4.2 SQL Clauses

```sql
SELECT YEAR(DOB) AS BirthYear, COUNT(*) AS ProgrammerCount
FROM programmers
GROUP BY BirthYear
ORDER BY ProgrammerCount DESC
LIMIT 1;
```

SQL Clauses

In SQL, clauses are keywords used to perform specific actions on the data stored in databases. Some of the fundamental SQL clauses include the WHERE, GROUP BY, HAVING, and ORDER BY clauses. Each of these clauses serves a distinct purpose in querying and manipulating data.

1. WHERE Clause:

The WHERE clause is used to filter rows from a table based on a specified condition. It allows you to extract only those records that satisfy the given criteria.

Syntax:

```sql
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

Example Queries:

```sql
-- Select programmers whose primary language is Python
SELECT Programmer_Name, Primary_Language
FROM programmers
WHERE Primary_Language = 'Python';

-- Select programmers who joined after a specific date
SELECT Programmer_Name, DOJ
FROM programmers
WHERE DOJ > '2000-01-01';
```

2. GROUP BY Clause:

The GROUP BY clause is used to group rows that have the same values into summary rows, typically in conjunction with aggregate functions (like COUNT, SUM, AVG, etc.).

Syntax:

```sql
SELECT column1, aggregate_function(column2)
FROM table_name
GROUP BY column1;
```

Example Queries:

```sql
-- Count the number of programmers in each gender category
SELECT GENDER, COUNT(*)
FROM programmers
GROUP BY GENDER;

-- Calculate the total salary expenditure for each primary language
SELECT Primary_Language, SUM(Salary) AS Total_Salary
FROM programmers
GROUP BY Primary_Language;
```

3. HAVING Clause:

The HAVING clause is used to filter records that result from a GROUP BY clause. It enables you to apply conditions to the grouped data.

Syntax:

```sql
SELECT column1, aggregate_function(column2)
FROM table_name
GROUP BY column1
HAVING condition;
```

Example Queries:

```sql
-- Find primary languages with an average salary greater than 15000
SELECT Primary_Language, AVG(Salary) AS Avg_Salary
FROM programmers
GROUP BY Primary_Language
HAVING AVG(Salary) > 15000;

-- Show only those gender categories with more than 2 programmers
SELECT GENDER, COUNT(*)
FROM programmers
GROUP BY GENDER
HAVING COUNT(*) > 2;
```

4. ORDER BY Clause:

The ORDER BY clause is used to sort the result set in ascending or descending order based on one or more columns.

Syntax:

```sql
SELECT column1, column2, ...
FROM table_name
ORDER BY column1 [ASC|DESC], column2 [ASC|DESC], ...;
```

Example Queries:

```sql
-- Sort programmers by their salaries in descending order
SELECT Programmer_Name, Salary
FROM programmers
ORDER BY Salary DESC;

-- Sort courses offered by institutes in alphabetical order
```

```
SELECT Institute, Course
FROM studies
ORDER BY Course;
```

Conclusion:

SQL clauses are essential components of querying and manipulating data in databases. Understanding how to use WHERE, GROUP BY, HAVING, and ORDER BY clauses enables you to perform various operations efficiently on large datasets. Each clause serves a specific purpose and can be combined to create complex and insightful queries.