|  |  |
| --- | --- |
| 1. | Create Table Name : Student and Exam  python_assignment_1699259813.pdf - Personal - Microsoft​ Edge Beta |
|  | Ans : |
|  | CREATE TABLE student  (  Rollno int not null PRIMARY KEY,  Name varchar(25),  Branch varchar(25)  ); |
|  | INSERT INTO student VALUES(1,'Jay','Computer Science');  INSERT INTO student VALUES(2,'Suhani','Electronic and com');  INSERT INTO student VALUES(3,'Kriti','Electorinic and com'); |
|  | CREATE TABLE exam  (  Rollno int NOT null,  S\_code varchar(30),  Marks int,  P\_code varchar(30),  Rollno int,  FOREIGN KEY(Rollno) REFERENCES student(Rollno),  ); |
|  | INSERT INTO exam VALUES(1,'CS11',50,'CS');  INSERT INTO exam VALUES(1,'CS12',60,'CS');  INSERT INTO exam VALUES(2,'EC101',66,'EC');  INSERT INTO exam VALUES(2,'EC102',70,'EC');  INSERT INTO exam VALUES(3,'EC101',45,'EC');  INSERT INTO exam VALUES(3,'EC102',50,'EC'); |
|  |  |
|  |  |
| 2. | Create table given below: Employee and IncentiveTable |
|  |  |
|  | Ans :  CREATE TABLE Employee  (  Employee\_id int NOT null PRIMARY KEY,  First\_name varchar(25),  Last\_name varchar(25),  Salary int,  Joining\_date varchar(30),  Department varchar(30)  ); |
|  | INSERT INTO employee VALUES (1,'John','Abraham',1000000,'01-JAN-03 12.00.00 AM','Banking');  INSERT INTO employee VALUES (2,'Michael','Clarke',800000,'01-JAN-03 12.00.00 AM','Insurance');  INSERT INTO employee VALUES (3,'Roy','Thomas',700000,'01-FEB-03 12.00.00 AM','Banking');  INSERT INTO employee VALUES (4,'Tom','Jose',600000,'01-FEB-03 12.00.00 AM','Insurance');  INSERT INTO employee VALUES (5,'Jerry','Pinto',650000,'01-FEB-03 12.00.00 AM','Insurance');  INSERT INTO employee VALUES (6,'Philip','Mathew',750000,'01-JAN-03 12.00.00 AM','Services');  INSERT INTO employee VALUES (7,'Testname1','123',650000,'01-JAN-03 12.00.00 AM','Services');  INSERT INTO employee VALUES (8,'Testname2','Lname%',600000,'01-FEB-03 12.00.00 AM','Insurance'); |
|  | CREATE TABLE Incentive  (  Employee\_id int,  Incentive\_date varchar(30),  Incentive\_amount int,  FOREIGN KEY(Employee\_id) REFERENCES employee(Employee\_id)  ); |
|  | INSERT INTO incentive VALUES (1,'01-FEB-2013',5000);  INSERT INTO incentive VALUES (2,'01-FEB-2013',3000);  INSERT INTO incentive VALUES (3,'01-FEB-2013',4000);  INSERT INTO incentive VALUES (1,'01-JAN-2013',4500);  INSERT INTO incentive VALUES (2,'01-JAN-2013',3500); |
|  |  |
|  |  |
| 3. | Get First\_Name from employee table using Tom name “Employee Name”. |
|  | Ans : |
|  | SELECT First\_name FROM employee WHERE First\_name='Tom'; |
|  |  |
|  |  |
|  |  |
| 4. | Get FIRST\_NAME, Joining Date, and Salary from employee table. |
|  | Ans : |
|  | SELECT First\_name ,Joining\_date ,Salary FROM employee; |
|  |  |
|  |  |
|  |  |
| 5. | Get all employee details from the employee table order by First\_Name Ascending and Salary descending? |
|  | Ans : |
|  | SELECT \* FROM Employee ORDER BY First\_name ,Salary DESC; |
|  |  |
|  |  |
|  |  |
| 6. | Get employee details from employee table whose first name contains ‘J’. |
|  | Ans : |
|  | SELECT \* FROM Employee WHERE First\_name LIKE 'J%'; |
|  |  |
|  |  |
|  |  |
| 7. | Get department wise maximum salary from employee table order by salaryascending? |
|  | Ans : |
|  | SELECT Department, MAX(Salary) FROM employee GROUP BY Department; |
|  |  |
|  |  |
|  |  |
| 9. | Select first\_name, incentive amount from employee and incentives table forthose employees who have incentives and incentive amount greater than 3000 |
|  | Ans : |
|  | SELECT First\_Name, Incentive\_amount FROM employee  JOIN incentive ON employee.Employee\_id=incentive.Employee\_id WHERE incentive.Incentive\_amount>3000; |
|  |  |
|  |  |
|  |  |
| 10. | Create After Insert trigger on Employee table which insert records in viewtable. |
|  | Ans : |
|  | CREATE TABLE viewtable  (  Employee\_id int,  First\_name varchar(20),  Last\_name varchar(20),  Salary int,  Joining\_date varchar(20),  Department varchar(20),  Date\_Time timestamp,  Action\_Performed text  ); |
|  | DELIMITER $$  CREATE TRIGGER employee1 AFTER INSERT ON employee FOR EACH ROW  BEGIN  INSERT INTO viewtable(Employee\_ID,First\_name,Last\_name,Salary,Joining\_date,Department,  action\_performed)  VALUES(new.Employee\_ID,new.First\_name,new.Last\_name,new.Salary,new.Joining\_date,  new.Department,'Record inserted');  END |
|  | INSERT INTO employee VALUES(9,'James','Bond',800000,'1-Jan-13 12.00.00 AM','Banking'); |
|  |  |
|  |  |
|  |  |
| 11. | Create table given below: Salesperson and Customer |
|  |  |
|  | Ans : |
|  | CREATE TABLE Salesperson  (  PKSNo int PRIMARY KEY,  SNAME varchar(20),  CITY varchar(20),  COMM varchar(20)  ) |
|  | INSERT INTO salesperson VALUES(1001,'Peel','London','.12');  INSERT INTO salesperson VALUES(1002,'Serres','San Jose','.13');  INSERT INTO salesperson VALUES(1004,'Motika','London','.11');  INSERT INTO salesperson VALUES(1007,'Rafkin','Barcelona','.15');  INSERT INTO salesperson VALUES(1003,'Axelrod','New York','.1'); |
|  |  |
|  | CREATE TABLE customer  (  PKCNM int,  CNAME varchar(20),  CITY varchar(20),  RATING int,  PKSNo int,  FOREIGN KEY(PKSNo) REFERENCES salesperson(PKSNo)  ); |
|  | INSERT INTO customer VALUES(201,'Hoffman','London',100,1001);  INSERT INTO customer VALUES(202,'Giovanne','Roe',200,1003);  INSERT INTO customer VALUES(203,'Liu','San Jose',300,1002);  INSERT INTO customer VALUES(204,'Grass','Barcelona',100,1002);  INSERT INTO customer VALUES(206,'Clemens','London',300,1007);  INSERT INTO customer VALUES(207,'Pereira','Reo',100,1004); |
|  |  |
|  |  |
| 13. | All orders for more than $1000. |
|  | Ans : |
|  | SELECT \* FROM customer WHERE amount>1000; |
|  |  |
| 14. | Names and cities of all salespeople in London with commission above 0.12 |
|  | Ans : |
|  | SELECT SNAME, CITY FROM salesperson WHERE CITY='London' AND COMM>0.12; |
|  |  |
|  |  |
| 15. | All salespeople either in Barcelona or in London |
|  | Ans : |
|  | SELECT \* FROM salesperson WHERE CITY='Barcelona' OR CITY='London'; |
|  |  |
|  |  |
| 16. | All salespeople with commission between 0.10 and 0.12. (Boundary valuesshould be excluded). |
|  | Ans : |
|  | SELECT \* FROM salesperson WHERE COMM BETWEEN 0.10 AND 0.12; |
|  |  |
|  |  |
| 17. | All customers excluding those with rating <= 100 unless they are located in Rome. |
|  | Ans : |
|  | SELECT \* FROM customer WHERE rating<=100 AND CITY='Reo'; |
|  |  |
|  |  |
| 18. | Write a SQL statement that displays all the information about all salespeople. |
|  | Ans : |
|  | SELECT \* FROM salesperson; |
|  |  |
|  |  |
| 19. | From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord\_no, ord\_date, purch\_amt. |
|  |  |
|  | Ans : |
|  | CREATE TABLE salesman  (  salesman\_id int PRIMARY KEY,  name varchar(20),  city varchar(20),  commission varchar(20)  ); |
|  | INSERT INTO salesman VALUES (5001,'James Hoog','New York','0.15');  INSERT INTO salesman VALUES (5002,'Nail Knite','Paris ','0.13');  INSERT INTO salesman VALUES (5005,'Pit Alex','London','0.11');  INSERT INTO salesman VALUES (5006,'Mc Lyon','Paris','0.14');  INSERT INTO salesman VALUES (5007,'Paul Adam','Rome','0.13');  INSERT INTO salesman VALUES (5003,'Lauson Hen','San Jose','0.12'); |
|  | CREATE TABLE orders  (  ord\_no int PRIMARY KEY,  purch\_amt varchar(20),  ord\_date date,  customer\_id int,  salesman\_id int,  FOREIGN KEY(salesman\_id) REFERENCES salesman(salesman\_id)  ); |
|  | INSERT INTO orders VALUES (70001,'150.5','2012-10-05','3005','5002');  INSERT INTO orders VALUES (70009,'270.65','2012-09-10','3001','5005');  INSERT INTO orders VALUES (70002,'65.26','2012-10-05','3002','5001');  INSERT INTO orders VALUES (70004,'110.5','2012-08-17','3009','5003');  INSERT INTO orders VALUES (70007,'948.5','2012-09-10','3005','5002');  INSERT INTO orders VALUES (70005,'2400.6','2012-07-27','3007','5001');  INSERT INTO orders VALUES (70008,'5760','2012-09-10','3002','5001');  INSERT INTO orders VALUES (70010,'1983.43','2012-10-10','3004','5006');  INSERT INTO orders VALUES (70003,'2480.4','2012-10-10','3009','5003');  INSERT INTO orders VALUES (70012,'250.45','2012-06-27','3008','5002');  INSERT INTO orders VALUES (70011,'75.29','2012-08-17','3003','5007');  INSERT INTO orders VALUES (70013,'3045.6','2012-04-25','3002','5001'); |
|  |  |
|  | SELECT ord\_no, ord\_date, purch\_amt FROM orders WHERE salesman\_id=5001; |
|  |  |
|  |  |
| 20. | From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro\_id, pro\_name, pro\_price, and pro\_com. |
|  | Ans : |
|  | CREATE TABLE item\_mast  (  PRO\_ID int,  PRO\_NAME varchar(20),  PRO\_PRICE varchar(20),  PRO\_COM int  ); |
|  | INSERT INTO item\_mast VALUES (101,'Mother Board','3200.00',15);  INSERT INTO item\_mast VALUES (102,'Key Board','450.00',16);  INSERT INTO item\_mast VALUES (103,'ZIP drive','250.00',14);  INSERT INTO item\_mast VALUES (104,'Speaker','550.00',16);  INSERT INTO item\_mast VALUES (105,'Monitor','5000.00',11);  INSERT INTO item\_mast VALUES (106,'DVD drive','900.00',12);  INSERT INTO item\_mast VALUES (107,'CD drive','800.00',12);  INSERT INTO item\_mast VALUES (108,'Printer','2600.00',13);  INSERT INTO item\_mast VALUES (109,'Refill cartridge','350.00',13);  INSERT INTO item\_mast VALUES (110,'Mouse','250.00',12); |
|  |  |
|  | SELECT pro\_id, pro\_name, pro\_price, pro\_com FROM item\_mast WHERE pro\_price BETWEEN 200 AND 600; |
|  |  |
|  |  |
| 21. | From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg. |
|  | Ans : |
|  | SELECT AVG(PRO\_PRICE) AS avg FROM item\_mast WHERE PRO\_COM=16; |
|  |  |
|  |  |
| 22. | From the following table, write a SQL query to display the pro\_name as 'Item Name' and pro\_priceas 'Price in Rs.' |
|  | Ans : |
|  | SELECT PRO\_NAME AS 'Item Name', PRO\_PRICE AS'Price in Rs.' FROM item\_mast; |
|  |  |
|  |  |
| 23. | From the following table, write a SQL query to find the items whose prices are higher than or equal to $250. Order the result by product price in descending, then product name in ascending. Return pro\_name and pro\_price. |
|  | Ans : |
|  | SELECT PRO\_NAME,PRO\_PRICE FROM item\_mast WHERE PRO\_PRICE>=250 ORDER BY PRO\_NAME ASC,PRO\_PRICE DESC; |
|  |  |
|  |  |
| 24. | From the following table, write a SQL query to calculate average price of the items for each company. Return average price and company code. |
|  | Ans : |
|  | SELECT AVG(PRO\_PRICE)AS average\_price,PRO\_COM AS companycode FROM item\_mast GROUP BY PRO\_COM; |