Experiment No.2

Environment: Microsoft Windows
Tools/ Language: Oracle

OBJECTIVE: To implement the restrictions and modification on the structure of the table.

Theory:

Data constraints: Besides the column name, column length and column data type, there are other parameters i.e. other data constraints that can be passed by the DBA at check creation time. The constraints can either be placed at column level or at the table level.

i. **Column Level Constraints:** If the constraints are defined along with the column definition, it is called a column level constraint.
ii. **Table Level Constraints:** If the data constraint attached to a specified column in a table reference the contents of another column in the table then the user will have to use table level constraints.

List of most used Constraint

- **NOT NULL**
- **DEFAULT**
- **UNIQUE**
- **CHECK**
- **PRIMARY KEY**
- **FOREIGN KEY**
  - **On delete cascade**
  - **On delete set null**

Null Value Concepts:-while creating tables if a row lacks a data value for particular column that value is said to be null. Column of any data types may contain null values unless the column was defined as not null when the table was created.

Syntax:

```
Create table tablename
(columnname data type (size) not null ……)
```

Note: Not null constraint cannot be defined at table level.

Primary Key:primary key is one or more columns is a table used to uniquely identify each row in the table. Primary key values must not be null and must be unique across the column. A multicolumn primary key is called composite primary key.

Syntax: primary key as a column constraint

```
Create table tablename
(columnname datatype (size) primary key,….)
```

Composite Primary key as a table constraint

```
Create table tablename
(columnnamedatatype (size), columnnamedatatype( size)…
Primary key (columnname,columnname));
```

Unique key concept:-A unique key is similar to a primary key except that the purpose of a unique key is to ensure that information in the column for each record is unique as with telephone or devices license numbers. A table may have many unique keys.

Syntax: Unique as a column constraint.

```
Create table table name
(columnname datatype (size) unique);
```

Unique as table constraint:

```
Create table tablename
(columnname datatype(size),columnname datatype(size)…
unique (columnname));
```

Default value concept: At the time of column creation, a default value can be assigned to it. When the user is loading a record with values and leaves this column empty, the DBA will automatically load this column with the default value specified. The data type of the default value should match the data type of the column.

Syntax:

```
Create table tablename
(columnname datatype (size) default value,….);
```

Note: The default value constraint cannot be specified at table level.

Foreign Key Concept: Foreign key represents relationship between tables. A foreign key is column whose values are derived from the primary key of the same attribute of some other table. A foreign key must have corresponding primary key value in the primary key table to have meaning.

Foreign key as a column constraint

Syntax :

```
Create table table name
(columnname datatype(size) references another-tablename);
```

Foreign key as a table constraint:

Syntax :

```
Create table name
(columnname datatype(size)….
foreign key(columnname) references table name);
```

Check Integrity Constraints: Use the check constraints when you need to enforce integrity rules that can be evaluated based on a logical expression. Following are a few examples of appropriate check constraints.

- A check constraints on the column ‘name’ of the Employee table so that the name is entered in upper case.
- A check constraint on the column ‘Emp_no’ of the Employee table so that no Emp_no value starts with ‘e’.

Syntax:

```
Create table tablename
(columnname datatype(size),……..
CONSTRAINT constraintname check(expression));
```

Modifying the Structure of Tables- Alter table command is used to changing the structure of a table. Using the alter table clause you cannot perform the following tasks:

(i) change the name of table
(ii) decrease the size of a column if table data exists and occupies larger size.

The following tasks you can perform through alter table command.

(i) Adding new columns:

Syntax:

```
ALTER TABLE tablename
ADD (newcolumnname newdatatype (size));
```

(ii) Modifying existing table

Syntax:

```
ALTER TABLE tablename
MODIFY (newcolumnname newdatatype (size));
```

(iii) Deleting a column

Syntax:

```
ALTER TABLE tablename
DROP COLUMN columnname;
```

Removing/Deleting Tables- Following command is used for removing or deleting a table.

Syntax:

```
DROP TABLE tablename;
```

Defining Integrity constraints in the ALTER TABLE command-

You can also define integrity constraints using the constraint clause in the ALTER TABLE command.

The following examples show the definitions of several integrity constraints.

(1) **Add PRIMARY KEY**-

Syntax:

```
ALTER TABLE tablename
ADD PRIMARY KEY(columnname);
```

(2) **Add FOREIGN KEY**-

Syntax:

```
ALTER TABLE tablename
ADD CONSTRAINT constraintname
FOREIGN KEY(columnname) REFERENCES tablename;
```

(3) **Add CHECK CONSTRAINT**-

Syntax:

```
ALTER TABLE tablename
ADD CONSTRAINT constraintname
Check(expression);
```

Dropping integrity constraints in the ALTER TABLE command-

You can drop an integrity constraint if the rule that it enforces is no longer true or if the constraint is no longer needed. Drop the constraint using the ALTER TABLE command with the DROP clause.

The following examples illustrate the dropping of integrity constraints.

(1) **DROP the PRIMARY KEY**-

Syntax:

```
ALTER TABLE tablename
DROP PRIMARY KEY;
```

(2) **DROP FOREIGN KEY**-

Syntax:

```
ALTER TABLE tablename
DROP FOREIGN KEY;
```

(3) **DROP CONSTRAINT**-

Syntax:

```
ALTER TABLE tablename
DROP CONSTRAINT constraintname;
```

Practical Assignment - 2

Department: Computer Engineering & Applications

Course:B.Tech. (CSE)

Subject: Database Management System Lab (CSE3083)

Year: 2nd **Semester:**3rd

1. Create the following tables and specify constraints at the time of creation.

Department

Column Name	Data Type	Size	Constraint
Deptno	number	3	primary key
Dname	varchar2	20	Unique
Location	varchar2	20	not null, department are located in Delhi, Pune, Agra

Employee

Column Name	Data Type	Size	Constraint
Empno	varchar2	5	primary key, should start with 'E'
Ename	varchar2	20	Unique
Designation	varchar2	20	not null
Salary	number	10	default 25000, must lie between 15000 and 50000
DOB	date		not null
Dno	number	3	foreign key (references department)

Candidate

Column Name	Data type	Size	Constraints
Candidate_ID	Number	6	Primary key of the table
Candidate_Name	Varchar2	20	Not Null
Candidate_Email	Varchar2	30	Unique, Must have '@' followed by '.' in between the email
Candidate_Dept	Varchar2	2	Default 'HR'
Manager_ID	Number	6	It can take only those values which are present in Candidate_ID column

2.Create the schemas as specified above without specifying any constraints.

College (**cName**: varchar2(10), **state**: varchar2**(**10**)**, **enrollment**: int)
Student (**sID**: int, **sName**: varchar2(10), **GPA**: number(2,1)**, sizeHS**: int)
Apply (**sID**: int, **cName**: varchar2(10), **major**: varchar2(20))

1. Add **cName** as **Primary key** in **College**.
2. Add **sID** as **Primary key** in **Student**.
3. Add **sID**, **cName**, **major** as **Primary key** in **Apply**.
4. Make **sID** in **Apply foreign key** referring table **student** and **cName** referring table **college**.
5. Increase data type size of **major** from 20 to **25**.
6. Add a new column **decision** in the **Apply** table keeping a constraint of **not null** for this column with data type **varchar2(3)**.
7. Change data type of **decision** in **Apply** to **char(1).**
8. Drop foreign key on column name **cName** from **Apply** table.
9. Remove column **sizeHS** from **Student** table.
10. Drop **primary key** from College
11. Make **cName, major** unique pairwise such as Stanford CS, Stanford EE.
12. Add **cName** as **Foreign Key** in **Apply** table referring table **College** using on delete cascade.
13. Modify foreign key on **sID** in **Apply** table to **foreign key on delete set null.**
14. Rename column **enrollment** to **enroll** in College Table.

Exercise

Customer Table

Column name	Datatype	Description	Constraints
CustomerId	Varchar2(6)	Unique id generated for each customer	Primary Key, Should start with 'C'
CustomerName	Varchar2(30)	Name of the customer	Not null
DateOfReg	Date	Date on which the customer registered	
UserId	Varchar2(15)	Decided at the time of registration	It should be unique
Password	Varchar2(15)	Decided at the time of registration	Not Null

BankInfo Table

Column name	Datatype	Description	Constraints	
AccountNo	Number(10)	Account no of customer		Composite Primary key
CustomerId	Varchar2(6)	Unique id provided to each customer when he/she is registered to purchase items	Foreign key referring to customer table (**ON DELETE CASCADE**)	

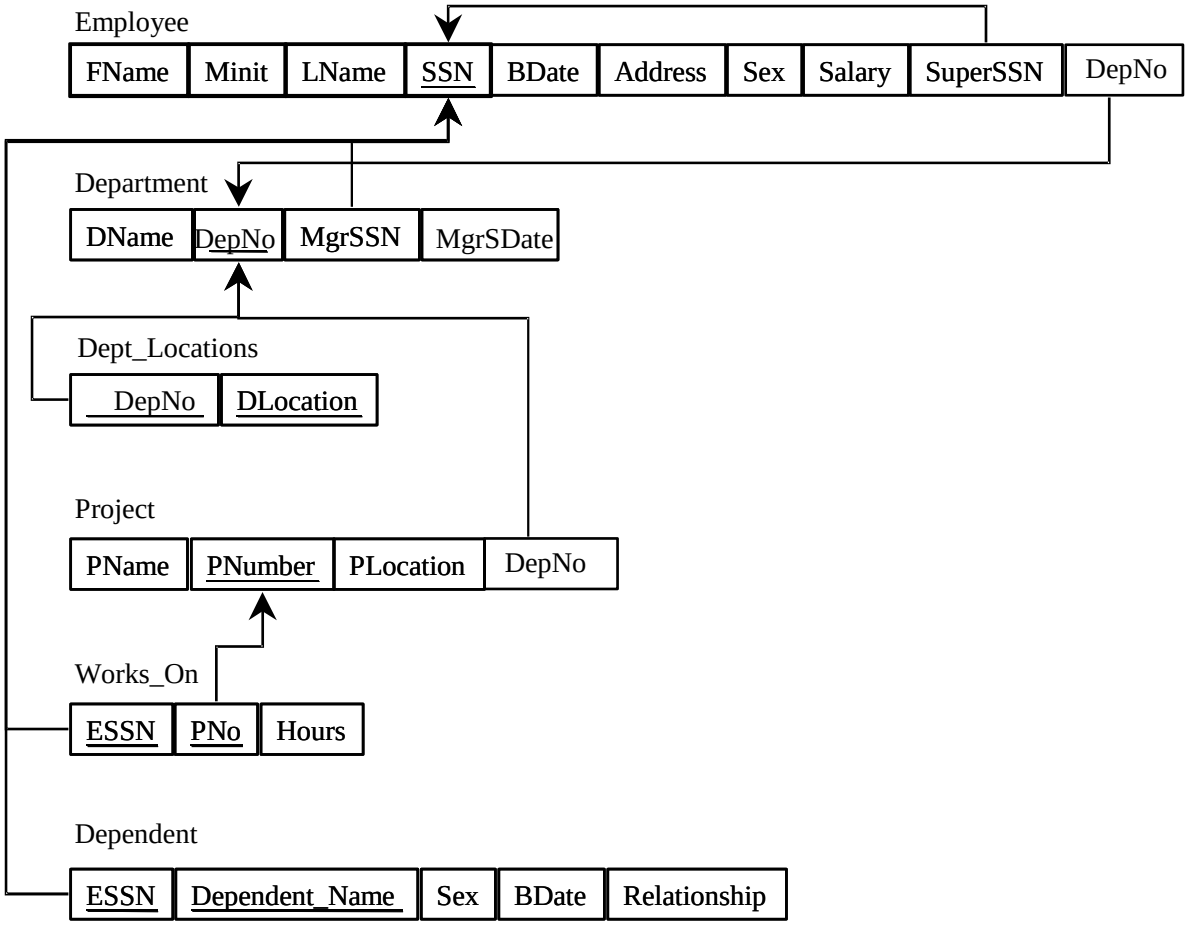
Billing Table

Column name	Datatype	Description	Constraints
BillId	Number(4)	Unique Id generated for each bill	Primary key
AccountNo	Number(10)	Account no which is used to pay the bill	Composite Foreign key to Bank info table
CustomerId	Varchar2(6)	Customer id of the customer who does the purchase of items	
BillDate	Date	The date of payment of bill	Default SYSDATE
PaymentType	Varchar2(12)	Type of Payment	Either Creditcard or Debitcard

Item Table

Colum name	Datatype	Description	Constraints
ItemId	Varchar2(6)	Unique Id provided for each item. (eg STN001 for stationery items)	Primary Key
ItemName	Varchar2(30)	Name of the item	Not Null
QtyOnHand	Number(3)	Current availability of item in the shop	Should be greater than ReOrderLevel (Table level constraint)
UnitPrice	Number(6,2)	Sell price of item per unit	Greater than 0
Class	Char(1)	Depending on the UnitPrice, items belongs to various Classes. eg: A,B,C etc.	Class of Item is 'A' if UnitPrice is less than 100, 'B' if UnitPrice is less than 1000, 'C' if UnitPrice is 1000 and above (Table level constraint)
UnitOfMeasurement	Varchar2(12)	Unit used to measure the quantity (eg Kilogram, dozen etc)	
ReOrderLevel	Number(3)	Minimum Quantity after which the supplier must be ordered for new stock	Greater than 0
ReorderQty	Number(3)	Minimum Quantity that can be ordered to the supplier	Greater than 0
Discount	Number(2)	Percentage discount on the item to the customer	

Convert the following relational schema into tables with proper constraints that are required.



Pre Experiment Questions

1. What is difference between table level constraint and column level constraint?
2. Difference between unique and primary key.
3. What is Foreign Key?
4. Difference between foreign and primary key

Post Experiment Questions

1. How to restrict domain of an attribute?
2. How to use NOT NULL constraint? Discuss the utility from application viewpoint.
3. How to alter Primary Key?
4. How to update a table by enforcing constraint?
5. Can we replace Primary Key with Unique Not Null?
6. Can we decrease the size of data type always?
7. What are the constraints that cannot be defined at table level?