Oracle date functions

Oracle provides a data type date that behaves like a timestamp. It stores a date and a time. Internally a date is stored in a special format not visible to the user. A user can store and retrieve dates and times through the following main functions:

```
to_date(str, format)
```

> `tr` is a sting of characters and `format` indicates how the string is supposed to be interpreted. The format components are listed in the following table. The function returns the internal representation of a date specifies by the string in the given format.

```
to_char(date, format)
```

> `date` is the internal representation of an Oracle date and `format` is as before. The function returns a character string representing the given date in the given format.

The format for a date is a string that may include as a substring one or more of the following:

Format	Description	Example
D	Day - number of day in the week	5
DD	Day – number of day within the month	26
DDD	Day – number of day in the year	235
DY	Day – three letter	SUN
DAY	Day – full name	SUNDAY
MM	Month - number	10
MON	Month - three letters	FEB
MONTH	Month - full name	FEBRUARY
Y	Year - last digit	3
YY	Year - last two digits	03
YYY	Year - last three digits	003
YYYY	Year - four digits	2003
HH12	Hour in 1-12 format	11
HH24	Hour in 0-24 format	19
MI	Minutes	54
SS	Seconds	45
AM	Displays AM or PM depending on the time	PM

Example

Suppose we have defined a table `Order(cid, item, quantity, odate)` which shows that at **odate** customer **cid** placed an order for quantiny many **items**. The query:

```
1  select
2  {
3    item to_char(odate, 'DD-MON-YY)
4       as order_date, to_char(odate, 'HH24:MI)
5          as order_time
6  }
7  from Order
8  where sid = 123456
```

may produce

item	order_date	order_time
pencil-BB	20-JAN-03	14:25
copier paper	14-FEB-03	18:10
...	...	...

Other functions

The following list contains most of the popular functions you can use with Oracle dates:

`date sysdate()` returns the current date and time

`date + int`, `date - int` You can add or subtract a number of days to a date to get that new date

`date1 - date2` returns the number of days between date 1 and date 2

`date next_day(date dat, string day)` Parameters: a date dat and a day of the week. It returns the date after dat whose day of the week is day

`date last_day(date dat)` returns the date that corresponds to the last day of the month in dat

`int months_between(date d1, date d2)` returns the number of the month between d1 and d2

`date least( date d1, date d2, ...., date dn)` returns the earliest of the given dates

`date greatest( date d1, date d2, ...., date dn)` returns the latest of the given dates

`date trunc(date dat)` returns the same date but the time is set to 12:00AM

`date round(date dat)` if dat is before 12:00 noon, it returns the same date with time is set to 12:00AM. Otherwise it returns the next day with time at 12:00AM