SPEAKER' , PRIC + 5
3  , PRIC+3) NEWPRIC
4  FROM SALES;
```

S_ID	ITEM	PRIC	NEWPRIC
1	MOUSE	12	14
2	KEYBOARD	25	28
3	SPEAKER	80	85
4	SPEAKER	250	255
5	MOUSE	35	37
6	MOUSE	35	37
7	KEYBOARD	80	83
8	MOUSE	30	32

9 rows selected.





تستخدم لجمع قيمة حقل معين

SUM

تستخدم لمعرفة عدد سجلات حقل معين

COUNT

تستخدم لمعرفة المتوسط الحسابي

AVG

تستخدم لمعرفة اكبر عدد

MAX

تستخدم لمعرفة اصغر عدد

MIN

تستخدم لمعرفة الانحراف المعياري

VARIANCE



هو استعلام داخل الاستعلام الرئيسي ويمكن استخدامه بعد ويير او فروم او هافينج كما انه يمكن ان يعود بقيمة وحيدة او قيم متعددة

```
SQL> SELECT * FROM SALES
2 WHERE PRIC > (SELECT MAX ( CUNT) FROM SALES);
```

```
SQL> SELECT ITEM , MAX(CUNT)FROM SALES
2 GROUP BY ITEM
3 HAVING MAX(CUNT) > (SELECT MAX (CUNT) FROM SALES WHERE ITEM = 'MOUSE');
```

ITEM	MAX(CUNT)
KEYBOARD	30

```
SQL> SELECT * FROM (SELECT * FROM
2 SALES WHERE ITEM = 'MOUSE');
```

S_ID	ITEM	CUST	PRIC	CUNT	TOTL	SDAT
1	MOUSE	TAMER	12	20	240	01/01/08
5	MOUSE	RAMY	35	20	80	04/01/08
6	MOUSE	ALY	35	20	700	06/01/08
8	MOUSE	KAREM	30	20	600	08/01/08


```

1 SELECT ITEM , PRIC
2 FROM SALES
3* WHERE PRIC IN (SELECT MIN (PRIC) FROM SALES GROUP BY ITEM )

```

ITEM	PRIC
KEYBOARD	25
MOUSE	12
KEYBOARD	80
SPEAKER	80

```

SQL> SELECT ITEM , PRIC
2 FROM SALES
3 WHERE PRIC < ANY ( SELECT PRIC
4 FROM SALES
5 WHERE ITEM ='MOUSE' AND ITEM <> 'MOUSE';
WHERE ITEM ='MOUSE' AND ITEM <> 'MOUSE'

```

Insert into

إضافة صف جديد للجدول

المجموعة العربية للكمبيوتر

AGC

Insert Into table (اسم العمود , column name) اسم الجدول
Values (value , القيم المراد إدخالها في الجدول ,
);

```

SQL> INSERT INTO ITEMS ( ITEM_ID , ITEM_NAME )
2 SELECT ITEM_ID,ITEM_NAME FROM ITEMS WHERE ITEM_ID=6;

1 row created.

```


Update

التعديل على صف واحد في الجدول

المجموعة العربية للكمبيوتر

AGC Academy

Update table اسم الجدول
Set column اسم العمود = Value القيمة الجديدة

```
SQL> UPDATE SALES SET PRIC = PRIC * 1.1;
```

DELETE

الحذف داخل الجدول

المجموعة العربية للكمبيوتر

AGC Academy

Delete From table اسم الجدول
Where condition الشرط الذي يقوم بعملية الحذف من خلاله

```
SQL> DELETE FROM ITEMS  
2 WHERE ITEM_ID> 5;
```

الحفظ و التراجع



تستخدم لحفظ الاجراء عند نقطة معينة

SAVEPOINT

تستخدم لاسترجاع البيانات الى ما قبل تنفيذ الاجراء عند نقطة معينة

ROLLBACK

Database Object



Data Type

نوع البيانات	الوصف
VARCHAR2 (Size)	بيانات حرفية
CHAR (Size)	بيانات حرفية
NUMBER	بيانات رقمية
DATE	الوقت و التاريخ
LONG	بيانات حرفية تخزن أكثر من ٢ حيجا
ROW AND LONG ROW	صفوف بيانات أولية
BLOB	بيانات أولية أكثر من ٤ حيجا

```
SQL> CREATE TABLE SALES ( S_ID NUMBER(4),
2  ITEM VARCHAR2(10)
3  ,CUST VARCHAR2(8),
4  PRIC NUMBER(5),
5  CUNT NUMBER(3)
6  ,TOTAL NUMBER(5),
7  SDATE DATE );
```

Table created.

```
SQL> CREATE TABLE ITEMS1
2  AS SELECT ITEM_ID, ITEM_NAME
3  FROM ITEMS
4  WHERE ITEM_ID=1;
```

Table created.

```
SQL> ALTER TABLE ITEMS1
2  ADD(REVAL NUMBER(4));
```

Tables

```
SQL> ALTER TABLE ITEMS1
2 MODIFY ( REVAL CHAR(4));

Table altered.

SQL> DROP TABLE ITEMS1;

Table dropped.

SQL> RENAME ITEM TO ITEMS;

Table renamed.

SQL> ALTER TABLE SALES
2 ADD CONSTRAINT SALES PRIMARY KEY (S_ID );
```

```
SQL> create VIEW VIEW1
2 AS SELECT * FROM SALES
3 WHERE CUST='ALAA';

View created.
```

Views

تستخدم لعمل اجراء معين وغالبا ما يكون الترقيم التلقائي

Sequence

Create Sequence Sequence Name
 [Increment By n]
 [Start with n]
 [{Maxvalue n \ noMaxvalue}]
 [{Minvalue n \ noMinvalue}]
 [{Cycle \ noCycle}]
 [{Cache n \ noCache}]

Sequence

```
SQL> CREATE SEQUENCE AUTO_NUMBER
2 INCREMENT BY 1
3 START WITH 1
4 MAXVALUE 100;
```

Sequence created.

```
SQL> SELECT SEQUENCE_NAME,MIN_VALUE,MAX_VALUE,INCREMENT_BY,LAST_NUMBER
2 FROM USER_SEQUENCES;
```

SEQUENCE_NAME	MIN_VALUE	MAX_VALUE	INCREMENT_BY	LAST_NUMBER
AUTO_NUMBER	1	100	1	1

```
SQL> insert Into ITEMS( ITEM_ID,ITEM_NAME,PRIC )
2 values ( AUTO_NUMBER.NEXTVAL,&ITEM_NAME,&PRIC);
Enter value for item_name: 'MOUSE'
Enter value for pric: 20
old 2: values ( AUTO_NUMBER.NEXTVAL,&ITEM_NAME,&PRIC)
new 2: values ( AUTO_NUMBER.NEXTVAL,'MOUSE',20)
```

1 row created.

```
SQL> ALTER SEQUENCE AUTO_NUMBER
2 INCREMENT BY 1
3 MAXVALUE 1000;
```

Sequence altered.

```
SQL> DROP SEQUENCE AUTO_NUMBER;
```

Sequence dropped.

تستخدم لتسريع عملية البحث عن حقل معين داخل الجدول

```
SQL> CREATE INDEX ITM
2 ON ITEMS (ITEM_ID);
```

```
SQL> DROP INDEX ITM;
```

Index dropped.

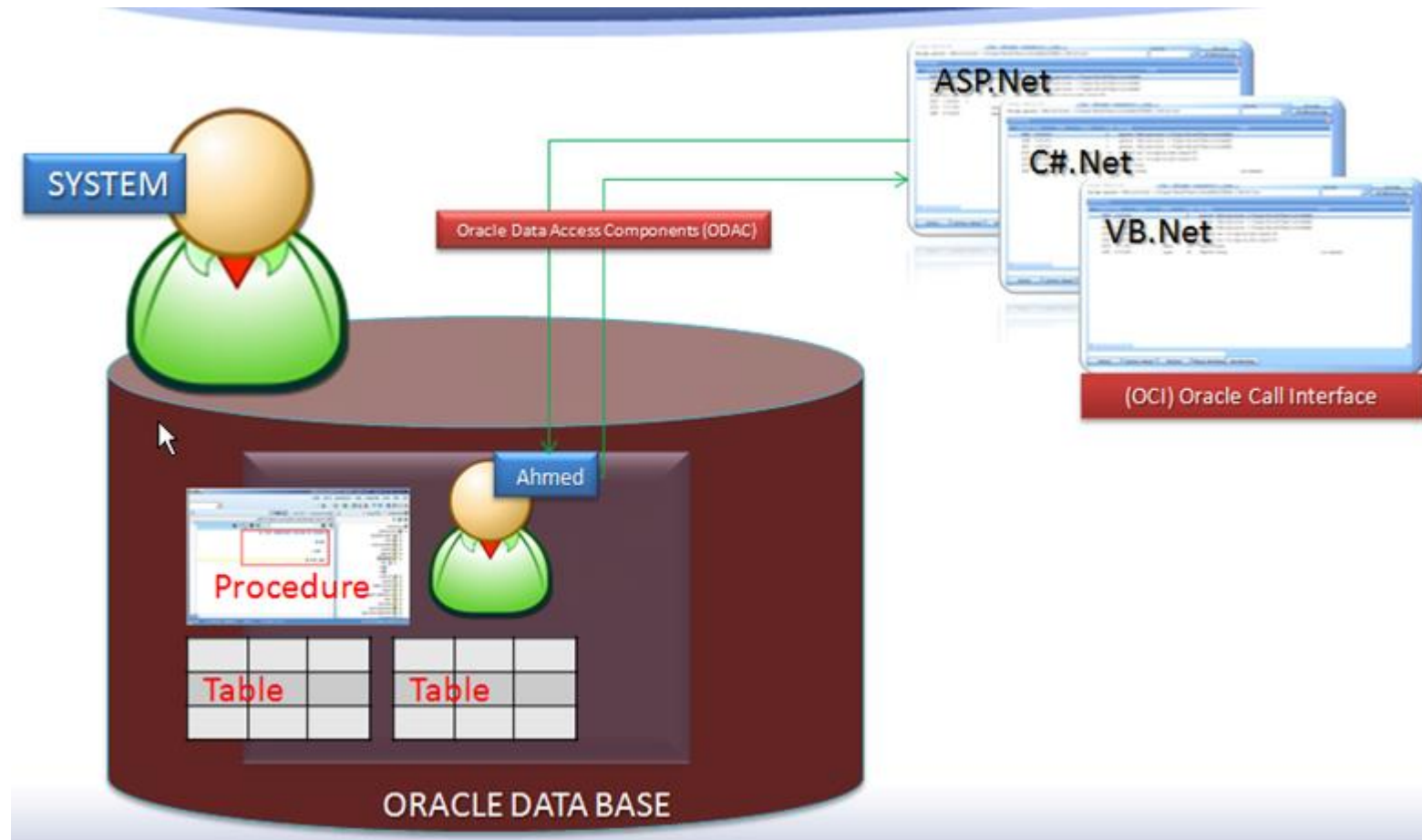
Indexes

تستخدم لانشاء اسم بديل لجدول معين

```
SQL> CREATE SYNONYM IT FOR ITEMS;
```

Synonym

Oracle Sql Developer Install




```

SQL> Create OR Replace Procedure      chk< A Number , B Number > AS
2
3   Z Number;
4
5   BEGIN
6
7   Z:= A + B;
8   DBMS_OUTPUT.PUT_LINE<Z>;
9
10  End chk;
11  /

```

Procedure created.

يتم كتابة الكود هنا

Oracle | Hardware and Software, Engineered to Work Together - Windows Internet Explorer

http://www.oracle.com/index.html

Edit View Favorites Tools Help

Oracle | Hardware and Software, Engineered to Work...

Welcome amr (Account | Help | Sign Out) United States Communities I am a...

Downloads Store Support Education Partners About

Databases

- Database 11g
- Database 10g Express Edition
- MySQL

Servers and Storage Systems

- Oracle Solaris
- Oracle Linux and VM
- Firmware
- See All

Developer Tools

- Oracle SQL Developer
- Developer and IDE
- Developer Tools for Visual Studio
- Enterprise Pack for Eclipse
- NetBeans IDE
- See All ...

Applications

- E-Business Suite

Popular Downloads

- Java for Developers
- Java for Your Computer
- JavaFX
- Oracle Solaris
- MySQL
- Fusion Middleware 11g
- Database 11g

Open Source

Oracle recognizes the power of open source software. We offer a single point of contact for all your open source needs. Learn more >

Clickability and Oracle Powering growth with MySQL

أولاً نقوم بالدخول هنا على موقع أوراكل

نانيا الضغط على تحميل Download

نانيا نقوم بالضغط على ORACLE SQL Developer

تسمية يجب القيام بتسجيل حساب لسيادتكم على الموقع من خلال الميل مجاناً


Create Table

Create Table

Schema: TAREK

Name: ITEMS


☐ Advanced









Table

DDL

Columns:

PK	Name	Data Type	Size	Not Null	Default	Comment
	ID	NUMBER		<input checked="" type="checkbox"/>		
	NAME	VARCHAR2	20	<input type="checkbox"/>		
	QUAN	VARCHAR2	20	<input type="checkbox"/>		
	DETAILS	VARCHAR2	20	<input type="checkbox"/>		



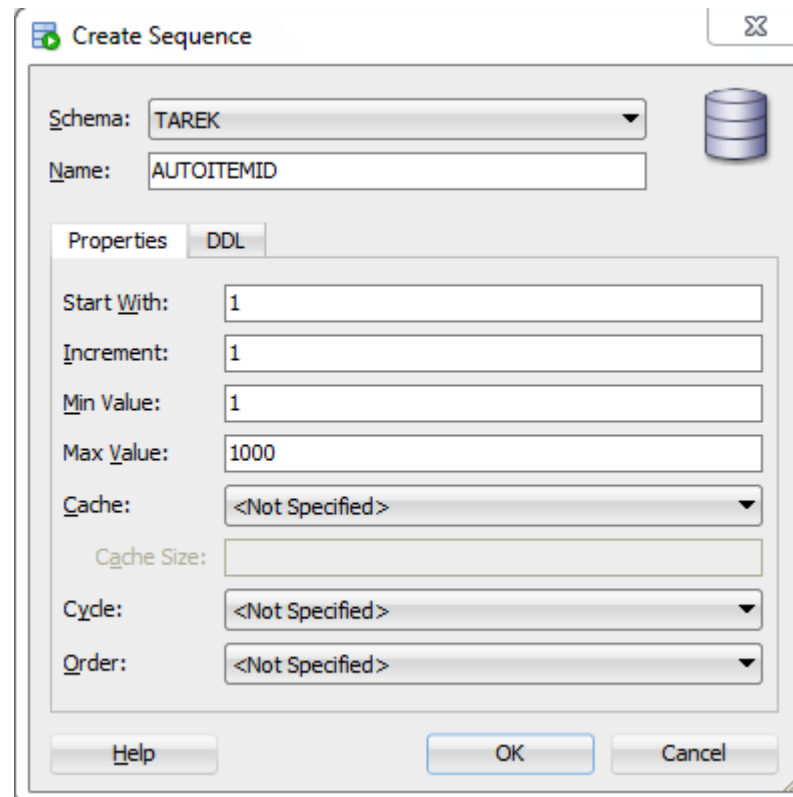


Help

OK

Cancel

Create Sequence



The image shows a 'Create Sequence' dialog box with a title bar containing a document icon and the text 'Create Sequence'. On the right side of the title bar is a close button (X). The dialog has two tabs: 'Properties' (selected) and 'DDL'. In the 'Properties' tab, there are two input fields at the top: 'Schema:' with a dropdown menu showing 'TAREK' and a database icon to its right, and 'Name:' with a text field containing 'AUTOITEMID'. Below these are several more fields: 'Start With:' (1), 'Increment:' (1), 'Min Value:' (1), 'Max Value:' (1000), 'Cache:' (a dropdown menu showing '<Not Specified>'), 'Cache Size:' (an empty text field), 'Cycle:' (a dropdown menu showing '<Not Specified>'), and 'Order:' (a dropdown menu showing '<Not Specified>'). At the bottom of the dialog are three buttons: 'Help', 'OK', and 'Cancel'.

Schema: TAREK

Name: AUTOITEMID

Properties DDL

Start With: 1

Increment: 1

Min Value: 1

Max Value: 1000

Cache: <Not Specified>

Cache Size:








Cycle: <Not Specified>

Order: <Not Specified>

Help OK Cancel

Create Procedure

Code | Grants | References | Dependencies | Errors | Profiles | Details

```
CREATE OR REPLACE PROCEDURE ADDITEM
(
    VNAME IN VARCHAR2
, VQUAN IN NUMBER
, VDETAILS IN VARCHAR2
) AS
BEGIN
    insert into ITEMS values(autoitemid.nextval, vname, vquan, vdetails);
    COMMIT;
END ADDITEM;
```

Calling procedure from vb.net

```
Public Class Form1

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1

        Dim cn As New ADODB.Connection
        cn.ConnectionString = "Driver={Microsoft ODBC for ORACLE};Server=myagc; UID=Ahmed ; PWD=123456;"
        cn.Open()

        Dim S As String
        S = "Begin NewItem('" & Item.Text & "',''" & Vendor.Text & "',''" & Phone.Text & "',''" & Email.Text

        cn.Execute(S)

    End Sub

End Class
```

Books Example

Steps

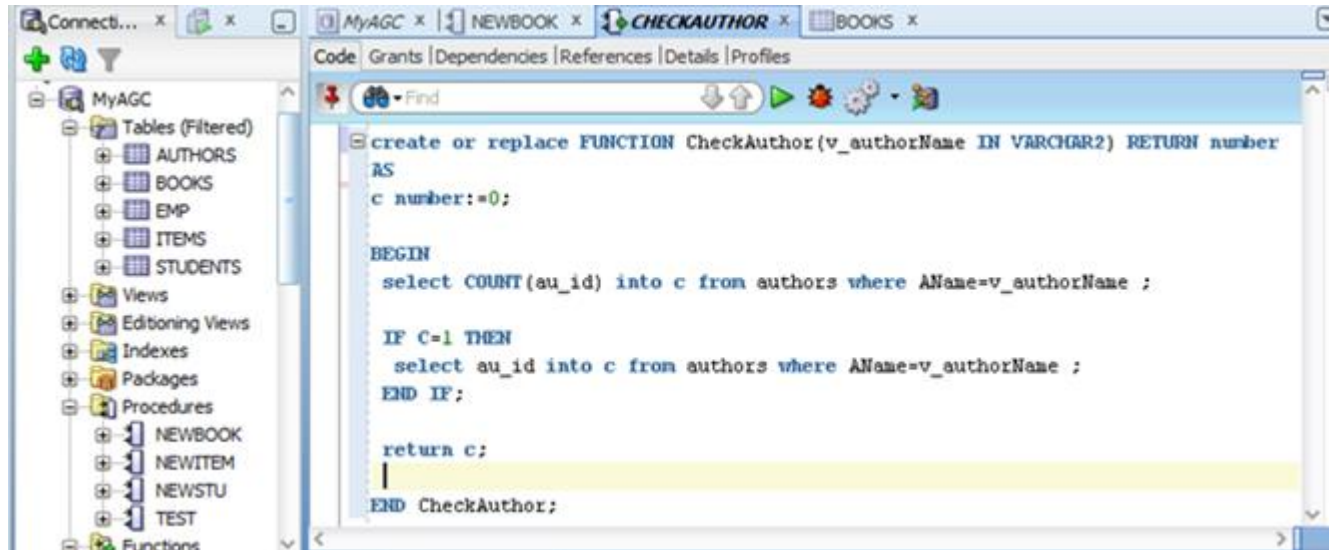
- 1- Create Authors Table.
- 2- Create Books Table.
- 3- Create CheckAuthor Function.
- 4- Create NewBook Procedure.

Books				
ISBN	Title	AuthorID	Pages	Lang
111	OOP	1	200	eng
222	ADO	1	240	eng
333	HTML	2	120	eng
444	Java	1	300	eng
555	ASP	2	250	eng

Authors	
Au_ID	aName
1	Smith
2	Mike

ChkAuthor Function

الهدف من هذه الدالة التأكد من وجود الكاتب بجدول المؤلفين من خلال البحث بدلالة اسم الكاتب والعودة برقمه ان وجد او بالقيمة 0 في حالة عدم وجوده



The screenshot shows a database IDE with a project named 'MyAGC'. The left sidebar displays a tree view of database objects: Tables (Filtered) including AUTHORS, BOOKS, EMP, ITEMS, and STUDENTS; Views; Editing Views; Indexes; Packages; Procedures; and Functions. The main editor window is titled 'CHECKAUTHOR' and shows the following SQL code:

```
create or replace FUNCTION CheckAuthor(v_authorName IN VARCHAR2) RETURN number
AS
c number:=0;

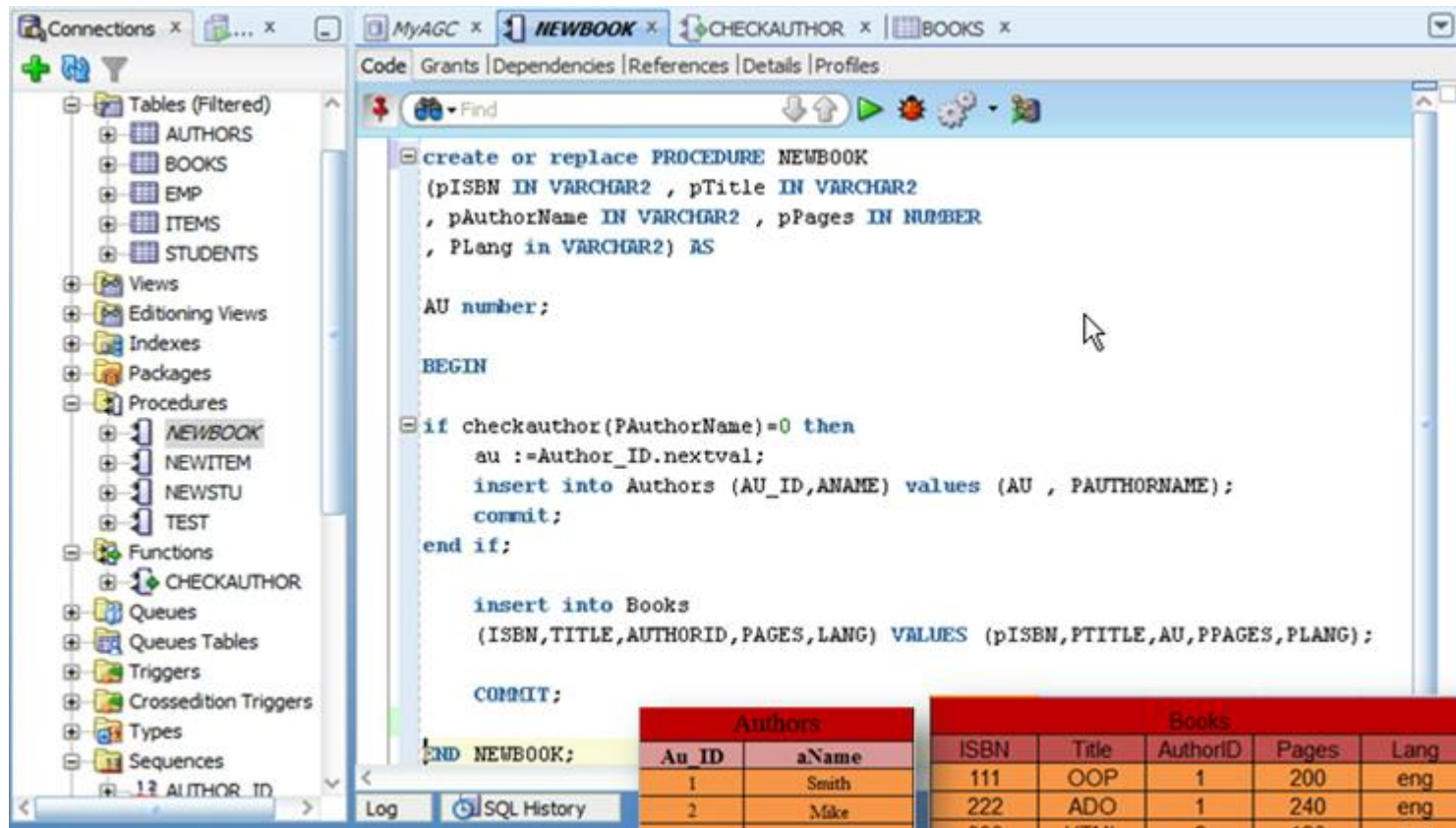
BEGIN
select COUNT(au_id) into c from authors where AName=v_authorName ;

IF C=1 THEN
select au_id into c from authors where AName=v_authorName ;
END IF;

return c;
END CheckAuthor;
```

NewBook Procedure

الهدف من هذا الاجراء هو اضافة كتاب جديد فى جدول الكتب مع التاكيد من وجود الكاتب فى جدول المؤلفين وفى حالة عدم وجوده يتم اضافته فى الجدول وذلك من خلال الدالة chkauthor



The screenshot displays a database IDE with the following components:

- Left Panel (Object Explorer):** Shows a tree view of database objects including Tables (Filtered), Views, Editing Views, Indexes, Packages, Procedures, Functions, Queues, Queues Tables, Triggers, Crossedition Triggers, Types, and Sequences. The 'NEWBOOK' procedure is selected under the 'Procedures' folder.
- Code Editor:** Contains the SQL code for the 'NEWBOOK' procedure. The code is as follows:

```
create or replace PROCEDURE NEWBOOK
(pISBN IN VARCHAR2 , pTitle IN VARCHAR2
, pAuthorName IN VARCHAR2 , pPages IN NUMBER
, Plang in VARCHAR2) AS

AU number;

BEGIN

if chkauthor(PAuthorName)=0 then
    au :=Author_ID.nextval;
    insert into Authors (AU_ID,ANAME) values (AU , PAUTHORNAME);
    commit;
end if;

insert into Books
(ISBN,TITLE,AUTHORID,PAGES,LANG) VALUES (pISBN,PTITLE,AU,PPAGES,PLANG);

COMMIT;

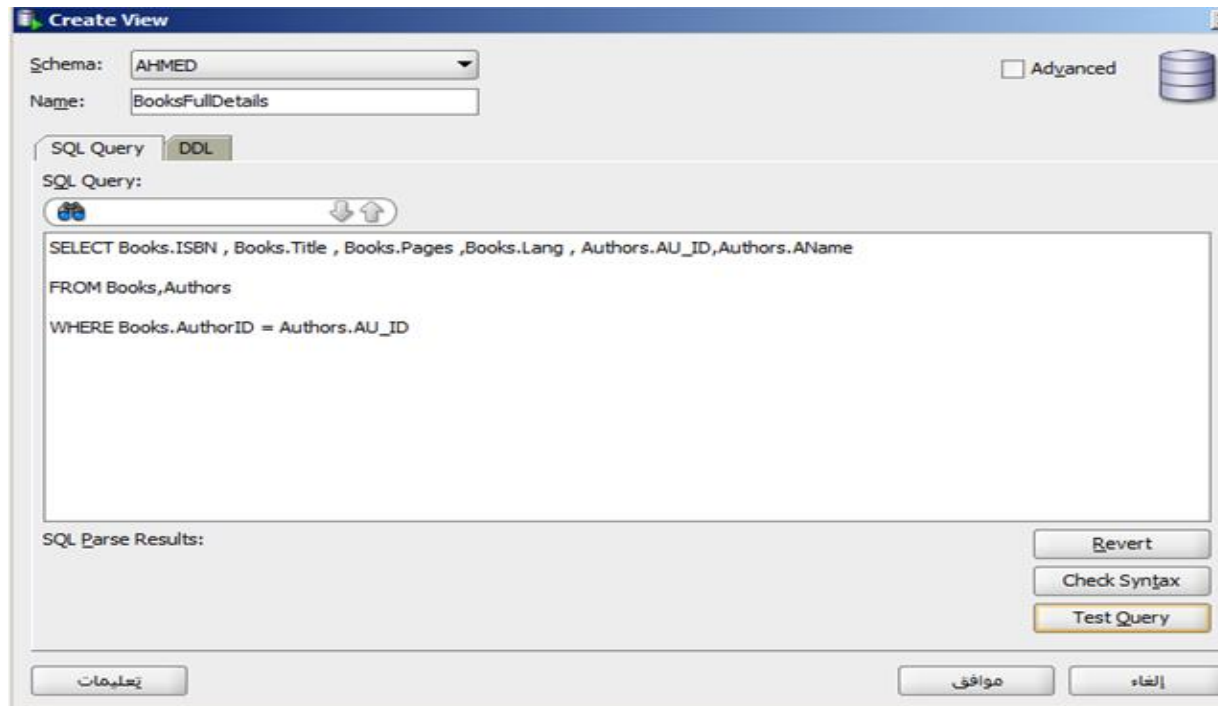
END NEWBOOK;
```
- Bottom Panel:** Displays two data tables: 'Authors' and 'Books'.

Authors	
Au ID	aName
1	Smith
2	Mike

Books				
ISBN	Title	AuthorID	Pages	Lang
111	OOP	1	200	eng
222	ADO	1	240	eng

BookFullDetails View

الهدف منه هو عمل استعلم يعرض بيانات الكتاب واسم ورقم المؤلف من خلال جدولى الكتب والمؤلفين



Schema: AHMED

Name: BooksFullDetails

☐ Advanced

SQL Query

SQL Query:

```
SELECT Books.ISBN , Books.Title , Books.Pages ,Books.Lang , Authors.AU_ID,Authors.AName
FROM Books,Authors
WHERE Books.AuthorID = Authors.AU_ID
```

SQL Parse Results:

Revert

Check Syntax

Test Query

إعلامات

موافق

إلغاء

Cursor

هو مؤشر يحتوى على عدد من المتغيرات بناءا على شرط معين ويمكن ان ياخذ شكل الجدول او بعض الحقول من الجدول ويكون موجود بالذاكرة

تعريف المتغيرات فى الذاكرة

تعريف المؤشر فى الذاكرة

فتح المؤشر

```
DECLARE
v_ID number;
v_NameItem Items.NameItem%Type;
v_Vendor ITEMS.VENDOR%Type;
v_Phone ITEMS.PHONE%Type;
v_Email ITEMS.EMAIL%Type;

CURSOR C_NameItem IS SELECT NameItem,Vendor,Phone,Email FROM Items;

BEGIN
OPEN C_NameItem;

LOOP
FETCH C_NameItem INTO V_NameItem,V_Vendor,V_Phone,V_Email;
Exit When C_NameItem%notfound;

DBMS_OUTPUT.PUT_LINE('Name Item : ' || V_NameItem);
DBMS_OUTPUT.PUT_LINE('Vendor : ' || V_Vendor);
DBMS_OUTPUT.PUT_LINE('Phone : ' || V_Phone);
DBMS_OUTPUT.PUT_LINE('Email : ' || V_Email);
DBMS_OUTPUT.PUT_LINE('-----');

END LOOP;

CLOSE C_NameItem;
END;
```

Name Item : aaa
Vendor : sss
Phone : ddd
Email : ffff

Name Item : IPhone
Vendor : CompuMe
Phone : 1234556
Email : aarmousa@agcit.net

Name Item : SoftWare
Vendor : AGC Academy
Phone : 98765
Email : video@agcit.net

Name Item : Galaxy
Vendor : AGC Academy
Phone : 9343434
Email : ahmed@agcit.net

0.03 seconds

```

DECLARE
v_ID number;
v_NameItem Items.NameItem%Type;
v_Vendor ITEMS.VENDOR%Type;
v_Phone ITEMS.PHONE%Type;
v_Email ITEMS.EMAIL%Type;

CURSOR C_NameItem IS SELECT NameItem,Vendor,Phone,Email FROM Items;

```

تعريف المؤشر في الذاكرة

C_NameItems

ID	NAMEITEM	VENDOR	PHONE	EMAIL
----	----------	--------	-------	-------

ID	NAMEITEM	VENDOR	PHONE	EMAIL
1	9 aaa	sss	ddd	fff
2	1 iPhone	CompuMe	1234556	amrmousa@agcit.net
3	2 SoftWare	AGC Academy	98765	video@agcit.net
4	3 Galaxy	AGC Academy	9343434	ahmed@agcit.net

SQL Control Statement

If .. Then

```
DECLARE
  i NUMBER:=15;
BEGIN
  IF i>10 THEN
    dbms_output.put_line('The Number >10 ' || i);
  ELSE
    dbms_output.put_line('The Number <10 ' || i);
  END IF;
END;
```

optarek35210

Loop .. End Loop

```
DECLARE
  i NUMBER(5);
BEGIN
  i:=1;
  LOOP
    IF i>4 THEN
      EXIT;
    END IF;
    DBMS_OUTPUT.PUT_LINE('i = ' || i);
    i:=i+1;
  END LOOP;
END;
```

```

DECLARE
  i NUMBER:=0;
BEGIN
  i:=1;
  LOOP
    EXIT WHEN i>4;
    DBMS_OUTPUT.PUT_LINE('i = ' || i);
    i:=i+1;
  END LOOP;
END;

```

While ..

```

WHILE i <= 4
LOOP
  DBMS_OUTPUT.PUT_LINE('i = ' || i);
  i:=i+1;
END LOOP;
END;

```

```

DECLARE
  i NUMBER:=0;
BEGIN
  FOR i IN 1..5
  LOOP
    DBMS_OUTPUT.PUT_LINE('i = ' || i);
  END LOOP;
END;

```

تم بحمد الله