1. Create tables for the above list given

**create** **table** users(user\_id **bigint** **AUTO\_INCREMENT** **primary** **key**,user\_name **varchar**(200) **not** **null**,email **varchar**(200) **not** **null** ,**password** **varchar**(200) **not** **null**,batch **varchar**(100) **not** **null**,course\_enrolled **bigint**,mentor\_assigned **int**);

**create** **table** codekata(user\_id **bigint**,topic\_id **int**,prob\_solved **INT** **not** **null**,geekcoins\_earned **int** **null**);

**create** **table** attendance(class\_dates **date**,user\_id **bigint**,is\_absent **boolean** **default** **false**);

**create** **table** topics(topic\_id **int** **AUTO\_INCREMENT** **primary** **key**,topic\_name **varchar**(100) **unique**,total\_probs **bigint**);

**create** **table** tasks(task\_id **int** **AUTO\_INCREMENT** **primary** **key**,user\_id **bigint**,Assigned\_date **Datetime**,Completed\_date **timestamp**,is\_completed **boolean** **default** **false**);

**create** **table** mentors(mentor\_id **int** **AUTO\_INCREMENT** **primary** **key**,mentor\_name **varchar**(100) **not** **null**,assigned\_batch **varchar**(100) **not** **null**);

**create** **table** company\_drives(company\_id **bigint** **AUTO\_INCREMENT** **primary** **key**,company\_name **varchar**(200) **not** **null**,user\_id **bigint**,is\_attended **boolean** **default** **false**);

**create** **table** courses(course\_id **bigint** **AUTO\_INCREMENT** **primary** **key**,course\_name **varchar**(200) **not** **null**,course\_fees **int** **not** **null**,course\_desc **varchar**(500) **null**);

**create** **table** students\_activated\_courses(user\_id **bigint**,course\_id **bigint**,fees\_paid **boolean** **default** **false**);

2. insert at least 5 rows of values in each table

**insert** **into** users(user\_name,email,**password**,batch,course\_enrolled,mentor\_assigned) **values**('Nisha','nisha@gmail.com','nisha+','B2WE',1,1),('Kumar','Kumar@yahoo.com','kumar\*','B1WE',2,1),

('Karthi','karthi@gmail.com','karthik3','B1WE',3,2),('sandhiya','sandhiya@yahoo.com','sandy345','B1WE',1,3),('divya','divya@gmail.com','divya24','B3WE',3,2);

**insert** **into** codekata (user\_id ,topic\_id ,prob\_solved,geekcoins\_earned) **values** (1,2,20,1),(5,1,40,50),(1,3,20,5);

**insert** **into** codekata (user\_id ,topic\_id ,prob\_solved) **values** (3,4,10),(2,5,10);

**insert** **into** attendance(class\_dates,user\_id,is\_absent) **values** ('2022-02-16',2,1),('2022-02-16',3,1);

**insert** **into** attendance(class\_dates,user\_id) **values** ('2022-02-16',4),('2022-02-16',5),('2022-02-16',1);

**insert** **into** topics (topic\_name,total\_probs) **values** ('JS',100),('React',50),('HTML',150),('CSS',100),('MYSQL',200);

**insert** **into** tasks(user\_id,Assigned\_date,is\_completed) **values** (1,'2022-02-10',1),(1,'2022-02-11',1),(3,'2022-02-12',1),(2,'2022-02-12',1),(3,'2022-02-13',1);

**insert** **into** mentors (mentor\_name,assigned\_batch) **values** ('Akash','B1WE'),('Parvathy','B2WE'),('Pradeep','B1WE'),('Eeswari','B2WE'),('Priya','B3WE');

**insert** **into** company\_drives(company\_name,user\_id,is\_attended) **values** ('TCS',1,1),('CTS',1,1),('Verizon',2,1),('Infosys',3,1);

**insert** **into** company\_drives (company\_name,user\_id) **values** ('Wipro',2);

**insert** **into** courses (course\_name ,course\_fees ,course\_desc) **values**('Python Developers',1000,'Useful to learn python'),('Full-stack',70000,'Useful to learn to build a site'),('DB COURSE',500,'Database');

**insert** **into** courses (course\_name ,course\_fees) **values** ('.Net development',5000),('Cybersecurirty',10000);

**insert** **into** students\_activated\_courses **values**(1,1,1),(2,2,1),(3,2,1),(4,3,1),(5,2,1);

3. get number problems solved in codekata by combining the users

**select** user\_id ,**sum**(prob\_solved) total\_prob\_solved **from** codekata **group** **by** user\_id ;

4. display the no of company drives attended by a user

**select** user\_id ,**count**(company\_name) No\_of\_drives\_attended **from** company\_drives **group** **by** user\_id ;

5. combine and display students\_activated\_courses and courses for a specific user groping them based on the course

**select** user\_id ,course\_name,sac.course\_id **from** students\_activated\_courses sac **inner** **join** courses c **on** sac.course\_id =c.course\_id **group** **by** course\_name ;

6. list all the mentors

**select** mentor\_id,mentor\_name **from** mentors;

7. list the number of students that are assigned for a mentor

**select** **count**(user\_id) No\_of\_students,mentor\_name **from** users **right** **join** mentors **on** mentor\_assigned =mentor\_id **group** **by** mentor\_id;