1. Create Tables

Persons Table

```
sql
Copy code
CREATE TABLE Persons (
    P_Id INT PRIMARY KEY,
    LastName VARCHAR(50),
    FirstName VARCHAR(50),
    Address VARCHAR(100),
    City VARCHAR(50)
);

INSERT INTO Persons (P_Id, LastName, FirstName, Address, City)
VALUES
(1, 'Hansen', 'Ola', 'Timoteivn 10', 'Sandnes'),
(2, 'Svendson', 'Tove', 'Borgvn 23', 'Sandnes'),
(3, 'Pettersen', 'Kari', 'Storgt 20', 'Stavanger');
```

Orders Table

```
sql
Copy code
CREATE TABLE Orders (
    O_Id INT PRIMARY KEY,
    OrderNo INT,
    P_Id INT
);

INSERT INTO Orders (O_Id, OrderNo, P_Id) VALUES
(1, 77895, 3),
(2, 44678, 3),
(3, 22456, 1),
(4, 24562, 1),
(5, 34764, 15);
```

2. INNER JOIN

Explanation

The `INNER JOIN` returns rows when there is at least one match in both tables.

Query

```
sql
Copy code
SELECT Persons.LastName, Persons.FirstName, Orders.OrderNo
FROM Persons
INNER JOIN Orders
ON Persons.P_Id = Orders.P_Id
ORDER BY Persons.LastName;
```

Result

LastName	FirstName	OrderNo
Hansen	Ola	22456
Hansen	Ola	24562
Pettersen	Kari	77895
Pettersen	Kari	44678

3. LEFT JOIN

Explanation

The `LEFT JOIN` returns all rows from the left table (`Persons`), even if there are no matches in the right table (`Orders`).

Query

```
sql
Copy code
SELECT Persons.LastName, Persons.FirstName, Orders.OrderNo
FROM Persons
LEFT JOIN Orders
ON Persons.P_Id = Orders.P_Id
```

```
ORDER BY Persons.LastName;
```

Result

LastName	FirstName	OrderNo
Hansen	Ola	22456
Hansen	Ola	24562
Pettersen	Kari	77895
Pettersen	Kari	44678
Svendson	Tove	NULL

4. RIGHT JOIN

Explanation

The `RIGHT JOIN` returns all rows from the right table (`Orders`), even if there are no matches in the left table (`Persons`).

Query

```
sql
Copy code
SELECT Persons.LastName, Persons.FirstName, Orders.OrderNo
FROM Persons
RIGHT JOIN Orders
ON Persons.P_Id = Orders.P_Id
ORDER BY Persons.LastName;
```

Result

LastName	FirstName	OrderNo
Hansen	Ola	22456
Hansen	Ola	24562
Pettersen	Kari	77895
Pettersen	Kari	44678
NULL	NULL	34764

5. FULL JOIN

Explanation

The `FULL JOIN` returns all rows when there is a match in one of the tables. If no match exists, `NULL` is returned for the missing values.

Query

```
sql
Copy code
SELECT Persons.LastName, Persons.FirstName, Orders.OrderNo
FROM Persons
FULL JOIN Orders
ON Persons.P_Id = Orders.P_Id
ORDER BY Persons.LastName;
```

Result

LastName	FirstName	OrderNo
Hansen	Ola	22456
Hansen	Ola	24562
Pettersen	Kari	77895
Pettersen	Kari	44678
Svendson	Tove	NULL
NULL	NULL	34764

1. CROSS JOIN

Explanation

The `CROSS JOIN` returns the Cartesian product of two tables, i.e., all possible combinations of rows from both tables.

Query

```
sql
Copy code
SELECT Persons.LastName, Persons.FirstName, Orders.OrderNo
FROM Persons
CROSS JOIN Orders
```

```
ORDER BY Persons.LastName;
```

Result

LastName	FirstName	OrderNo
Hansen	Ola	77895
Hansen	Ola	44678
Hansen	Ola	22456
Hansen	Ola	24562
Hansen	Ola	34764
Svendson	Tove	77895
Svendson	Tove	44678
Svendson	Tove	22456
Svendson	Tove	24562
Svendson	Tove	34764
Pettersen	Kari	77895
Pettersen	Kari	44678
Pettersen	Kari	22456
Pettersen	Kari	24562
Pettersen	Kari	34764

2. SELF JOIN

Explanation

The `SELF JOIN` is used to join a table to itself. It is often used for hierarchical data, such as an employee-manager relationship.

Query

```
sql
Copy code
SELECT A.LastName AS Employee, B.LastName AS Manager
FROM Persons A
LEFT JOIN Persons B
ON A.P_Id = B.P_Id - 1;
```

Result

Employee	Manager
Hansen	Svendson
Svendson	Pettersen
Pettersen	NULL

3. UPDATE JOIN

Explanation

An `UPDATE JOIN` updates rows in one table based on a condition in another table.

Query

Update the city of all persons who have placed orders.

```
sql
Copy code
UPDATE Persons
SET City = 'Updated City'
FROM Persons
INNER JOIN Orders
ON Persons.P_Id = Orders.P_Id;
```

Query to Verify

```
sql
Copy code
SELECT * FROM Persons;
```

Result

P_Id	LastName	FirstName	Address	City
1	Hansen	Ola	Timoteivn 10	Updated City
2	Svendson	Tove	Borgvn 23	Sandnes
3	Pettersen	Kari	Storgt 20	Updated City

4. DELETE JOIN

Explanation

A `DELETE JOIN` deletes rows from a table based on a condition in another table.

Query

Delete persons who have not placed any orders.

```
sql
Copy code
DELETE FROM Persons
FROM Persons
LEFT JOIN Orders
ON Persons.P_Id = Orders.P_Id
WHERE Orders.P_Id IS NULL;
```

Query to Verify

```
sql
Copy code
SELECT * FROM Persons;
```

Result

P_Id	LastName	FirstName	Address	City
1	Hansen	Ola	Timoteivn 10	Updated City
3	Pettersen	Kari	Storgt 20	Updated City