Referential integrity contraints:

To define the column in the parent table as a primary key and the same column in the child table as a foreign key referring to the corresponding parent entry.

Foreign key:

A column or combination of columns included in the definition of referential integrity which would refer to a referenced key.

Referenced key :

It is a unique or a primary key which is defined on a column belonging to the parent table.

Child table :

This table depends upon the values present in the referenced key of the parent table which is referred by a foreign key.

Parent table:

This table determines whether insertion or updation of data can be done child table. This table would be referred by child's table foriegn key.

On-delete cascade clause:

If all the rows under the referenced key column in a parent table are deleted then all rows in the child table with dependent foreign key column will also be deleted automatically.

```
SQL> create table emp1(eno number, ename
varchar2(15),job varchar2(15),constraint eno_pk1
primary key(eno));

SQL> create table emp2(eno number,ename
varchar2(15),constraint eno_fk1 foreign key(eno)
references emp1(eno) on Delete Cascade);
```

SINGLE ROW FUNCTIONS :

A single row function or a scalar function returns only one value for every row queried in the table.

* Date function
* Numeric function
* Character function
* Conversion function
* Miscellaneous function

The examples that follows mostly used the system table "dual". Dual table has one column defined to be of varchar2 type and contains only one row with value 'x'.

Add_months :

This date function returns a date after adding a specified date with the specified number of months.

add_months(d,n)

d is the date

n represents the number of months.

```
SQL> select add_months(sysdate,1) from dual;
```

Last_day:

The format is last_day(d),which returns the date corresponding to the last day of the month.

```
SQL>select last_day(sysdate) from dual;
```

Months_between:

To find out the number of months between two dates.

months_between(d1,d2)

```
SQL>select months_between ('01-nov-2008','01-nov-2007')
from dual;
```

Next_day:

Next_day(d, day)

where d represents date and day implies any weekday.

```
SQL> select next_day(sysdate,'tuesday') from dual;
```

Round :

This function returns the date which is rounded to the unit specified by default date will be rounded to the nearest day.

SQL>Select round(to_date('18-nov-2008','dd-mon-yy'),'month') from dual; or day or year

Truncate :

This function returns starting day of the current month.

SQL>Select trunc(to_date('22-nov-2008','dd-mon-yyy'),'month') from dual; or day or year

Greatest :

The function is greatest (d1,d2,.....) where d1 and d2 are dates. This function returns the latest date present in the argument.

```
SQL>select greatest (to_date(sysdate),to_date('23-nov-2008','dd-mon-yy') from dual;
```

CHARACTER FUNCTIONS:

~~~~~~~~~~~~~~~~~~~~~~~~

concatenation : (||)

~~~~~~~~~~~~~~~~~

This function is used to concatenate two strings together.

SQL>select column1 || column2 from <table_name>;

SQL>select column1 ||' , '|| column2 from <table_name>;

ASCII:

This function is used to return the decimal representation in the database character set of the first character of the string.

SQL>select ascii('a') from dual;

CHR :

~~~~

Returns the character having the binary equivalent to the string in either the database character set or the national character set.

SQL>Select chr(97) from dual;

CONCAT:

~~~~~~~

Concatenates two strings together (same as ||)

SQL>select concat(column1,column2) from <table name>;

INITCAP:
~~~~~~~

Initial capital, capitalizes the first letter of a word or series of words.

SQL>Select initcap('hello') from dual;

LENGTH:
~~~~~~~

Tells the length of a string.

LOWER:
~~~~~~~

Converts every letter in a string to lowercase.

UPPER :
~~~~~~~

Converts every letters in a string to uppercase.

LPAD:

~~~~~

Left pad. Makes a string a certain length by adding a certain set of characters to the left.

SQL>select lpad(first_name,10),last_name from employee;

SQL>select lpad(first_name,10,'~') from employee;

LTRIM:

~~~~~

Left trim. Trims all the occurances of any one of a set of characters off the left side of a string.

SQL>Select ltrim('xyzadams',xyz') from dual;

RPAD:

Right pad. Makes a string a certain length by adding a certain set of characters to the right.

```
SQL>select rpad(first_name,20,'-') from employee;
```

RTIRM:

Right trim. Trims all the occurances of any one of a set of characters off the right side of a string.

```
SQL>select rtirm('xyzadams','adams') from dual;
```

TRANSLATE: (char,from,to)
~~~~~~~~~~

```
SQL>select translate('jack','j','b') from dual;
```

REPLACE: (char,search_string,replace_string)
~~~~~~~~

```
SQL>select replace('jack and jue','j','bl') from dual;
```

SUBSTR (char,m,n);
~~~~~~~

```
SQL>select substr('abcdefg',3,2) from dual;
```
~~~~~~~