**Create Table**

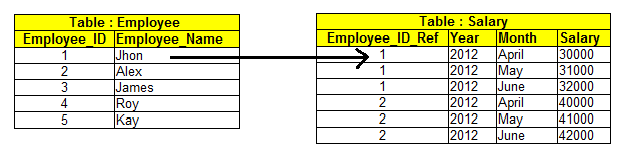
1.Write a SQL statement to create a simple table countries including columns country\_id,country\_name and region\_id

2.Write a SQL statement to create a table countries set a constraint NULL

3.create table locations including columns.

|  |  |
| --- | --- |
| LOCATION\_ID | decimal(4,0) |
| STREET\_ADDRESS | varchar(40) |
| POSTAL\_CODE | varchar(12) |
| CITY | varchar(30) |
| STATE\_PROVINCE | varchar(25) |
| COUNTRY\_ID | varchar(2) |

4.



**Alter Table**

1.Write a SQL statement to rename the table countries to country\_new.

2.Write a SQL statement to add a columns ID as the first column of the table locations

3.Write a SQL statement to add a column region\_id after state\_province to the table locations

4.Write a SQL statement change the data type of the column country\_id to integer in the table locations

**Insert Table**

1.Write a SQL statement to insert 3 rows by a single insert statement

2.Write a SQL statement to insert rows into the table countries in which the value of country\_id column will be unique and auto incremented

3.Write a SQL statement to insert rows only for country\_id and country\_name.

**Update Table**

**Employees Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EMPLOYEE\_ID | FIRST\_NAME | EMAIL | SALARY | COMMISSION\_PCT | DEPARTMENT\_ID |
| 100 | Steven | SKING | 24000.00 | 0.00 | 90 |
| 101 | Neena | NKOCHHAR | 1700.00 | 0.00 | 90 |
| 102 | Lex | LDEHAAN | 17000.00 | 0.00 | 90 |
| 103 | Alexander | AHUNOLD | 9000.00 | 0.00 | 60 |
| 104 | Bruce | BERNST | 6000.00 | 0.00 | 60 |
| 105 | David | DAUSTIN | 4800.00 | 0.00 | 60 |
| 106 | Valli | VPATABAL | 4200.00 | 0.00 | 60 |
| 107 | Diana | DLORENTZ | 12008.00 | 0.00 | 110 |
| 205 | Shelley | SHIGGINS | 8300.00 | 0.00 | 110 |
| 206 | William | WGIETZ | 8300.00 | 0.00 | 110 |

1.Write a SQL statement to change the email and commission\_pct column of employees table with 'not available' and 0.10 for all employees

2.Write a SQL statement to change the email and commission\_pct column of employees table with 'not available' and 0.10 for those employees whose department\_id is 110.

3.Write a SQL statement to change salary of employee to 8000 whose ID is 105, if the existing salary is less than 5000.

4. Write a SQL statement to increase the minimum and maximum salary of PU\_CLERK by 2000 as well as the salary for those employees by 20% and commission percent by .10.

**Select.**

**Salesman Table**

|  |  |  |  |
| --- | --- | --- | --- |
| salesman\_id | name | city | commission |
| 5001 | James Hoog | New York | 0.15 |
| 5002 | Nail Knite | Paris | 0.13 |
| 5005 | Pit Alex | London | 0.11 |
| 5006 | Mc Lyon | Paris | 0.14 |
| 5003 | Lauson Hen |  | 0.12 |
| 5007 | Paul Adam | Rome | 0.13 |

**Orders Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ord\_no | purch\_amt | ord\_date | customer\_id | salesman\_id |
| 70001 | 150.5 | 05-10-2012 | 3005 | 5002 |
| 70009 | 270.65 | 10-09-2012 | 3001 | 5005 |
| 70002 | 65.26 | 05-10-2012 | 3002 | 5001 |
| 70004 | 110.5 | 17-08-2012 | 3009 | 5003 |
| 70007 | 948.5 | 10-09-2012 | 3005 | 5002 |
| 70005 | 2400.6 | 27-07-2012 | 3007 | 5001 |
| 70008 | 5760 | 10-09-2012 | 3002 | 5001 |
| 70010 | 1983.43 | 10-10-2012 | 3004 | 5006 |
| 70003 | 2480.4 | 10-10-2012 | 3009 | 5003 |
| 70012 | 250.45 | 27-06-2012 | 3008 | 5002 |
| 70011 | 75.29 | 17-08-2012 | 3003 | 5007 |
| 70013 | 3045.6 | 25-04-2012 | 3002 | 5001 |

**Customers Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| customer\_id | cust\_name | city | grade | salesman\_id |
| 3002 | Nick Rimando | New York | 100 | 5001 |
| 3005 | Graham Zusi | California | 200 | 5002 |
| 3001 | Brad Guzan | London |  | 5005 |
| 3004 | Fabian Johns | Paris | 300 | 5006 |
| 3007 | Brad Davis | New York | 200 | 5001 |
| 3009 | Geoff Camero | Berlin | 100 | 5003 |
| 3008 | Julian Green | London | 300 | 5002 |
| 3003 | Jozy Altidor | Moscow | 200 | 5007 |

**Products Table**

|  |  |  |  |
| --- | --- | --- | --- |
| PRO\_ID | PRO\_NAME | PRO\_PRICE | PRO\_COM |
| 101 | Mother Board | 3200 | 15 |
| 102 | Key Board | 450 | 16 |
| 103 | ZIP drive | 250 | 14 |
| 104 | Speaker | 550 | 16 |
| 105 | Monitor | 5000 | 11 |
| 106 | DVD drive | 900 | 12 |
| 107 | CD drive | 800 | 12 |
| 108 | Printer | 2600 | 13 |
| 109 | Refill cartridge | 350 | 13 |
| 110 | Mouse | 250 | 12 |

**Departments table**

|  |  |  |  |
| --- | --- | --- | --- |
| DEPARTMENT\_ID | DEPARTMENT\_NAME | MANAGER\_ID | LOCATION\_ID |
| 10 | Administration | 200 | 1700 |
| 20 | Marketing | 201 | 1800 |
| 30 | Purchasing | 114 | 1700 |
| 40 | Human Resources | 204 | 2400 |
| 50 | Shipping | 11 | 1500 |
| 60 | IT | 103 | 1400 |
| 70 | Public Relations | 204 | 2700 |
| 80 | Sales | 145 | 2500 |
| 90 | Executive | 100 | 1700 |
| 100 | Finance | 108 | 1700 |
| 110 | Accounting | 205 | 1700 |

1.Write a SQL statement to display all the information of all salesmen

2.Write a SQL statement to display specific columns like name and commission for all the salesmen

3.Write a SQL statement to display names and city of salesman, who belongs to the city of Paris

4.Write a query to display the columns in a specific order like order date, salesman id, order number and purchase amount from for all the orders

5.Write a query which will retrieve the value of salesman id of all salesmen, getting orders from the customers in orders table without any repeats

6. Write a SQL statement to display all the information for those customers with a grade of 200

7. 1.Write a SQL query to find all the products with a price between Rs.200 and Rs.600

**Aggregation Function**

1.Write a SQL statement to find the total purchase amount of all orders.

2.Write a SQL statement to find the average purchase amount of all orders.

3.Write a SQL statement to find the number of salesmen currently listing for all of their customers

4.Write a SQL statement know how many customer have listed their names.

5.Write a SQL statement to get the maximum purchase amount of all the orders

**Relational Operator**

1.Write a query to display all customers with a grade above 100

2.Write a query statement to display all customers in New York who have a grade value above 100

3.Write a SQL statement to display all customers, who are either belongs to the city New York or had a grade above 100

4.Write a SQL statement to display either those orders which are not issued on date 2012-09-10 and issued by the salesman whose ID is 505 and below or those orders which purchase amount is 1000.00 and below.

**SORTING and FILTERING**

1.Write a query in SQL to display the full name (first and last name), and salary for those employees who earn below 6000

2.Write a query in SQL to display the first and last name, and department number for all employees whose last name is "Ernst".

3.Write a query in SQL to display the full name (first and last), salary, and department number for those employees whose first name does not containing the letter M and make the result set in ascending order by department number

4.Write a query in SQL to display the full name (first and last name), and salary for all employees who does not earn any commission

**Subqueries**

1.Write a query to display all the orders from the orders table issued by the salesman 'Paul Adam'

2.Write a query to display all the orders for the salesman who belongs to the city London

3. Write a query to find all the orders issued against the salesman who works for customer whose id is 3007

4.Write a query to display the commission of all the salesmen servicing customers in Paris

**Joins**

1.Write a query in SQL to display the first name, last name, department number, and department name for each employee

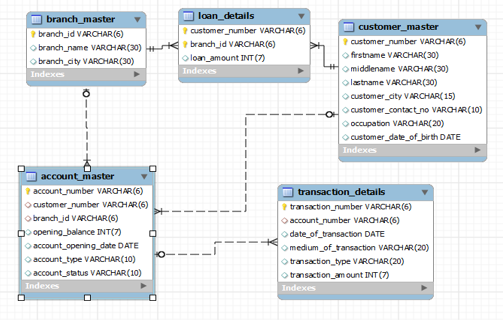
2.Write a query in SQL to display the first name, last name, department number and department name, for all employees for departments 80 or 40

3.Write a query in SQL to display the first name of all employees including the first name of their manager

4.Write a query in SQL to display all departments including those where does not have any employee

5.Write a query in SQL to display the first name, last name, department number and name, for all employees who have or have not any department

**Bank management System**



1.Write a query to display account number, customer’s number, customer’s firstname,lastname,account opening date.

Display the records sorted in ascending order based on account number.

SELECT account\_number,am.customer\_number,firstname,lastname,account\_opening\_date

FROM customer\_master cm INNER JOIN account\_master am

ON cm.customer\_number=am.customer\_number

ORDER BY account\_number

2.Write a query to display the numberof customer’s from Delhi. Give the count an alias name of Cust\_Count.

SELECT count(customer\_number) Cust\_Count

FROM customer\_master

WHERE customer\_city='Delhi'

3. Write a query to display the customer number, customer firstname,account number for the customer’s whose accounts were created after 15th of any month.

Display the records sorted in ascending order based on customer number.

SELECT am.customer\_number, firstname, account\_number

FROM customer\_master cm INNER JOIN account\_master am

ON cm.customer\_number=am.customer\_number

WHERE extract(day from account\_opening\_date)>15

ORDER BY am.customer\_number

4. Write a query to display customer number, customer's first name, account number where the account status is terminated.

Display the records sorted in ascending order based on customer number.

SELECT am.customer\_number,firstname, account\_number

FROM customer\_master cm INNER JOIN account\_master am

ON cm.customer\_number=am.customer\_number

WHERE account\_status='Terminated'

ORDER BY am.customer\_number.

5. Write a query to display the number of customers who have registration but no account in the bank.

Give the alias name as Count\_Customer for number of customers.

SELECT count(customer\_number) Count\_Customer

FROM customer\_master

WHERE customer\_number NOT IN (SELECT customer\_number FROM account\_master)

6. Write a query to display the firstname of the customers who have more than 1 account. Display the records in sorted order based on firstname.

Select firstname

FROM customer\_master cm INNER JOIN account\_master am

ON cm.customer\_number=am.customer\_number

group by firstname

having count(account\_number)>1

order by firstname;

7. Write a query to display the number of clients who have asked for loans but they don’t have any account in the bank though they are registered customers. Give the count an alias name of Count.

SELECT count(ld.customer\_number) Count

FROM customer\_master cm INNER JOIN loan\_detailsld

ON cm.customer\_number=ld.customer\_number

WHERE cm.customer\_number NOT IN ( SELECTcustomer\_number FROM account\_master)

8. Write a query to show the branch name,branch city where we have the maximum customers.

For example the branch B00019 has 3 customers, B00020 has 7 and B00021 has 10. So branch id B00021 is having maximum customers. If B00021 is Koramangla branch Bangalore, Koramangla branch should be displayed along with city name Bangalore.

In case of multiple records, display the records sorted in ascending order based on branch name.

selectbranch\_name,branch\_city

FROM branch\_master INNER JOIN account\_master

ON branch\_master.branch\_id=account\_master.branch\_id

group by branch\_name

9. Write a query to display the customer’s firstname who have multiple accounts (atleast 2 accounts). Display the records sorted in ascending order based on customer's firstname.

SELECT firstname

FROM customer\_master INNER JOIN account\_master

ON customer\_master.customer\_number=account\_master.customer\_number

GROUP BY firstname

having count(firstname)>=2 order by firstname;

10. Write a query to display account id, customer’s firstname, customer’slastname for the customer’s whose account is Active.

Display the records sorted in ascending order based on account id /account number.

SELECT account\_number, firstname, lastname

FROMcustomer\_master cm INNER JOIN account\_master am

ON cm.customer\_number=am.customer\_number

WHERE account\_status='Active'

ORDER BY account\_number.

**Jdbc Exercise(Using Statement and PreparedStatement)**

1.Let’s write code to insert a new record into the table Users with following details:

* + username: bill
  + password: secretpass
  + fullname: Bill Gates
  + email: bill.gates@microsoft.com

2. Select all records from the Users table and print out details for each record.

3. Write a code to update all the details of “Bill Gates”.

4. Write a code to delete a record whose username field contains “bill”.