# Queries & Results

(All queries and csv files saved in Documents\CS1P)

## Query 1

SELECT JobTitle  
FROM Staff  
WHERE Name\_Forename = 'Stephen'   
AND Name\_Surname = 'Brewster';

**JobTitle**  
professor

JobsOfStephen := (JobTitle)(Name\_Forename = ‘Stephen’(Staff) Name\_Surname = ‘Brewster’(Staff))

## Query 2

SELECT ExamMark  
FROM Student\_Takes\_Course, Course  
WHERE Title = 'CS-1Q'   
AND Student\_Takes\_Course.Course\_fk = Course.CourseID

**ExamMark**  
NULL  
88  
47  
32

CS1Q := Title=’CS-1Q’(Course)  
CS1QExams = (ExamMark)(Student\_Takes\_Course Course\_fk=Course CS1Q)

## Query 3

SELECT Name\_Forename, Name\_Surname, ExamMark  
FROM Student\_Takes\_Course, Course, Student  
WHERE Course.Title = 'CS-1Q'   
AND Student\_Takes\_Course.Course\_fk = Course.CourseID   
AND Student.MatricNo = Student\_Takes\_Course.Student\_fk  
ORDER BY Name\_Surname ASC

**Name\_Forename,Name\_Surname,ExamMark**  
Jane,Jones,32  
Jane,Jones,NULL  
Sally,Saunders,47  
Sam,Smith,88

CS1Q := Title=’CS-1Q’(Course)  
CS1QAssignments := CS1Q  Course\_fk=CourseID Student\_Takes\_Course  
CS1QStudents := CS1QAssignments MatricNo=Student\_fk Student  
ShowCS1QStudents := (Name\_Forename, Name\_Surname, ExamMark)(CS1QStudents)

## Query 4

SELECT Staff.Name\_Forename, Staff.Name\_Surname  
FROM Course, Course\_TaughtBy\_Lecturer, Lecturer, Staff  
WHERE Course.Title = 'CS-1Q'   
AND Lecturer.StaffID = Staff.StaffNo   
AND JobTitle = 'professor'   
AND Course.CourseID = Course\_TaughtBy\_Lecturer.Course\_fk   
AND Course\_TaughtBy\_Lecturer.Lecturer\_fk = Lecturer.StaffID

**Name\_Forename,Name\_Surname**  
Chris,Johnson

CS1Q := Title=’CS-1Q’(Course)  
CS1QLecturers = CS1Q Course\_fk=CourseID Course\_TaughtBy\_Lecturer  
Professors := JobTitle=’professor’(Staff)  
StaffProfessors := Professors StaffID=StaffNo Lecturer  
CS1QStaffProfessors := CS1QLecturers Lecturer\_fk=StaffID Lecturer  
ShowCS1QProfessors := (Name\_Forename, ­­­Name\_Surname)(CS1QStaffProfessors)

## Query 5

SELECT JobTitle, count(StaffNo)  
FROM Staff  
GROUP BY JobTitle

**JobTitle,count(StaffNo)**  
lecturer,3  
professor,3  
research assistant,1  
research student,1  
senior lecturer,1

## Query 6

SELECT Name\_Forename, Name\_Surname  
FROM Tutor, Staff, Tutorial\_Group\_2  
WHERE Tutor.StaffID\_fk = Staff.StaffNo   
AND Tutorial\_Group\_2.Tutor\_fk = Tutor.StaffID\_fk   
AND Staff.JobTitle <> 'lecturer'   
AND Tutorial\_Group\_2.TutorialRoom = 11

**Name\_Forename,Name\_Surname**Alex,Ng

## Query 7

SELECT Student\_fk, AVG(ExamMark), AVG(PracticalMark)  
FROM Student\_Takes\_Course  
GROUP BY Student\_fk

**Student\_fk,AVG(ExamMark),AVG(PracticalMark)**1234000,55.0000,15.0000  
1235000,57.0000,11.0000  
1236000,81.0000,16.0000  
1237000,47.0000,10.0000  
1238000,47.0000,10.0000

## Query 8

SELECT Student\_fk, Student.Name\_Forename, Student.Name\_Surname, AVG(ExamMark), AVG(PracticalMark)  
FROM Student\_Takes\_Course, Student  
WHERE Student\_Takes\_Course.Student\_fk = Student.MatricNo AND (Student.Name\_Surname = 'Smith' OR Student.Name\_Surname = 'Saunders')  
GROUP BY Student\_fk

**Student\_fk,Name\_Forename,Name\_Surname,AVG(ExamMark),AVG(PracticalMark)**1236000,Sam,Smith,81.0000,16.00001237000,Sally,Saunders,47.0000,10.0000

## Query 9

SELECT Name\_Forename, Name\_Surname  
FROM Staff  
WHERE StaffNo = ManagedBy

**Name\_Forename,Name\_Surname**  
Muffy,Calder

# Challenges & Reflections

(i)

The design and building process specifically for constructing a database consists of three main stages. The first is requirements analysis, which is needed to learn about the data that must be stored and accessed by different users, i.e., what data is actually needed and going to be used. The second is the conceptual design, usually achieved by making ER diagrams, which helps understand the relationships apparent in the data and how best to store it. The conceptual design allows for much easier transition into the last stage: the logical design, which is the actual building of the relational schema. This transition is important because the end goal of making a database is having a real, working database, but it is achievable much more easily if the conceptual design is clear and understandable for all involved parties (users, clients, team of designers, programmers) in the design of the database.

(ii)

My biggest challenges while building and designing the database were the transitions from requirements analysis to conceptual design and from conceptual design to logical design. The first transition cost me a point from the marks because I did not yet understand the different data types available in databases, that is, I did not yet know that there is a tinyint data type for expressing boolean/binary values (the part-time attribute), so I created too many entity subtypes as per the instructions for entities having different attributes. Additionally, the second transition was challenging because I was still confused about many-to-many relationships being made into relations. To solve these issues I reread the appropriate lecture slides and got the database working in the end. While making the database in MySQL Workbench, I experienced some troubles with foreign keys, which were solved after hours of experimenting and rebuilding relations.

(iii)

I knew some basics of SQL before when building a Java Derby database, but this exercise helped me gain a solid understanding of the fundamental principles of SQL and the relational algebra behind it, with the additional knowledge of how to use a very powerful DBMS (MySQL Workbench) to its potential. This basic understanding can lead to future career prospects in data science for me, or it can be useful just as part of a broader collection of knowledge of computing.

Word Count: 374