**Basic selection instruction**

* **SELECT statement** specifies what is to be retrieved (target)
* **FROM statement** specifies tables to be queried (source/range)
* **WHERE statement** specifies restriction on values to be processed (predicate)
* **SELECT COUNT(\*) FROM table WHERE condition** counts the output records

**Scenarios:**

1. **Requirement**: Get all names of students that major in ‘Games’ from the Student table

**SELECT** stuname **FROM** Student **WHERE** major = ‘Games’;

1. **Requirement:** Get student names who enrolled course COMP2031

**SELECT** stuname FROM Student, Enrolled   
**WHERE** subjectId = ‘COMP2031’   
**AND** Student.studentId = Enrolled.studentId;

**//Since Student table does not have subjectId, we need to link the table with the other table that has it together by matching the foreign key (studentId)**

**//by connecting 2 tables, the table will share attributes from another table**

1. **Requirement:** List the names of the staff and the name of the department they work with.

SELECT name **(belongs to staffmember)**, dname **(belongs to dept)**

FROM Staffmember, Department

WHERE Staffmember.deptID = Department.deptID

**IntroSQL Task1 Solution:**

**Show which classes are being taught, who’s teaching and when**

**SELECT** name, sname, day, classtime, room

**FROM** Class, Subject, Staffmember

**WHERE** teacher = staffId

**AND** Subject.subjectId = Class.subjectId

**IntroSQL Task2 Solution:**

**Generate a list of students who are enrolled on module COMP9311 (Linking 2 tables tgt)**

**SELECT** stuname

**FROM** Student, Enrolled

**WHERE** subjectId = ’COMP9311’

**AND** Student.subjectId = Enrolled.subjectId**;**

**IntroSQL Task3 Solution:**

**Generate a class list for all classes taught for Esme Lettitia (staff number 989) (names of students and subject taught) (Linking 4 tables tgt)**

Since stuname, sname and teacher are from 3 different tables, we link the tables together by using studentId and subjectId that is shared outside tables.

We cannot link directly from Class to Subject because this only shares sname and teacher between two tables, and leave out stuname

**SELECT** stuname, sname

**FROM** Student, Enrolled, Class, Subject

**WHERE** teacher = 989

**AND** Class.subjectId = Enrolled.subjectId (ALTERNATIVE 1)

**AND** Class.subjectId = Subject.subjectId (ALTERNATIVE 2)

**AND** Student.studentid = Enrolled.studentid

**AND** Subject.subjectid = Enrolled.subjectid

Alternative path (as long as you   
connect all tables its fine)

Main entity of information