SQL JOIN Assignment

select \* from orders;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ORD\_NO** | **PURCH\_AMT** | **ORD\_DATE** | **CUSTOMER\_ID** | **SALESMAN\_ID** |
| 70001 | 150.5 | 05-OCT-12 | 3005 | 5002 |
| 70009 | 270.65 | 10-SEP-12 | 3001 | 5005 |
| 70002 | 65.26 | 05-OCT-12 | 3002 | 5001 |
| 70004 | 110.5 | 17-AUG-12 | 3009 | 5003 |
| 70005 | 2400.6 | 27-JUL-12 | 3007 | 5001 |
| 70008 | 5760 | 10-SEP-12 | 3002 | 5001 |
| 70007 | 948.5 | 10-SEP-12 | 3005 | 5002 |

select \* from customer;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CUSTOMER\_ID** | **CUST\_NAME** | **CITY** | **GRADE** | **SALESMAN\_ID** |
| 3002 | Nick Rimando | New York | 100 | 5001 |
| 3007 | Brad Devis | New York | 200 | 5001 |
| 3005 | Graham Zusi | California | 200 | 5002 |
| 3008 | Julian Green | London | 300 | 5002 |
| 3004 | Fabian Johnson | Paris | 300 | 5006 |
| 3009 | Geoff Cameron | Berlin | 100 | 5003 |
| 3003 | Jozv Altidor | Moscow | 200 | 5007 |

select \* from salesman;

|  |  |  |  |
| --- | --- | --- | --- |
| **SALESMAN\_ID** | **NAME** | **CITY** | **COMMISSION** |
| 5001 | James Hoog | New York | .15 |
| 5002 | Nail Knite | Paris | .13 |
| 5005 | Pit Alex | London | .11 |
| 5007 | Paul Adam | Rome | .13 |
| 5003 | Lauson Hen | San Jose | .12 |
| 5006 | Mc Lyon | Paris | .14 |

Q.1From the above tables write a SQL query using joins to find the salesperson and customer who belongs to same city. Return Salesman, cust\_name and city.

-> Select salesman.name, customer.cust\_name, salesman.city from customer inner join salesman on customer.city = salesman.city;

|  |  |  |
| --- | --- | --- |
| **NAME** | **CUST\_NAME** | **CITY** |
| James Hoog | Nick Rimando | New York |
| Nail Knite | Fabian Johnson | Paris |
| Pit Alex | Julian Green | London |
| Mc Lyon | Fabian Johnson | Paris |

Q.2 From the above tables write a SQL query to find the salesperson(s) and the customer(s) he handle. Return Customer Name, city, Salesman, commission.

->Select salesman.name,customer.cust\_name, salesman.city, salesman.commission from customer inner join salesman on customer.salesman\_id = salesman.salesman\_id;

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME** | **CUST\_NAME** | **CITY** | **COMMISSION** |
| James Hoog | Brad Devis | New York | .15 |
| James Hoog | Nick Rimando | New York | .15 |
| Nail Knite | Julian Green | Paris | .13 |
| Nail Knite | Graham Zusi | Paris | .13 |
| Paul Adam | Jozv Altidor | Rome | .13 |
| Lauson Hen | Geoff Cameron | San Jose | .12 |
| Mc Lyon | Fabian Johnson | Paris | .14 |

Q.3From the above tables write a SQL query to find the details of an order. Return ord\_no, ord\_date, purch\_amt, Customer Name, grade, Salesman, commission

-> Select orders.ord\_no,orders.ord\_date,orders.purch\_amt, customer.cust\_name, customer.grade, salesman.name, salesman.commission from orders inner join customer on customer.customer\_id = orders.customer\_id inner join salesman on orders.salesman\_id = salesman.salesman\_id;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ORD\_NO** | **ORD\_DATE** | **PURCH\_AMT** | **CUST\_NAME** | **GRADE** | **NAME** | **COMMISSION** |
| 70001 | 05-OCT-12 | 150.5 | Graham Zusi | 200 | Nail Knite | .13 |
| 70002 | 05-OCT-12 | 65.26 | Nick Rimando | 100 | James Hoog | .15 |
| 70004 | 17-AUG-12 | 110.5 | Geoff Cameron | 100 | Lauson Hen | .12 |
| 70005 | 27-JUL-12 | 2400.6 | Brad Devis | 200 | James Hoog | .15 |
| 70008 | 10-SEP-12 | 5760 | Nick Rimando | 100 | James Hoog | .15 |
| 70007 | 10-SEP-12 | 948.5 | Graham Zusi | 200 | Nail Knite | .13 |

Q.4 From the above tables write a SQL query to find those orders where order amount exists between 500 and 2000. Return ord\_no, purch\_amt, cust\_name, city.

-> Select orders.ord\_no,orders.purch\_amt, customer.cust\_name, customer.city from orders inner join customer on customer.customer\_id = orders.customer\_id where orders.purch\_amt between 500 and 2000;

|  |  |  |  |
| --- | --- | --- | --- |
| **ORD\_NO** | **PURCH\_AMT** | **CUST\_NAME** | **CITY** |
| 70007 | 948.5 | Graham Zusi | California |

Q. 5 From the above tables write a SQL query to display the cust\_name, customer city, grade, Salesman, salesman city. The result should be ordered by ascending on customer\_id

-> select customer.cust\_name,customer.city, customer.grade, salesman.name, salesman.city from customer left join salesman on customer.salesman\_id = salesman.salesman\_id order by customer.customer\_id;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CUST\_NAME** | **CITY** | **GRADE** | **NAME** | **CITY** |
| Nick Rimando | New York | 100 | James Hoog | New York |
| Jozv Altidor | Moscow | 200 | Paul Adam | Rome |
| Fabian Johnson | Paris | 300 | Mc Lyon | Paris |
| Graham Zusi | California | 200 | Nail Knite | Paris |
| Brad Devis | New Yourk | 200 | James Hoog | New York |
| Julian Green | London | 300 | Nail Knite | Paris |
| Geoff Cameron | Berlin | 100 | Lauson Hen | San Jose |

Q.6Write a SQL statement to make a list in ascending order for the salesmen who works either for one or more customer or not yet join under any of the customers.

-> select customer.cust\_name, salesman.name from customer right join salesman on customer.salesman\_id = salesman.salesman\_id order by salesman. salesman\_id;

|  |  |
| --- | --- |
| **CUST\_NAME** | **NAME** |
| Nick Rimando | James Hoog |
| Brad Devis | James Hoog |
| Graham Zusi | Nail Knite |
| Julian Green | Nail Knite |
| Geoff Cameron | Lauson Hen |
| - | Pit Alex |
| Fabian Johnson | Mc Lyon |
| Jozv Altidor | Paul Adam |

Q.7 Write a SQL statement to make a report with customer name, city, order no., order date, purchase amount for those customers from the existing list who placed one or more orders or which order(s) have been placed by the customer who is not on the list.

-> Select orders.ord\_no, customer. cust name, customer.city,orders.ord\_date,orders.purch\_amt from orders left join customer on orders.customer\_id = customer.customer\_id;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ORD\_NO** | **CUST\_NAME** | **CITY** | **ORD\_DATE** | **PURCH\_AMT** |
| 70008 | Nick Rimando | New York | 10-SEP-12 | 5760 |
| 70002 | Nick Rimando | New York | 05-OCT-12 | 65.26 |
| 70005 | Brad Devis | New Yourk | 27-JUL-12 | 2400.6 |
| 70007 | Graham Zusi | California | 10-SEP-12 | 948.5 |
| 70001 | Graham Zusi | California | 05-OCT-12 | 150.5 |
| 70004 | Geoff Cameron | Berlin | 17-AUG-12 | 110.5 |
| 70009 | - | - | 10-SEP-12 | 270.65 |