Multi-Table Queries
SCO Database

Multi-Table Queries - Cartesian Product
SCO Database

There is no practical situation where one would use a Cartesian product. We show it here simply because it is the basis for an equijoin query, which is important

Multi-Table Queries - Cartesian Product
SCO Database

There is no practical situation where one would use a Cartesian product. We show it here simply because it is the basis for an equijoin query, which is important

```
SELECT *
FROM ORDERS, CUSTOMERS
```

Multi-Table Queries
SCO Database

Now we present multiple options for the same query

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Query 1 (equijoin, no table aliases)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Query 1 (equijoin, no table qualifiers)

```
SELECT Salesperson
FROM ORDERS,CUSTOMER
WHERE Name=Customer AND City='Charlotte'
```

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Query 2 (equijoin using table qualifiers)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Query 2 (equijoin using table qualifiers)

```
SELECT ORDERS.Salesperson
FROM ORDERS,CUSTOMER
WHERE CUSTOMER.Name=ORDERS.Customer AND City='Charlotte'
```

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Query 3 (Using table aliases)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte

Query 3 (Using table aliases)

```
SELECT o.Salesperson
FROM ORDERS o, CUSTOMER c
WHERE c.Name=o.Customer AND City='Charlotte'
```

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 4 (Using subquery with IN)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 4 (Using subquery with IN)

```
SELECT Salesperson
FROM ORDERS
WHERE Customer IN
(   SELECT Name
    FROM CUSTOMER
    WHERE City = 'Charlotte'
)
```

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 5 (using = instead of IN doesn't work because subquery returns more than 1 row)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 5 (using = instead of IN doesn't work because subquery returns more than 1 row)

```
SELECT Salesperson
FROM ORDERS
WHERE Customer =
(   SELECT Name
    FROM CUSTOMER
    WHERE City = 'Charlotte'
)
```

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 6 (using EXISTS)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 6 (using EXISTS)

```
SELECT Salesperson
FROM ORDERS
WHERE EXISTS
   ( SELECT *
     FROM CUSTOMER
     WHERE ORDERS.Customer = CUSTOMER.Name AND City = "Charlotte"
    )
```

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 7 (using JOIN)

Multi-Table Queries
SCO Database

Find the names of all Salespersons who have had an order with a customer from Charlotte.

Query 7 (using JOIN)

```
SELECT Orders.Salesperson
FROM CUSTOMER JOIN ORDERS on Orders.Customer = Customer.Name
WHERE  City = "Charlotte"
```