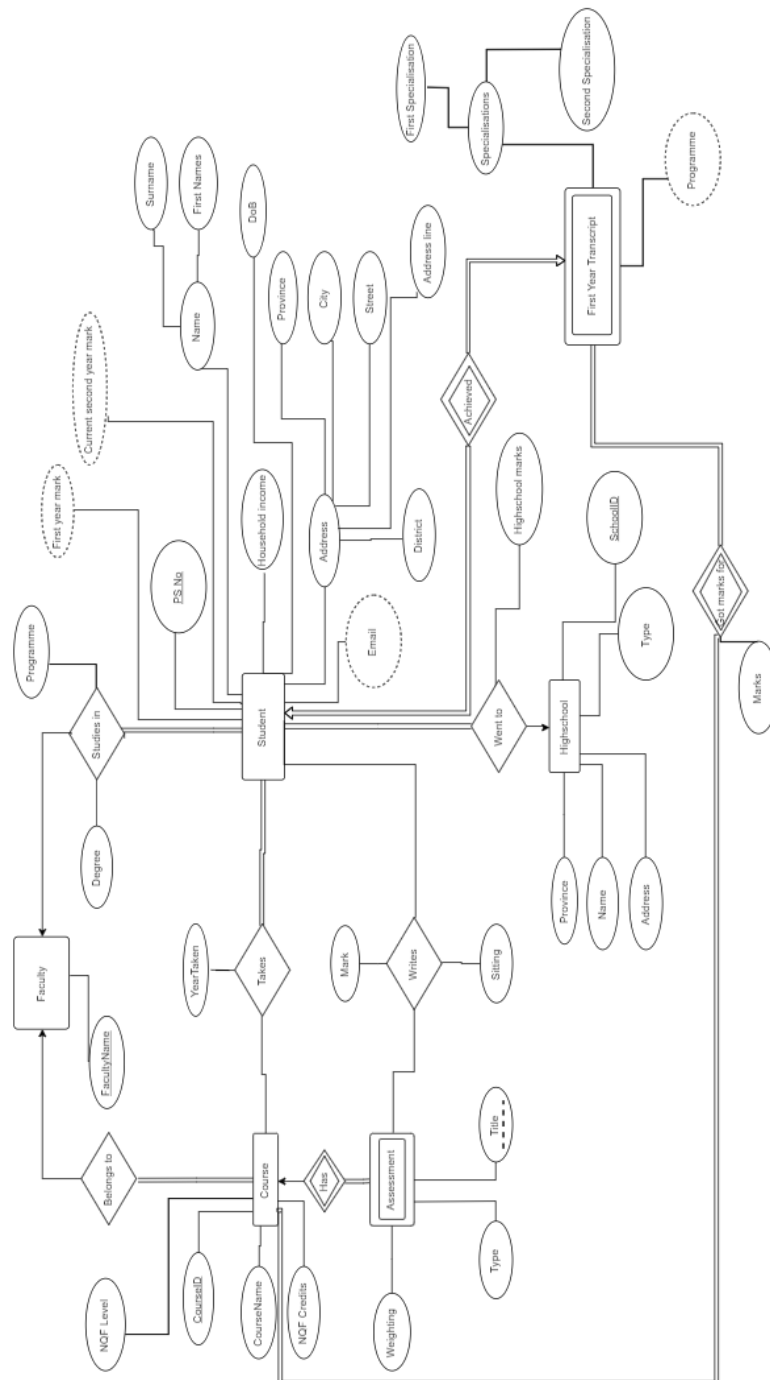


CMRJOS001, FRSMIC017, GNXDEL001, JVXANE002, KTHYUS001
Assignment 5 Group 18

ER diagram:



b) **Entities in database:**

Student:

Student(PS_Number, Student_ID , FirstNames, Last_Name, Household_Income, Street_Address, Area, Address_Line)

Tuple	Description
PS_Number	Unique identifier, in this case the peoplesoft identification number of the student
Student_Number	The Student Number of the student (Unique, secondary identifier format 'SSSNNNXXX') with SSS being the first three consonants of the students surname, nnn being the first three letters of their names and XXX referring to a number appended to the end of the unique identifier to ensure uniqueness
First_Names	The first name of the student
Last_Name	The last name of the student
Household_Income	The net annual household income of the student
Street_Address	The registered street address of the student (House Number, Road Name)
Area	The area in which the student lives
Address_Line	Any extra information referring to the residence of the student

Highschool:

Highschool(School_Reg_Number, Name , Type, Province, Street_Address, Area, Address_Line)

Tuple	Description
School_Reg_Number	The registration number of the highschool the student attended Used as a unique identifier
Name	The name of the school the student attended
Type	The type (Private/Public) of school the student attended
Province	The province that the school is in
Street_Address	The street address of the school
Area	The area in which the school is based
Address_Line	Any extra information referring to the address of the school

Course

Course(Course_Code, Course_Name, Credits)

Tuple	Description
Course_Code	The course code of the course, as used in the UCT system. Used as a unique identifier
Course_Name	The name of the course
Credits	The amount of credits the course is worth

Faculty

Faculty(Faculty_ID, Faculty_Name)

Tuple	Description
Faculty_ID	A unique identifier associated with each faculty
Faculty_Name	The name of the faculty

Assessments

The assessments table contains the assessments for a certain course their type (Assignment, Test, Examination) as well as their weighting overall

Assessments(Course_Code, Title, Type, Weighting)

Tuple	Description
Course_Code	The course with which the Assessment is associated, used as a partial key for the table
Title	The name of the test written, used as a partial key for the table
Type	The type of assessment (Test, Assignment, Exam)
Weighting	The contribution of the assessment to the final mark

FirstYearTranscript

The FirstYearTranscript table represents the transcript of the student attained in their first year of study

FirstYearTranscript(PS_Number, FirstSpecialisation, SecondSpecialisation)

Tuple	Description
Student_PS	The peoplesoft number of the student who got the transcript
FirstSpecialisation	The first specialisation of the student who achieved the transcript
SecondSpecialisation	The second specialisation of the student who achieved the transcript

Relationship tables:

HighSchoolMarks

The HighSchoolMarks table links the students to their respective highschools and contains the marks achieved by the students

HighSchoolMarks(Student_ID, School_Reg_Number, Mark)

Tuple	Description
Student_ID	Unique identifier of the student which went to the highschool (The PS ID)
School_Reg_Number	Unique identifier of the school which the student went to
Mark	The mark achieved by the student (Average)

StudentFaculty

The StudentFaculty table relates the student to the faculty in which they are studying and contains information on the degree they are studying for

StudentFaculty(Faculty_ID, Student_ID, Degree)

Tuple	Description
Faculty_ID	The unique identifier associated with the faculty in which the student is studying
Student_ID	The unique identifier associated with the student who is studying in the faculty
Degree	The degree which the student is studying for
Programme	The programme which is displayed on the transcript

StudentCourses

The StudentCourses table relates the student to the courses that they are taking, and the year in which they are taking them

StudentCourses(PS_Number, Course_Code, YearTaken)

Tuple	Description
Student_PS_No	A unique identifier referring to the student who is taking the course (The peoplesoft number)
Course_Code	The unique identifier referring to the course code
YearTaken	The year in which the student took the course (1/2)

CourseFaculty

The CourseFaculty table relates the Courses to the Faculty within which they exist

CourseFaculty(Course_Code, Faculty_ID)

Tuple	Description
Course_Code	The course code (unique identifier) of the course which is being related to the faculty
Faculty_ID	Unique identifier associated with the faculty which the course is being related to

StudentAssessments

The StudentAssessments table relates the students to the tests, containing their marks and the sitting in which they wrote each attempt for their second year courses so far in the year

StudentAssessments(Course_Code, Title, PS_Number, Mark, Sitting)

Tuple	Description
Course_Code	The unique Course_Code of the course which had the assessment
Title	The title of the assessment
Student_PS	The peoplesoft number associated with the student who wrote the test, refers to the primary key of the student
Mark	The mark achieved by the student
Sitting	The sitting in which the student achieved the mark

TranscriptMarks

The marks associated with each course on the students first year transcript

TranscriptMarks(Course_ID, Student_ID, Marks)

Tuple	Description
Course_ID	The unique identifier associated with the course for which the mark was achieved
Student ID	The unique identifier associated with the student who the transcript belongs to
Mark	The mark achieved with the course

c) An example of a functional dependency in this scenario would be the relationship between the Student Number and a Student Name.

d) A functional dependency is a relationship between two attributes, usually between a primary key and another non-key attribute within a table.

e) The integrity constraints to consider are domain constraints, referential-constraints and triggers. An example of a domain constraint would be that marks have to be a valid percentage i.e student marks should be greater or equal to zero and less than or equal to 100. An example of a referential-constraint would be that every reference to a Student ID should point to a valid Student ID in the database, henceforth, every foreign key in the database should point to a valid and existing primary key in another table. Finally, an example of a trigger would be that a change in a student's mark in their first year would automatically update the student's average on their transcript.

f) An example of a transitive dependency in this scenario would be the final-school-year results and first-year results at UCT as the relationship is dependent on the student ID.

g) An example of a partial functional dependency in this scenario would be the PS Number and the Students Name as a non-prime attribute depends functionally on a part of the primary key.

h) The High School Marks table is a many-to-many relationship and is in 3rd Normal Form. This is because any information referring to the highschool itself has been placed in a separate table (This being the Highschool table), this includes the name of the highschool attended, whether it is private or not and so forth, this data was instead grouped for each highschool and added to the Highschool table, this ensures uniformity and the use of minimal amounts of storage space.

i) Each member has contributed equally for this assignment. Each member designed their own ER diagram and we decided on one. All together, we answered the subsequent questions by choosing the best answers from the suggestions of our members.

j) (COMPLETED ASSIGNMENT 5 EVALUATION ON VULA)

k)

