TIME Format

When a DATE value is displayed, Oracle must first convert that value from the special internal format to a printable string. The conversion is done by a function TO_CHAR, according to a DATE *format*. Oracle's default format for DATE is "DD-MON-YY". Therefore, when you issue the query

```
select b from x;
```

you will see something like:

```
B
---------
01-APR-98
```

Whenever a DATE value is displayed, Oracle will call TO_CHAR automatically with the default DATE format. However, you may override the default behavior by calling TO_CHAR explicitly with your own DATE format. For example,

```
SELECT TO_CHAR(b, 'YYYY/MM/DD') AS b
FROM x;
```

returns the result:

```
B
----------------------------------------------------------------------------
1998/04/01
```

The general usage of TO_CHAR is:

```
TO_CHAR(<date>, '<format>')
```

where the <format> string can be formed from over 40 options. Some of the more popular ones include:

, for example.

MM	Numeric month (*e.g.*, 07)
MON	Abbreviated month name (*e.g.*, JUL)
MONTH	Full month name (*e.g.*, JULY)
DD	Day of month (*e.g.*, 24)
DY	Abbreviated name of day (*e.g.*, FRI)
YYYY	4-digit year (*e.g.*, 1998)
YY	Last 2 digits of the year (*e.g.*, 98)
RR	Like YY, but the two digits are ``rounded" to a year in the range 1950 to 2049. Thus, 06 is considered 2006 instead of 1906
AM (or PM)	Meridian indicator
HH	Hour of day (1-12)
HH24	Hour of day (0-23)
MI	Minute (0-59)
SS	Second (0-59)

You have just learned how to output a DATE value using TO_CHAR. Now what about inputting a DATE value? This is done through a function called TO_DATE, which converts a string to a DATE value, again according to the DATE format. Normally, you do not have to call TO_DATE explicitly:

Whenever Oracle expects a DATE value, it will automatically convert your input string using TO_DATE according to the default DATE format "DD-MON-YY". For example, to insert a tuple with a DATE attribute, you can simply type:

```
insert into x values(99, '31-may-98');
```

Alternatively, you may use TO_DATE explicitly:

```
insert into x
values(99, to_date('1998/05/31:12:00:00AM', 'yyyy/mm/dd:hh:mi:ssam'));
```

The general usage of TO_DATE is:

```
TO_DATE(<string>, '<format>')
```

where the <format> string has the same options as in TO_CHAR.

Finally, you can change the default DATE format of Oracle from "DD-MON-YY" to something you like by issuing the following command in sqlplus:

```
alter session set NLS_DATE_FORMAT='<my_format>';
```

The change is only valid for the current sqlplus session.

++

You can try this small SQL script for yourself

```
create table testtime(

starttime date

);




insert into testtimevalues(to_date('1998/05/31:12:00:00AM','yyyy/mm/dd:hh:mi:ssam'));

insert into testtimevalues(to_date('2010/09/15:11:25:28AM','yyyy/mm/dd:hh:mi:ssam'));




select to_char(starttime, 'hh:mi:ss') as starttime from testtime;
```