19. Create tables student_info and student_branch and Write SQL Query to perform following Join Operations

```
-- Create student_info table
CREATE TABLE student_info (
    student_id INT PRIMARY KEY,
    name VARCHAR(100),
    age INT,
    city VARCHAR(50)
);

-- Insert data into student_info table
INSERT INTO student_info VALUES (1, 'John Doe', 20, 'New York');
INSERT INTO student_info VALUES (2, 'Alice Smith', 22, 'Los Angeles');
INSERT INTO student_info VALUES (3, 'Bob Johnson', 21, 'Chicago');

-- Create student_branch table
CREATE TABLE student_branch (
    branch_id INT PRIMARY KEY,
    branch_name VARCHAR(50),
    student_id INT,
    FOREIGN KEY (student_id) REFERENCES student_info(student_id)
);

-- Insert data into student_branch table
INSERT INTO student_branch VALUES (1, 'Branch A', 1);
INSERT INTO student_branch VALUES (2, 'Branch B', 2);
INSERT INTO student_branch VALUES (3, 'Branch C', 3);

-- Join Operations
-- Cross Join
SELECT * FROM student_info CROSS JOIN student_branch;

-- Inner Join
SELECT * FROM student_info INNER JOIN student_branch ON student_info.student_id =
student_branch.student_id;
```

-- Right Outer Join

```
SELECT * FROM student_info RIGHT OUTER JOIN student_branch ON
student_info.student_id = student_branch.student_id;
```

-- Full Outer Join

```
SELECT * FROM student_info FULL OUTER JOIN student_branch ON
student_info.student_id = student_branch.student_id;
```