**COMPUTER SCIENCE**



**MYSQL file**

**NAME- YUV JINDAL**

**CLASS- XII-C**

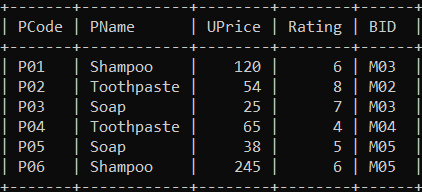
**ROLL NUMBER- 19**

**INDEX**

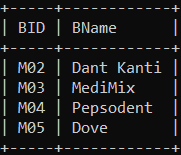
1. Consider the tables PRODUCTS and BRAND and write sql queries and output of the sql queries respectively.
2. Write the SQL queries and output of the SQL queries based on the relations TEACHER and PLACEMENT.
3. Write SQL queries and output of the SQL queries based on the tables EMPLOYEE and DEPARTMENT.
4. Write SQL queries and find the output of the SQL queries based on the tables TRAINER and COURSE.
5. Write SQL commands for the queries and output of the queries based on the tables WATCHES and SALE.

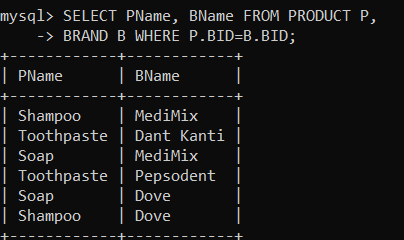
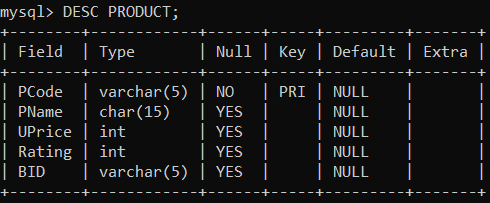
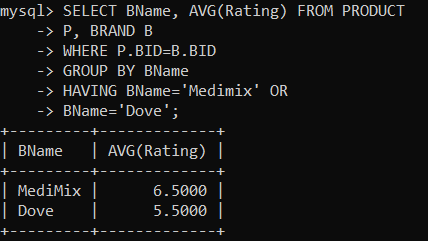
Q1. Consider the tables PRODUCTS and BRAND and write sql queries and output of the sql queries respectively.

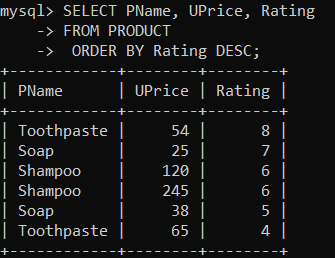
**PRODUCTS**

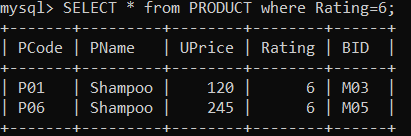


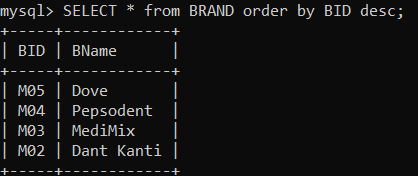
**BRAND**



1. Display product name and brand name from the tables PRODUCT and BRAND.
2. Display the structure of the table PRODUCT.
3. Display the average rating of Medimix and Dove brands.
4. Display the name, price, and rating of products in descending order of rating.

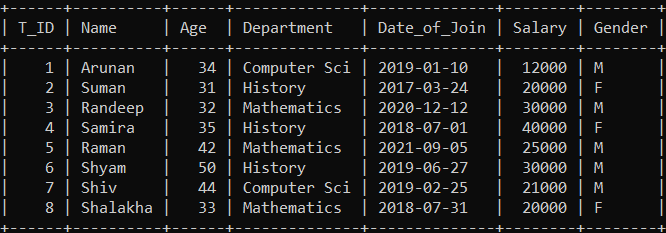


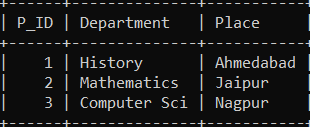
1. SELECT \* from PRODUCT where Rating=6;
2. SELECT \* from BRAND order by BID desc;



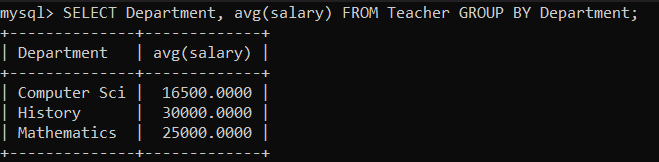
Q2. Write the SQL queries and output of the SQL queries based on the relations TEACHER and PLACEMENT.

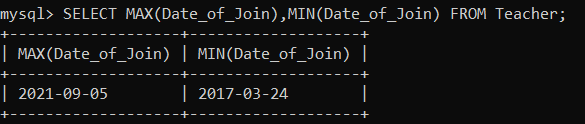
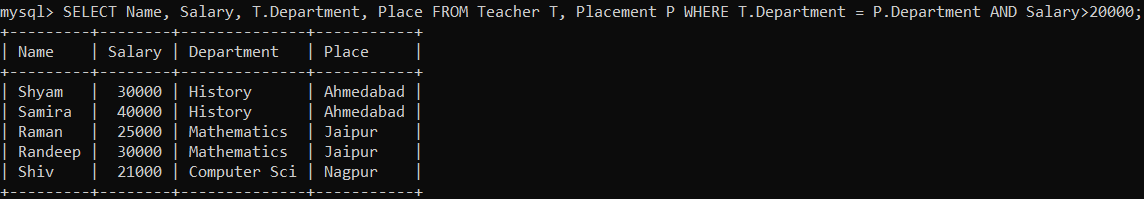
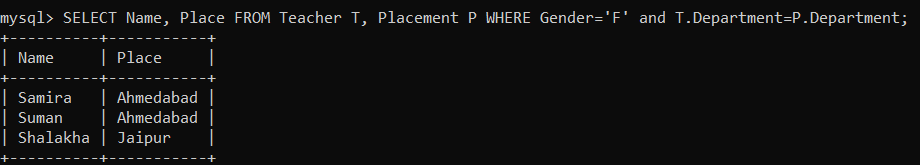
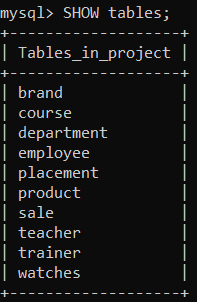
**TEACHER**

****

**PLACEMENT**

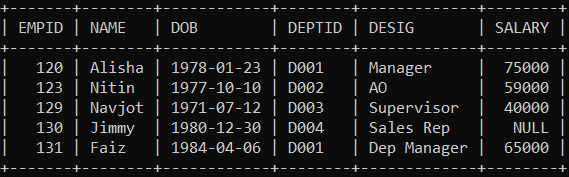
1. SELECT Department, avg(salary) FROM Teacher GROUP BY Department;



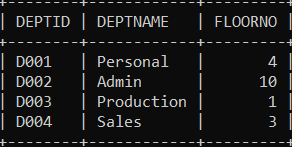
1. SELECT MAX(Date\_of\_Join),MIN(Date\_of\_Join) FROM Teacher;
2.  SELECT Name, Salary, T.Department, Place FROM Teacher T, Placement P WHERE T.Department = P.Department AND Salary>20000;
3. SELECT Name, Place FROM Teacher T, Placement P WHERE Gender='F' and T.Department=P.Department;
4. Write the command to view all tables in a database.

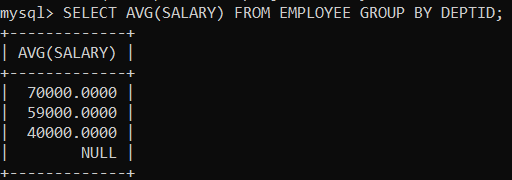
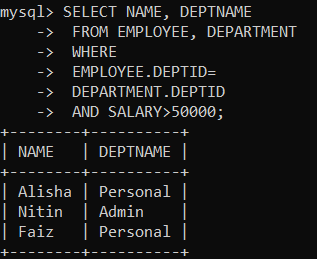
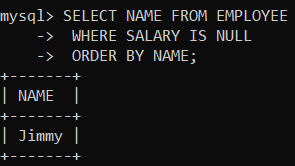
Q3. Write SQL queries and output of the SQL queries based on the tables EMPLOYEE and DEPARTMENT.

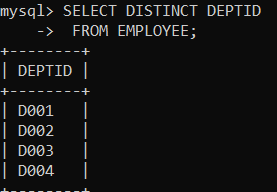
**EMPLOYEE**

****

**DEPARTMENT**

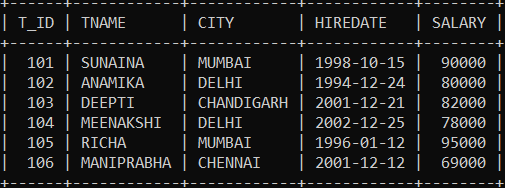


1. To Display the average salary of all employees, department wise.
2. To Display name and respective department name of each employee whose salary is more than 50000.
3. To Display the names of all employees whose salary is not known, in alphabetical order.
4. To Display DEPTID from the table EMPLOYEE without any repetition.

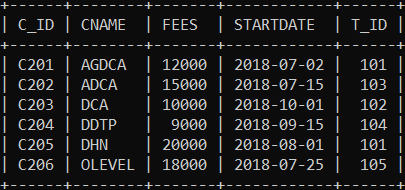


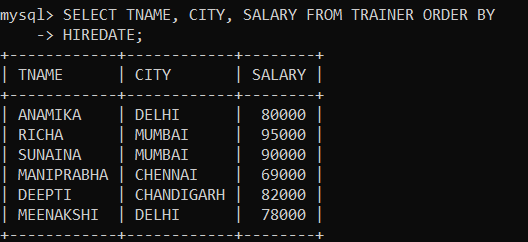
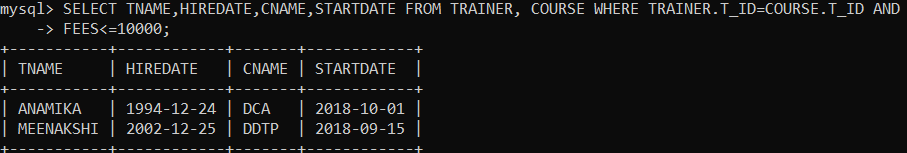
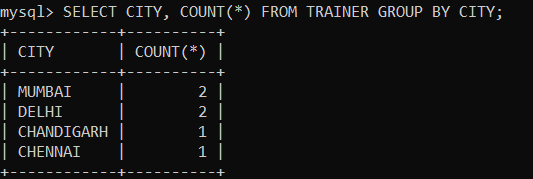
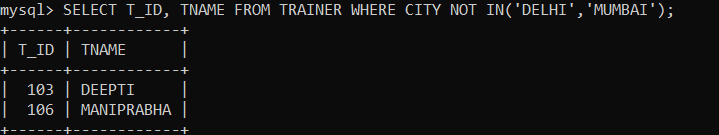
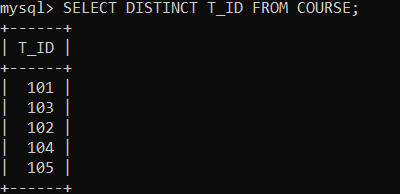
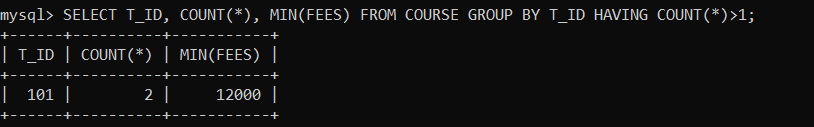
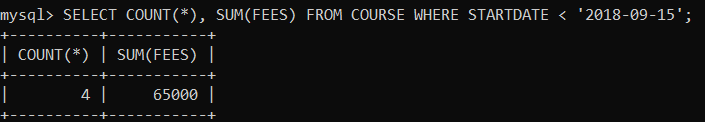
Q4. Write SQL queries and find the output of the SQL queries based on the tables TRAINER and COURSE.

**TRAINER**

****

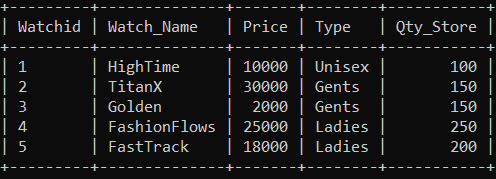
**COURSE**

****

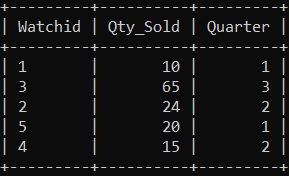
1. ****Display the Trainer Name, City & Salary in descending order of their Hiredate.
2. ****To display TNAME, HIREDATE, CNAME, STARTDATE from tables TRAINER and COURSE of all those courses whose FEES is less than or equal to 10000.
3. ****To display number of Trainers from each city.
4. ****SELECT T\_ID, TNAME, FROM TRAINER WHERE CITY NOT IN(‘DELHI’, ‘MUMBAI’);
5. ****SELECT DISTINCT T\_ID FROM COURSE;
6. ****SELECT T\_ID, COUNT(\*), MIN(FEES) FROM COURSE GROUP BY T\_ID HAVING COUNT(\*)>1;
7. SELECT COUNT(\*), SUM(FEES) FROM COURSE WHERE STARTDATE< ‘2018-09-15’;

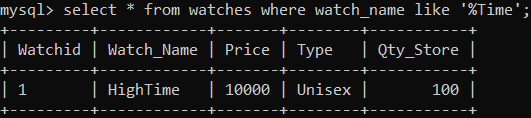
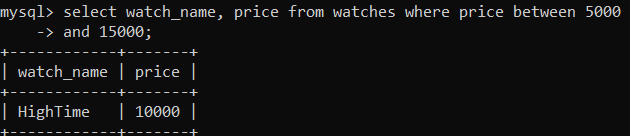
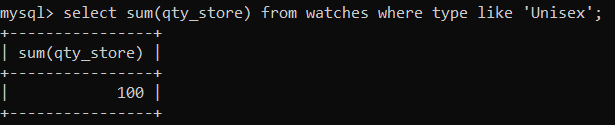
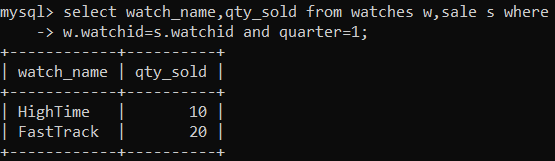
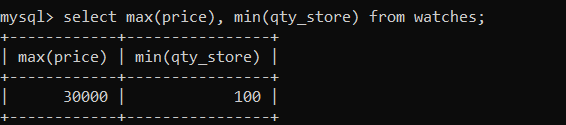
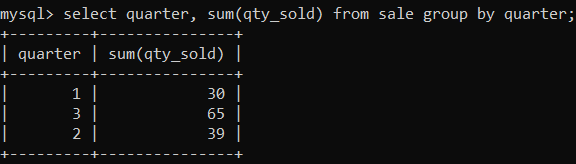
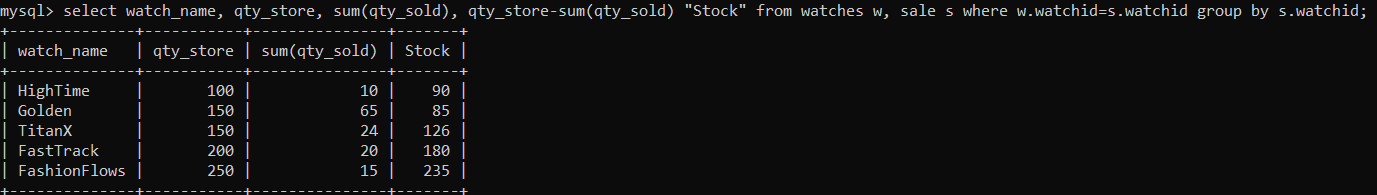
Q5. Write SQL commands for the queries and output of the queries based on the tables WATCHES and SALE.

**WATCHES**

****

**SALE**

****

1. To display all the details of those watches whose name ends with ‘Time’.
2. To display watch’s name and price of those watches which have price range in between 5000-15000.
3. To display total quantity in store of Unisex type watches.
4. To display watch name and their quantity sold in first quarter.
5. select max(price), min(qty\_store) from watches;
6. select quarter, sum(qty\_sold) from sale group by quarter;
7. select watch\_name, qty\_store, sum(qty\_sold), qty\_store-sum(qty\_sold) “Stock” from watches w, sale s where w.watchid=s.watchid group by s.watchid;