```sql
DROP TABLE IF EXISTS Student;

CREATE TABLE Student(
    sID INTEGER,
    sName VARCHAR(255) NOT NULL,
    sAddress VARCHAR(255),
    sYear INTEGER,
    CONSTRAINT pk_student
        PRIMARY KEY (sID)
);

DROP TABLE IF EXISTS Module;

CREATE TABLE Module(
    mCode VARCHAR(255) NOT NULL,
    mCredits INTEGER,
    mTitle VARCHAR(255) NOT NULL,
    CONSTRAINT pk_mo
        PRIMARY KEY (mCode)
);


DROP TABLE IF EXISTS Enrollment;

CREATE TABLE Enrollment(
    sID INTEGER,
    mCode VARCHAR(255),
    CONSTRAINT fk_en_stu
        FOREIGN KEY (sID) REFERENCES Student (sID),
    CONSTRAINT fk_en_mo,
        FOREIGN KEY (mCode) REFERENCES Module (mCode)
);

INSERT INTO Student
VALUES(1,"Smith","5 Arnold Close",2),
      (2,"Brooks","7 Holly Avenue",2),
      (3,"Anderson","15 Main Street",3),
      (4,"Evans","Flat 1a, High Street",2),
      (5,"Harrison","Newark Hall",1),
      (6,"Jones","Southwell Hall",1);


INSERT INTO Module
VALUES("G51DBS",10,"Database Systems"),
      ("G51PRG",20,"Programming"),
```

```
      ("G51IAI",10,"Artificial Intelligence"),
      ("G52ADS",10,"Algorithms");



INSERT INTO Enrollment
VALUES(1,"G52ADS"),
      (2,"G52ADS"),
      (5,"G51DBS"),
      (5,"G51PRG"),
      (5,"G51IAI"),
      (4,"G52ADS"),
      (6,"G51PRG"),
      (6,"G51IAI");

--Exercise2

--Q1
SELECT * FROM Student
    ORDER BY sYear DESC,sName ASC;

--Q2
SELECT mCode, COUNT (*) AS Number
FROM Enrollment, Student
WHERE Student.sID=Enrollment.sID
GROUP BY mCode;


--Q3
SELECT sID,
       sName
FROM Student
WHERE NOT EXISTS(
    SELECT *
    FROM Enrollment
    WHERE Student.sID=Enrollment.sID
);

--Q3.another way
SELECT sID,
       sName
FROM Student
WHERE Student.sID NOT IN(
    SELECT sID
    FROM Enrollment
```

```
);

/*
SELECT sID,
       sName
FROM Student
WHERE Student.sID NOT IN Enrollment.sID;
                        (should be a column)

*/

--Q4

SELECT sName, mTitle
FROM Enrollment NATURAL JOIN Module NATURAL Join Student
ORDER BY sID;

--Q5

SELECT Enrollment.mCode,mTitle,COUNT (*) AS Number
FROM Enrollment, Module, Student
WHERE Student.sID=Enrollment.sID
      AND
      Enrollment.mCode=Module.mCode
GROUP BY Enrollment.mCode;
--HOWEVER, IT CAN NOT SHOW 0 S

--Q5.2
SELECT Enrollment.mCode,mTitle,COUNT(Student.sID) AS Number
FROM Enrollment
LEFT OUTER JOIN Module
ON Enrollment.mCode=Module.mCode
LEFT OUTER JOIN Student
ON Enrollment.sID= Student.sID
GROUP BY Enrollment.mCode;

--Q6
SELECT sName AS name,
       SUM(mCredits) AS total
       FROM Student, Module, Enrollment
       WHERE Student.sID=Enrollment.sID
             AND
             Module.mCode=Enrollment.mCode
GROUP BY sName
```

```
Order BY SUM(mCredits) DESC;
```