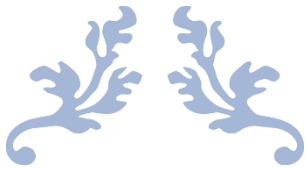


ABERTUTORS

CS27020 Assignment

NOVEMBER 8, 2019

MARCOS GARCÍA JUÁREZ
mag95@aber.ac.uk

1. UNF

Tutor Name CHAR(24)	**Tutor Contact** CHAR(254)	**Subject** CHAR(24)	**Location** CHAR(24)
Jackie	01970666543	Math English	Aberystwyth
Ahmed	07723456788	Cymraeg Art	Borth Aberystwyth Bow Street Llandre
Jeff	jeff@stemtutors.net	Math Science	Machylleth Talybont Bow Street
Dafydd	07845333444	Cymraeg	Machylleth
Peter	pete@languagetutors.com	English	Bow Street
Brian	brian@arttutors.org	Art	Aberystwyth Llarhystud

Name CHAR(24)	**Contact Details** CHAR(254)	**Subjects Wanted** CHAR(24)	**Location** CHAR(24)
Elin Davies	elin@example.com	Maths, Science	Machylleth
Steve Smith	steve@example.com	English	Llarhystud

In the case of Names, Subjects, Subjects Wanted and Location the Datatype CHAR(24) should fulfil any required attribute given, or at least an abbreviation of its value, as longer strings would just make it harder to read the database.

Contact details have been represented with the datatype CHAR(254) as the maximum characters admitted for an email address are 254. This will also allow the user to introduce different telephone number lengths or special characters in case a tutor with a foreign telephone number is added.

2.Functional Dependencies

- Name: Functional dependency of contact details. Could be considered functional dependency of subject in case there is only one tutor with this name in the department, however different people can have the same name.
- Contact details: Primary Key due to their uniqueness (emails and telephone numbers aren't allowed to be repeated) and the fact you can access the rest of the information from them.
- Subject: functional dependency of contact details. Could be considered as a Candidate Key, but due to the circumstantial conditions of its functional dependencies considering it a Primary Key would be risky.
- Location: functional dependency of contact details. Could be considered Candidate Key due to its uniqueness, but it lacks functional dependencies.

3. Normalization

1NF

Tutors table

Tutor Name	Tutor Contact
Jackie	01970666543
Ahmed	07723456788
Jeff	jeff@stemtutors.net
Dafydd	07845333444
Peter	pete@languagetutors.com
Brian	brian@arttutors.org

Student table

Name	Contact Details
Elin Davies	elin@example.com
Steve Smith	steve@example.com

Tutor Location table

Contact Details (FK)	Location
01970666543	Aberystwyth
07723456788	Borth
07723456788	Aberystwyth
07723456788	Bow Street
07723456788	Llandre
jeff@stemtutors.net	Machylleth
jeff@stemtutors.net	Talybont
jeff@stemtutors.net	Bow Street
07845333444	Machylleth
pete@languagetutors.com	Bow Street
brian@arttutors.org	Aberystwyth
brian@arttutors.org	Llarhystud

Student Location table

Contact Details(FK)	Location
elin@example.com	Machylleth
steve@example.com	Llarhystud

Subject wanted table

Contact Details(FK)	Subject Wanted
elin@example.com	Maths
elin@example.com	Science
steve@example.com	English

Subject taught table

Contact Tutor (FK)	Subject
01970666543	Math
01970666543	Science
07723456788	Cymraeg
07723456788	Art
jeff@stemtutors.net	Math
jeff@stemtutors.net	Science
07845333444	Cymraeg
pete@languagetutors.com	English
brian@arttutors.org	Art

Commentary on the 1NF conversion:

We can use both the contact details of the student and the tutors as the Foreign Keys of the newly created Subject, Subject Wanted and Location tables. In the case of the Location table we can how Contact Details merge into one single attribute. Thanks to this, we can now access most of the repeated data through the Contact Details PKs in the Student and Tutor tables.

2NF

In the previous conversion we have already achieved the Second Normal Form, as all non-prime attributes depend on the CK, in this case the PK.

3NF

Tutors table

Tutor Name	Tutor Contact
Jackie	01970666543
Ahmed	07723456788
Jeff	jeff@stemtutors.net
Dafydd	07845333444
Peter	pete@languagetutors.com
Brian	brian@arttutors.org

Students table

Name	Contact Details
Elin Davies	elin@example.com
Steve Smith	steve@example.com

Student-Location Table

Contact Details (FK)	Location Name(FK)
elin@example.com	Machylleth
steve@example.com	Llarhyshtud

Student-Subject Table

Contact Details (FK)	Subject Name(FK)
elin@example.com	Maths
elin@example.com	Sciences

Tutor-Location Table

Contact Details (FK)	Location Name(FK)
01970666543	Aberystwyth
07723456788	Borth
07723456788	Aberystwyth
07723456788	Bow Street
07723456788	Llandre
jeff@stemtutors.net	Machylleth
jeff@stemtutors.net	Talybont
jeff@stemtutors.net	Bow Street
07845333444	Machylleth
pete@languagetutors.com	Bow Street
brian@arttutors.org	Aberystwyth
brian@arttutors.org	Llaryshtud

Tutor-Subject Table

Contact Tutor (FK)	Subject Name(FK)
01970666543	Math
01970666543	Science
07723456788	Cymraeg
07723456788	Art
jeff@stemtutors.net	Math
jeff@stemtutors.net	Science
07845333444	Cymraeg
pete@languagetutors.com	English
brian@arttutors.org	Art

Subject

Subject Name
Math
Science
Cymraeg
Art
English

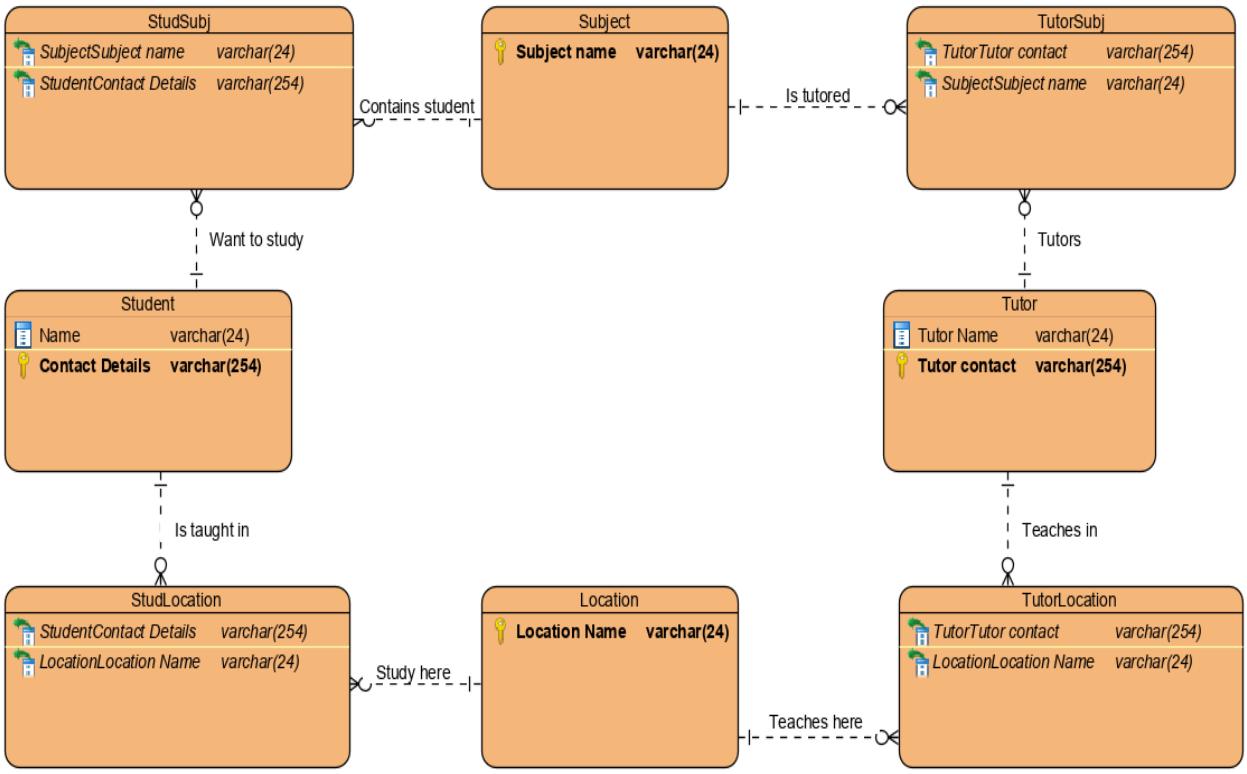
Location

Location Name
Aberystwyth
Bow Street
Llandre
Talybont
Machylleth
Borth

Commentary on the 3NF conversion:

As specified in the theory, the 3NF reduces data repetition in tables by dividing them. The result has been several intermediate tables that can be later used to connect the main 4 tables: Location, Subject, Tutors and Students.

4. ER Modelling



5. Implementation in PostgreSQL

```
                    Table "mag95.students"
    Column      |       Type        | Collation | Nullable | Default
----------------+-------------------+-----------+----------+---------
 name           | character varying |           |          |
 contact_details | character varying |           | not null |
Indexes:
    "students_pkey" PRIMARY KEY, btree (contact_details)
```

Students table

```
                    Table "mag95.tutors"
   Column     |       Type        | Collation | Nullable | Default
--------------+-------------------+-----------+----------+---------
 tutor_name    | character varying |           |          |
 tutor_contact | character varying |           | not null |
Indexes:
    "tutors_pkey" PRIMARY KEY, btree (tutor_contact)
```

Tutors table

```
                    Table "mag95.location"
   Column     |       Type        | Collation | Nullable | Default
--------------+-------------------+-----------+----------+---------
 location_name | character varying |           | not null |
Indexes:
    "location_pkey" PRIMARY KEY, btree (location_name)
```

Location table

```
                    Table "mag95.subject"
   Column    |       Type        | Collation | Nullable | Default
-------------+-------------------+-----------+----------+---------
 subject_name | character varying |           | not null |
Indexes:
    "subject_pkey" PRIMARY KEY, btree (subject_name)
```

Subject table

```
cs27020_19_20_mag95=> \d studentsubject
                  Table "mag95.studentsubject"
   Column    |       Type        | Collation | Nullable | Default
-------------+-------------------+-----------+----------+---------
 studcontact | character varying |           | not null |
 subname     | character varying |           | not null |


cs27020_19_20_mag95=> \d tutorsubject
                  Table "mag95.tutorsubject"
   Column   |       Type        | Collation | Nullable | Default
------------+-------------------+-----------+----------+---------
 tutcontact | character varying |           | not null |
 subname    | character varying |           | not null |


cs27020_19_20_mag95=> \d tutorlocation
                  Table "mag95.tutorlocation"
   Column   |       Type        | Collation | Nullable | Default
------------+-------------------+-----------+----------+---------
 tutcontact | character varying |           | not null |
 locname    | character varying |           | not null |


cs27020_19_20_mag95=> \d studentlocation
                 Table "mag95.studentlocation"
   Column    |       Type        | Collation | Nullable | Default
-------------+-------------------+-----------+----------+---------
 studcontact | character varying |           | not null |
 locname     | character varying |           | not null |
```

Intermediate tables

Explanation:

The four main tables have been implemented with their previously defined Primary Key. Intermediate tables have been defined without primary keys (being their columns the Foreign Keys of the main tables). These columns have been defined as not-nullabe, in order to avoid incomplete relations.

Some column names have been redefined in order to make them more distinguishable for the posterior querying.

6. Queries

i. All teachers who teach Math classes.

INPUT:

```
SELECT tutor_name
FROM tutorsubject, tutors
WHERE subname='MATH'
and tutorcontact=tutcontact;
```

OUTPUT:

```
tutor_name
----------
Jackie
Jeff
```

Explanation:

The query selects the tutor name of all the tutors teaching maths. We will use the tutorsubject table to find the tutor teaching Math and the tutors table to retrieve his name after having linked both tables.

ii. All appropriate tutors for Art in Aberystwyth

INPUT:

```
SELECT tutor_name
FROM tutorlocation, tutorsubject, tutors
WHERE subname='Art' and
        locname='Aberystwyth' and
        tutor_contact=tutorlocation.tutcontact and
        tutor_contact=tutorsubject.tutcontact;
```

OUTPUT:

```
tutor_name
----------
Ahmed
 Brian
```

Explanation:

We retrieve the tutor names by linking the tutorlocation, tutorsubject and tutors table by the tutor_contact attribute. After this, we ask for those teaching in Aberystwyth the Arts module.

iii. All students without an appropriate tutor.

INPUT:

```
SELECT name
FROM students
WHERE name NOT IN
(SELECT name
FROM students, studentlocation, tutorlocation, tutorsubject,studentsubject,tutors
WHERE
studentlocation.locname=tutorlocation.locname and
studentsubject.subname=tutorsubject.subname and
contact_details=studentlocation.studcontact and
contact_details=studentsubject.studcontact and
tutor_contact= tutorlocation.tutcontact and
tutor_contact= tutorsubject.tutcontact);
```

OUTPUT:

```
   name
-------------
 Steve Smith
```

Explanation:

We create an internal query that looks for all the students that can be assigned to a specific tutor that teaches in his location his wanted subjects. The main query retrieves the name of those students that did not appear in the internal query, or in other words, those that can't access a tutor with their specifications.

7. Self Evaluation

1. 20/20
   Followed the guidance and listed both attributes and types. Specific information has been explained too.
2. 8/10
   Indicated the functional dependencies and primary key, along with an explanation for its election. Could not infer candidate keys, due to the previous explanations.
3. 10/15
   Created example tables from the original UNF table. Though it seems correct, 3NF conversion might have extra usability transformations that should not be considered at this point.
4. 7/10
   ER model where I show the different tables along with their attributes, their use and relations. The model is not semantically faithful to the posterior implementation and some relations description are not as satisfactory as I would like them to be. However, the overall concept of the model is equivalent to its posterior implementation.
5. 20/30
   Implemented the database and works correctly. Specified primary keys in the main 4 tables, though the intermediate ones lack a proper definition of Foreign Key, that I tried to resolve by giving them the same functionality to their attributes.
   Tried to find a way to automatically populate these intermediate tables but couldn't achieve a functional tool, so data population is more laboured than wanted.
6. 10/10
   Queries work as expected. Further explanations have been added.