**INTEGRITY CONSTRAINTS**

Integrity Constraints are used to prevent entry of invalid information into tables. There are five Integrity Constraints Available in Oracle. They are :

* Not Null
* Primary Key
* Foreign Key
* Check
* Unique

**Not Null**

By default all columns in a table can contain null values. If you want to ensure that a column must always have a value, i.e. it should not be left blank, then define a NOT NULL constraint on it.

Always be careful in defining NOT NULL constraint on columns, for example in employee table some employees might have commission  and some employees might not have any commission. If you put NOT NULL constraint on COMM column then you will not be able insert rows for those employees whose commission is null. Only put NOT NULL constraint on those column which are essential for example in EMP table ENAME column is a good candidate for NOT NULL constraint.

**Primary Key**

Each table can have one primary key, which uniquely identifies each row in a table and ensures that no duplicate rows exist. Use the following guidelines when selecting a primary key:

* Whenever practical, use a column containing a sequence number. It is a simple way to satisfy all the other guidelines.
* Minimize your use of composite primary keys. Although composite primary keys are allowed, they do not satisfy all of the other recommendations. For example, composite primary key values are long and cannot be assigned by sequence numbers.
* Choose a column whose data values are unique, because the purpose of a primary key is to uniquely identify each row of the table.
* Choose a column whose data values are never changed. A primary key value is only used to identify a row in the table, and its data should never be used for any other purpose. Therefore, primary key values should rarely or never be changed.
* Choose a column that does not contain any nulls. A PRIMARY KEY constraint, by definition, does not allow any row to contain a null in any column that is part of the primary key.
* Choose a column that is short and numeric. Short primary keys are easy to type. You can use sequence numbers to easily generate numeric primary keys.

For example in EMP table EMPNO column is a good candidate for PRIMARY KEY.

To define a primary key on a table give the following command.

alter table emp add constraint emppk primary key (empno);

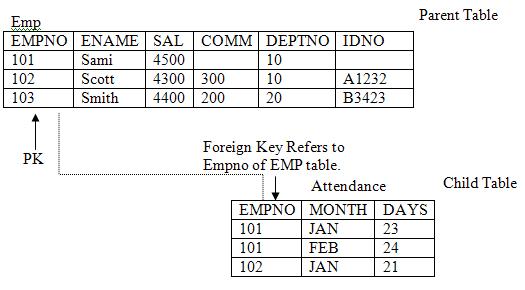
The above command will succeed only if the existing values are compliant i.e. no duplicates are there in EMPNO column. If EMPNO column contains any duplicate value then the above command fails and Oracle returns an error indicating of non compliant values.

Whenever you define a PRIMARY KEY Oracle automatically creates a index on that column. If an Index already exist on that column then Oracle uses that index.

**FOREIGN KEY**

On whichever column you put FOREIGN KEY constraint then the values in that column must refer to existing values in the other table. A foreign key column can refer to primary key or unique key column of other tables. This Primary key and Foreign key relationship is also known as PARENT-CHILD relationship i.e. the table which has Primary Key is known as PARENT table and the table which has Foreign key is known as CHILD table. This relationship is also known as REFERENTIAL INTEGRITY.

The following shows an example of parent child relationship.



Here EMPNO in attendance table is a foreign key referring to EMPNO of EMP table.

alter table attendance add constraint empno\_fk   
     foreign key (empno) references emp(empno);

The above command succeeds only if EMPNO column in ATTENDANCE table contains values which are existing in EMPNO column of EMP table. If any value is not existing then the above statement fails and Oracle returns an error indicating non compliant values.

Some points to remember for referential integrity

* You cannot delete a parent record if any existing child record is there. If you have to first delete the child record before deleting the parent record. In the above example you cannot delete row of employee no. 101 since it’s child exist in the ATTENDANCE table. However, you can delete the row of employee no. 103 since no child record exist for this employee in ATTENDANCE table.  If you define the FOREIGN KEY with ON DELETE CASCADE option then you can delete the parent record and if any child record exist it will be automatically deleted.

To define a foreign key constraint with ON DELETE CASCADE option give the following command.

ALTER TABLE attendance ADD CONSTRAINT empno\_fk   
          FOREIGN KEY (empno) REFERENCES emp(empno)  
              ON DELETE CASCADE;

From Oracle version 9i, Oracle has also given a new feature i.e. ON DELETE SET NULL . That is it sets the value for foreign key to null whenever the parent record is deleted.

To define a foreign key constraint with ON DELETE SET NULL option give the following command.

ALTER TABLE attendance ADD CONSTRAINT empno\_fk   
          FOREIGN KEY (empno) REFERENCES emp(empno)  
              ON DELETE SET NULL;

* You also cannot drop the parent table without first dropping the FOREIGN KEY constraint from attendance table. However if you give CASCADE CONSTRAINTS option in DROP TABLE statement then Oracle will automatically drop the references and then drops the table.

**CHECK**

Use the check constraint to validate values entered into a column. For example in the above ATTENDANCE table, the DAYS column should not contain any value more than 31. For this you can define a CHECK constraint as given below

alter table attendance add constraint dayscheck  
             check (days <= 31);

Similarly if you  want the salaries entered in to SAL column of employee table should be between 1000 and 20000 then you can define a CHECK constraint on EMP table as follows

alter table emp add constraint sal\_check   
                   check (sal between 1000 and 20000);

You can define as  many check constraints on a single column as you want there is no restrictions on number of check constraints.

**UNIQUE KEY**

Unique Key constraint is same as primary key i.e. it does not accept duplicate values, except the following differences

* There can be only on Primary key per table. Whereas, you can have as many Unique Keys per table as you want.
* Primary key does not accept NULL values whereas, unique key columns can be left blank.
* You can also refer to Unique key from Foreign key of other tables.

On which columns you should put Unique Key Constraint ?

It depends on situations, first situation is suppose you have already defined a Primary key constraint on one column and now you have another column which also should not contain any duplicate values, Since a table can have only one primary key,  you can define Unique Key constraint on these columns.  Second situation is when a column should not contain any duplicate value but it should also be left blank. For example in the EMP table IDNO is a good candidate for Unique Key because all the IDNO’s are unique but some employees might not have ID Card so you want to leave this column blank.

To define a UNIQUE KEY constraint on an existing table give the following command.

alter table emp add constraint id\_unique unique (idno);

Again the above command will execute successfully if IDNO column contains complying values otherwise you have to remove non complying values and then add the constraint.