

## الفصل السادس استخدام الدوال Built-in function

الدوال في SQL عبارة عن برامج صغيرة تمكنك من أداء الكثير من الوظائف التي تحتاجها أثناء تعاملك مع قواعد البيانات ولذلك فالعبارة Built-in Functions تعني أن هذه الدوال موجودة داخل SQL وكل ما عليك أنك تستدعيها للتنفيذ. سوف نتعلم في هذا الفصل:

- ◆ ما هي الدوال Built-In Functions.
- ◆ الدوال الإحصائية.
- ◆ دوال التاريخ والوقت.
- ◆ الدوال الرياضية (الحسابية).
- ◆ دوال تقريب الأرقام.
- ◆ دوال التعامل مع الحروف.
- ◆ دوال متنوعة.

تستخدم معظم لغات البرمجة الدوال الجاهزة أو **Built-in Functions** وهي عبارة عن برامج جاهزة يتم استدعائها لأداء وظيفة معينة ولذلك يتم ترجمتها أحيانا ب"الوظائف الجاهزة" وهي تسهل التعامل مع البيانات وتمكنك من الحصول علي النتائج بسرعة كما في حالة الرغبة في الحصول علي مجموع السجلات أو متوسطاتها أو تحويل الحروف الصغيرة إلي حروف كبيرة... الخ .

فيما يلي سوف نقسم الدوال ألي مجموعات متشابهة ليسهل عليك دراستها وتذكرها .

### الدوال الإحصائية :

يمكنك استخدام جملة **SELECT** لتنفيذ بعض العمليات الحسابية علي بيانات الجدول باستخدام بعض الدوال **Functions** يوضح الجدول التالي الدوال الإحصائية التي يمكن استخدامها مع جملة **SELECT** . ووظيفة كل منها :

الدالة	وظيفتها
<b>AVG</b>	تحسب المتوسط الحسابي لمجموع القيم الموجودة في حقل معين . فإذا اشتملت جملة <b>SELECT</b> علي الاختيار <b>WHERE</b> فإنها تحسب المتوسط الحسابي للسجلات التي ينطبق عليها الشرط الموجود في الاختيار <b>WHERE</b> فقط
<b>COUNT</b>	تحسب عدد السجلات في الجدول أو عدد السجلات التي ينطبق عليها الشرط المحدد بالاختيار <b>WHERE</b>
<b>MIN</b>	تحسب القيمة الصغرى لمجموعة قيم موجودة في حقل معين أو تلك التي ينطبق عليها الشرط <b>WHERE</b>
<b>MAX</b>	تحسب القيمة الكبرى لمجموعة قيم موجودة في حقل معين أو تلك التي ينطبق عليها الشرط <b>WHERE</b>
<b>SUM</b>	تحسب مجموع القيم الموجودة في حقل معين أو تلك التي ينطبق عليها الشرط <b>WHERE</b>

وظيفتها	المدالة
تستخدم للحصول علي الانحراف المعياري لبيانات حقل في جميع السجلات أو تلك التي ينطبق عليها الشرط WHERE	STDDEV

المثال التالي يحسب القيمة الصغرى والكبرى ومتوسط مرتبات الموظفين في جدول

Personal انظر شكل ٦ - ١

use employee ;

Database changed

select \* from personal ;

select min <salary>minimum, max<salary> maximum,  
avg<salary> average from personal;

Select \* from personal;

ahmed	mohammad	m	27	3500	2717572	2001	1999
israa	mohammad	f	29	4300	22334453	2000	1995
medo	yakoub	m	55	9999	2901346	2001	1971
amir	fahmy	m	33	4520	1122334	2002	1985
badr	moon	m	43	5400	9876543	1999	1979
kareem	ramadan	m	37	5600	8877665	1998	1975
farida	samir	f	28	3300	6734521	2003	1997

select min <salary> minimum, max <salary> maximum,  
avg<salary> average from personal;

Minimum	maximum	average
3300	9999	5231-2857

شكل ٦-١

المثال التالي يمكنك من الحصول على مجموع حقل salary

select sum ( salary ) total from personal ;

total  
27500

## دوال التاريخ والوقت :

فيما يلي نشرح الدوال التي تتعامل مع التاريخ والوقت ولا يستغنى أي مبرمج في

أي نظام عن الحاجة لمعرفة الوقت و التاريخ الحاليان .أو إضافة أو طرح يوم أو شهر أو مدة

زمنية لحقل تاريخي للحصول على نتائج معينة . تابع معنا الشرح التالي :

### الدالة Last\_Day

هل تعرف كيف تحسب السنة الكبيسة؟ وهل تعرف عدد الأيام في كل شهر من الأشهر الميلادية ومتى يكون فبراير ٢٨ ومتى يكون ٢٩ يوم؟ وما هي الشهور ذات الثلاثين يوماً والأخرى ذات الواحد وثلاثون يوماً؟ تستطيع باستخدام query في Oracle أن تعرف آخر يوم في الشهر لأي شهر تريد باستخدام الوظيفة Last\_day ، أما بيني وبينك فان السنة الكبيسة هي التي تقبل القسمة على ٤ مثل سنة ٢٠٠٤ فهي تقبل القسمة على ٤ دون باق ، هذه السنة يكون عدد أيامها ٣٦٦ يوماً لأن شهر فبراير فيها ٢٩ يوماً .

SELECT \* FROM TASKS;

task	startdate	enddate
delivery	01-APR-2004	28-APR-2004
installaton	29-APR-2004	15-MAY-2004
testing	01-MAY-2004	15-JUN-2004
training	01-JUL-2004	31-JUL-2004
OJT	01-AUG-2004	31-DEC-2004
handover	05-JAN-2005	20-JAN-2005

نحاول الحصول الآن على آخر يوم في كل شهر من حقل startdate ولنر النتيجة :

SELECT STARTDATE, LAST\_DAY(STARTDATE) LAST  
FROM TASKS;

startdate	last
01-Feb-2004	29-Feb-2004
29-Apr-2004	30-Apr-2004
01-May-2004	31-May-2004
01-Jul-2004	31-Jul-2004
01-Aug-2004	31-Aug-2004
05-Jan-2005	31-Jan-2005
16-Feb-2005	28-Feb-2005

نلاحظ أن فبراير ٢٠٠٤ عبارة عن ٢٩ يوم بينما فبراير ٢٠٠٥ هو ٢٨ يوم فقط

### الدالة Months\_Between

تقوم بعملية طرح بين حقلين من حقول التاريخ والنتيجة تكون عدد الأشهر ما بين

الحقلين ، مثال :

```
SELECT TASK, STARTDATE,
        ENDDATE, MONTHS_BETWEEN(ENDDATE, STARTDATE)
        DURATION FROM TASKS;
```

TASK	STARTDATE	ENDDATE	DURATION
ONE	15-MAY-2004	12-NOV-2004	6

*إضافة تاريخ أو شهر للتوقيت ADD\_MONTHS/ ADD\_DATE*

هي في Oracle الوظيفة ADD\_Months بينما في mysql تكتب هكذا

date\_add نفترض أننا نريد إضافة شهرين لتاريخ انتهاء المهمة Task Enddate فلنبدأ بكتابة query هكذا:

```
SELECT TASK, STARTDATE, ENDDATE ORIGINAL,
        2 ADD_MONTHS(ENDDATE,2) NEW,
        3 FROM TASKS,
```

TASK	STARTDATE	ORIGINAL	NEW
DELIVERY	1-APR-2004	28-APR-2004	28-JUN-2004
INSTALLATON	29-APR-2004	15-MAY-2004	15-JUL-2004
TESTING	1-MAY-2004	15-JUN-2004	15-AUG-2004
TRAINING	1-JUL-2004	31-JUL-2004	31-SEP-2004
OJT	1-AUG-2004	31-DEC-2004	28-FEB-2005
HANDOVER	5-JAN-2005	20-JAN-2005	2-MAR-2004

*دالة إظهار تاريخ اليوم*

تاريخ اليوم أو ما يطلق عليه Current Date نحتاجه غالباً ليكون جزء من أي

تقرير تتم طباعته من قواعد البيانات ، ويتم تخزين التاريخ عند تثبيت نظام قواعد البيانات

لأول مرة ويقوم برنامج التثبيت بأخذ التاريخ من نظام التشغيل الذي يتم تثبيت نظام

قواعد البيانات عليه مثل Windows, Unix, Linux ولأخذ أمثلة تاريخ اليوم من

ثلاثة من قواعد البيانات :

**SELECT SYSDATE FROM DUAL**

النتيجة

08-FEB-2005

**الدوال الرياضية (الحسابية) :**

تقوم الدوال الرياضية بتنفيذ بعض العمليات الحسابية علي بيانات الجدول ، وفيما يلي نوضح كيفية التعامل مع أهم هذه الوظائف :

**القيمة المطلقة (ABS) Absolute Values**

تستخدم الدالة ( ABS ) للحصول على القيمة المطلقة لحقل معين ، مثال :

```
select * from numbers;
```

x	y
3.1415	4
-45	-787
5	9
-67.667	42
15	55
-7.2	5.3

نحاول الآن أن نحصل على القيمة الحقيقية (المطلقة) للحقل x

```
SELECT ABS(X) "Absolute_Value"
FROM NUMBERS;
```

```
Asolute_Value
```

```
-----
3.1415
  45
  5
67.667
  15
  7.2
```

**دوال تقريب الأرقام CEIL and Floor**

الوظيفة الأولى CEIL تقوم بالتقريب الى أقرب رقم صحيح ، بينما الوظيفة الثانية

Floor تقوم بالتقريب الى أكبر رقم صحيح أي أنها تقريبا تحذف الكسور دون تقريب .

والآن لنجرب الوظيفتين CEIL and FLOOR ونر النتيجة :

Select x, CEIL(x) ceiling, FLOOR(x) floor from numbers;

x	Ceiling	Floor
3.1415	4	3
-45	1	-45
5	5	5
-67.667	1	-68
15	15	15
-7.2	1	-8

دالة باقي القسمة MOD

تعطى هذه الدالة الباقي بعد قسمة الرقم الأول على الرقم الثاني . المثال التالي يظهر

بيانات حقلي X,Y من جدول NUMBERS ثم يظهر باقي قسمة الأول علي الثاني :

SELECT X, Y, MOD(X, Y) reminder FROM NUMBERS;

x	y	Reminder
03.1415	4.000	3.1415
-45.0000	0.707	0.2480
05.0000	9.000	5.0000
-67.6670	42.000	16.3330
15.0000	55.000	15.0000
-7.2000	5.300	3.4000

دالة الجذر التربيعي SQRT

نلاحظ أن استخراج الجذر التربيعي يتم على الأرقام الموجبة فقط ولا يتم على القيم

السالبة ، لهذا سوف نحاول استخراج الجذر التربيعي لكل من x و y كالتالي:

SELECT \*, SQRT(ABS(X)) FROM NUMBERS;

هنا استخدمنا ABS(X) لكي نتلافى مشكلة القيم السالبة . والنتيجة هي :

x	SQRT (ABS(X))
3.1415	1.77242771
-45	6.70820393
5	2.23606798
-67.667	8.22599538
15	3.87298335
-7.2	2.68328157

**الدالة EXP:**

ترفع رقم إلي أس :

**SELECT X, EXP(X) EXP FROM NUMBERS;**

x	EXP
3.1415	23.1385
-45	2.9E-20
5	148.413
-67.667	4.1E-30
15	3269017
-7.2	0.00075

**حوال التعامل مع الحروف**

**الدالة CHR**

تعطي الحرف المقابل لكوند ASCII في الجدول الشهير ASCII المستخدم في معظم

الحواسب ، لنأخذ مثالا :

**SELECT \* FROM LETTERS;**

FIRTSNAME	LASTNAME	LTR	CODE
AHMED	ALI	A	32
ASSAD	HELMY	J	67
RANIA	MOKHTAR	C	65
HAJAR	MOHAMMAD	M	87
HEBA	MOHAMMAD	A	77
ISRAA	MOHAMMAD	G	52

الحقل CODE عبارة عن قيم لحروف ، تعال معي نكتشف هذه الحروف :

**SELECT CODE, CHR(CODE) FROM LETTERS;**

CODE	CHR(CODE)
32	
67	C
65	A
87	W
77	M
52	4

نلاحظ أن القيمة ٣٢ مقابلها شاغر ، في الحقيقة فإن قيمة ٣٢ هي مسافة Space أو لنقل

. Blank

### دالة ضم الحقول CONCAT

لكي نضم محتويات حقلين ليكونا حقلا واحدا يضم القيمتين بفواصل أو بدون فاصل فاننا نستخدم الدالة CONCAT وتستخدم نفس الاستخدام في Oracle SQL و أيضا Mysql، حاول التدقيق في الصياغة حيث سوف نضم الحقلين ثم سنضع في المرة التالية مسافة بين الحقلين ثم في المرة الأخيرة سنضع فاصلة بين الحقلين. المثال التالي يدمج بيانات حقل FIRSTNAME وحقل LASTNAME ويضع الناتج في عمود باسم

CONCATENATED

```
SELECT firstname, lastname,  
CONCAT(FIRSTNAME, LASTNAME) concatenated  
FROM LETTERS;
```

FIRSTNAME	LASTNAME	Concatenated
AHMED	ALI	AHMEDALI
ASSAD	HELMY	ASSADHELMY
RANIA	MOKHTAR	RANIAMOKHTAR
HAJAR	MOHAMMAD	HAJARMOHAMMAD
HEBA	MOHAMMAD	HEBAMOHAMMAD
ISRAA	MOHAMMAD	ISRAAMOHAMMAD

وفيما يلي ندمج بيانات الحقلين بعد وضع مسافة بينهما :

```
SELECT firstname, lastname,  
CONCAT(FIRSTNAME," ", LASTNAME) concatenated  
FROM LETTERS;
```

FIRSTNAME	LASTNAME	Concatenated
AHMED	ALI	AHMED ALI
ASSAD	HELMY	ASSAD HELMY
RANIA	MOKHTAR	RANIA MOKHTAR
HAJAR	MOHAMMAD	HAJAR MOHAMMAD
HEBA	MOHAMMAD	HEBA MOHAMMAD
ISRAA	MOHAMMAD	ISRAA MOHAMMAD

أما المثال التالي فيستخدم فاصلة بين الحقلين بعد دمجهما :

```
SELECT firstname, lastname,
CONCAT(FIRSTNAME," ", LASTNAME) concatenated
FROM LETTERS;
```

FIRTSNAME	LASTNAME	Concatenated
AHMED	ALI	AHMED,ALI
ASSAD	HELMY	ASSAD,HELMY
RANIA	MOKHTAR	RANIA,MOKHTAR
HAJAR	MOHAMMAD	HAJAR,MOHAMMAD
HEBA	MOHAMMAD	HEBA,MOHAMMAD
ISRAA	MOHAMMAD	ISRAA,MOHAMMAD

دالة تحويل الحروف كلها الى حروف كبيرة UPPER

```
SELECT FIRSTNAME, UPPER(FIRSTNAME) FROM LETTERS;
```

Firtsname	UPPER (FIRTSNAME)
Ahmed	AHMED
Assad	ASSAD
Rania	RANIA
Hajar	HAJAR
Heba	HEBA
Israa	ISRAA

دالة تحويل الحروف كلها الى حروف صغيرة LOWER

```
SELECT FIRSTNAME, LOWER(FIRSTNAME) FROM LETTERS;
```

FIRTSNAME	LOWER(FIRSTNAME)
AHMED	ahmed
ASSAD	assad
RANIA	rania
HAJAR	hajar
HEBA	heba
ISRAA	israa

الدالة SUBSTR

تستخدم لاستخراج جزء من عبارة معينة . ولا بد أن تحدد لها ثلاثة معاملات الأول العبارة النصية التي ستستخرج جزء منها ، والثاني الحرف الذي ستبدأ من عنده ، والثالث عدد الأحرف التي ستستخرجها .

المثال التالي يستخرج ثلاثة حروف فقط من حقل الاسم الأول ابتداء من الحرف الثاني في الحقل

```
SELECT FIRSTNAME , SUBSTR ( FIRSTNAME,2,3 )  
FROM LETTERS
```

FIRSTNAME	SUB
Fahmy	ahm
Mustafa	ust
Husny	usn
Fuad	uad

