**Exercise 2**

**Data Definition Language (DDL)**

**More datatypes:**

| **Name** | **Storage Size** | **Description** | **Low Value** | **High Value** | **Resolution** |
| --- | --- | --- | --- | --- | --- |
| timestamp | 8 bytes | both date and time (no time zone) | 4713 BC | 294276 AD | 1 microsecond / 14 digits |
| date | 4 bytes | date (no time of day) | 4713 BC | 5874897 AD | 1 day |
| time | 8 bytes | time of day (no date) | 00:00:00 | 24:00:00 | 1 microsecond / 14 digits |

**e.g. documents** (

1. CREATE TABLE documents (

document\_id int PRIMARY KEY,

header\_text VARCHAR (255) NOT NULL,

posting\_date DATE NOT NULL DEFAULT CURRENT\_DATE);

1. CREATE TABLE shifts (

      id int PRIMARY KEY,

      shift\_name VARCHAR NOT NULL,

     start\_at TIME NOT NULL,  end\_at TIME NOT NULL);

1. INSERT INTO shifts(id, shift\_name, start\_at, end\_at)

VALUES (1, 'Morning', '08:00:00', '12:00:00'),

       (2, 'Afternoon', '13:00:00', '17:00:00'), (3, 'Night', '18:00:00', '22:00:00');

1. Create table timestamp\_demo(id int, ts timestamp);
2. INSERT INTO timestamp\_demo (id, ts)

VALUES (1,'2016-06-22 19:10:25' );

**An option for Foreign Key:**

A foreign key with an on delete cascade means that if a record in the parent table is deleted, then the corresponding records in the child table will automatically be deleted.

A foreign key with an on update cascade means that if a record in the parent table is updated, then the corresponding records in the child table will automatically be updated.

Example:

Create table product (prod\_id int primary key, prod\_name varchar(15), price numeric(6,2));

Create table order (order\_id int, product\_id int, quantity int, primary key(order\_id,product\_id), foreign key (product\_id) references product on delete cascade on update cascade);

Consider the schema you did for last lab session:

Department (dept\_name, building, budget)

Instructor (ID, name,dept\_name, salary)

Course (Course\_id, title, dept\_name, credits)

Now create these tables and add suitable constraints including cascade options:

Student (ID, name, state, dept\_name)

Pre-req\_Course(Course\_id, prereq\_courseID)

Advisor (Student\_id, instructor\_id)

Register (Student\_id, Course\_id, grade)

-----------------------------------

**Example Alter statements in PostgreSQL**

To add a column of type varchar to a table:

1. ALTER TABLE student ADD COLUMN tuition\_fee numeric(9,2);
2. ALTER TABLE student ADD COLUMN bus\_fee numeric(9,2);

To drop a column from a table:

1. ALTER TABLE student DROP COLUMN bus\_fee;

To change the types of two existing columns in one operation:

1. ALTER TABLE student

ALTER COLUMN state TYPE varchar(80),

ALTER COLUMN name TYPE varchar(30);

**To add a check constraint**

1. alter table student add constraint fee\_check check(fee>0);

**To remove the constraint**

1. alter table student drop constraint qty\_chk;

**To rename a column**

1. alter table student rename column state to stt;

**To rename a table**

1. alter table student rename to student\_123;
2. Add a column each of your choice with date datatype and time datatype
3. Add a column CGPA to student table and insert suitable data.
4. Find the difference between

Drop table student;

Delete from student;

Truncate student;

1. Find the students whose name starts with 'A' or 'C' (name like ‘A%’ or name like ‘C%’)
2. Find the number of students in ‘CS’ department. (count)
3. Display the student details sorted by name alone. (order by name in descending order)
4. Display the student details sorted by name and state together. (order by name,state)
5. Find the students coming from Kerala and with a cgpa > 8.
6. Find the students coming from Kerala or with a cgpa > 8.
7. Find the students with a CGPA between 5 and 8.
8. Find the students coming from Kerala, Andhra and Tamil Nadu. (Use IN)
9. Find the department names of students without duplicate values.
10. Find the maximum and minimum CGPA from student table.
11. Find the total number of students in CS department.
12. Display the name of students under a column with heading ‘Student\_name’ instead of name which is the name of your column.
13. Find the sum and average of tuition fee.
14. Find the total fee as tuition fee+bus fee for student
15. Find the number of students who have not paid bus\_fee.