Name: \_\_\_\_\_\_\_\_\_\_\_\_\_Irene Kyei\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Instructions:

|  |
| --- |
|  |
| DROP TABLE IF EXISTS student; |
|  | CREATE TABLE student |
|  | ( |
|  | id serial PRIMARY KEY, |
|  | first\_name character varying, |
|  | last\_name character varying, |
|  | email character varying, |
|  | gender character varying, |
|  | work\_phone character varying, |
|  | book\_preference\_hardcopy boolean |
|  | ); |
|  |  |
|  | copy student(first\_name,last\_name,email,gender,work\_phone,book\_preference\_hardcopy) |
|  | --set the path for file location of student\_data.csv |
|  | from 'C:\NITEEN\Training\data\student\_data.csv' |
|  | delimiter ',' CSV header |
|  |  |
|  |  |
|  |  |
|  | DROP TABLE IF EXISTS student\_marks; |
|  | CREATE TABLE student\_marks |
|  | ( |
|  | id serial PRIMARY KEY, |
|  | student\_reg\_id integer, |
|  | student\_id integer, |
|  | unit2 integer, |
|  | unit3 integer, |
|  | unit4 integer, |
|  | unit5 integer |
|  | ); |
|  |  |
|  | copy student\_marks(student\_reg\_id,student\_id,unit2,unit3,unit4,unit5) |
|  | --set the path for file location of student\_marks.csv |
|  | from 'C:\NITEEN\Training\data\student\_marks.csv' |
|  | delimiter ',' CSV header |
|  |  |
|  |  |
|  | Sample questions: |
|  |  |
|  | -- students with the highest marks in Unit 4 |
|  |  |
|  | -- students scored between 89 and 100 unit4 |
|  |  |
|  | Open ended questions: |
|  | -- Take a closer look at the tables that you created and come up with 10 different scenarios/ questions and form SQL |
|  | -- Ask your colleagues |
|  |  |

Answers

create table student(

id serial PRIMARY KEY,

first\_name character varying,

last\_name character varying,

email character varying,

gender character varying,

work\_phone character varying,

book\_preference\_hardcopy boolean

);

select\*from student;

copy student(first\_name,last\_name,email,gender,work\_phone,book\_preference\_hardcopy)

from 'C:\Program Files\PostgreSQL\14\bin\Student\_data.csv'

delimiter ',' CSV header;

select\*from student;

create table student\_marks(

id serial PRIMARY KEY,

student\_reg\_id integer,

student\_id integer,

unit2 integer,

unit3 integer,

unit4 integer,

unit5 integer

);

select\*from student\_marks;

copy student\_marks(student\_reg\_id,student\_id,unit2,unit3,unit4,unit5)

from 'C:\Program Files\PostgreSQL\14\bin\student\_marks.csv'

delimiter ',' CSV header;

select\*from student\_marks;

--Questions

--1. Suppose the passing score is 90. List students by student names with a passing score of 90 and above in unit 2.

select distinct first\_name, last\_name, unit2 from student, student\_marks

where unit2>=90;

--2. Students scored between 89 and 100 unit4

select first\_name, last\_name, unit4 from student, student\_marks

where unit4 between 89 and 100;

--3. Students with the highest marks in Unit 4

select max(unit3), first\_name, last\_name from student\_marks, student

Group by first\_name, last\_name;

--4. From student, order first name by asceding order and last name by descending order

select first\_name, last\_name, email, gender, work\_phone, book\_preference\_hardcopy, student\_reg\_id, unit2, unit3, unit4, unit5

from student, student\_marks

order by first\_name, last\_name, email, gender, work\_phone, book\_preference\_hardcopy, student\_reg\_id, unit2, unit3, unit4, unit5 desc;

select \* from student

order by first\_name asc, last\_name desc;

--5. Inner join common columns between the 2 tables.

select student\_marks.student\_id, student.id from student\_marks

inner join student

on student\_marks.student\_id = student.id;