a)

A relational database organises data into tables which can be linked or related based on data common to each. A database should enable three important concepts, structured data storage, simplified data searching and analysis and finally simultaneous data sharing and accessibility. Relational databases are constructed by connecting multiple tables together, often joined on primary and foreign keys, theses connections are known as ‘relations’ hence the name relational database. Many different organisations and individuals use relational databases for a variety of different tasks, this can be on a large scale such as storing data for online stores such as eBay and Amazon or at a lower level such as keeping records of your own movie collection. Relational databases have certain criteria that must be met to ensure that they run correctly, here are a couple of the most important ones, each single row must contain only one single record also known as a tuple, furthermore the columns of a relational database must contain data from the same field, this is known as an attribute of the data. Relational databases must run as efficiently as possible, this is particularly evident on a large scale where thousands of records are kept, adhering to certain normal forms help with such efficiency by decomposing tables to reduce redundancy.

When designing a relational database, it is important to follow a framework that addresses certain rules, these rules must be followed to allow the database to operate as intended. Part of such framework is establishing use cases; a use case is a description of how a user interacts with a system and/or software to achieve a goal, the user is often referred to as the ‘actor’. A use case is broken down into key components, description of the system, definition of the actor, preconditions, the flow (steps the actor takes) and the post conditions. Use case frequency is important to acknowledge due to the efficiency, as highlighted before efficiency is essential for the database to operate correctly. Efficiency is measured by how often an action is expected to occur and how many concurrent users are carrying it out.

Another part of such framework is having certain rules to follow, organisations have rules, policies and legislation that govern how the actors interact with the data, these are known as business rules. Business rules don’t just apply to general businesses but anyone or any organisation who processes data. Relational databases can require details about exactly what data should be stored in them and can tell us whether any data fields can be left empty, these can be highlighted by the Business rules. Business rules can also be enforced at various levels depending on the requirements, some can be easy to enforce, and others can be extremely difficult.

In reference to the dataset provided in this assessment below there is an example of a use case for that data and the relevant business rules that apply to it.

**Use Case**: Booking an appointment

**Actor**: Patient

**Goal**: Patient successfully books an appointment at the doctor’s surgery in Newtown

**Actions**: Patient logs on to Newtown’s appointment portal

Patient provides their name, address, date of birth and phone number

Patient provides a description of their issue

Patient choses a registered doctor

Patient choses a date and time of the appointment

Portal provides confirmation

**Business Rules**

User must be registered at the doctor’s surgery, therefore have login details

User must provide valid name, address, date of birth and phone number

User must be registered with the doctor they wish to visit

User must choose an available appointment slot

b)

In relational databases entities are a single example of a member of an entity type, an entity type is essentially a category, if there are more than one entity types involved in a table then multiple tables should be used, it is important to recognise that these entities can have relationships with other entities of the same or other types. These relationships can take different forms, they can be binary meaning that they are between two different entities, and they can also be unary between two records in the same entity. A vital aspect of a relationship is its cardinality which describes how many records are involved in a particular relationship in a schema. One to one, one to many and many to many. Participation is also used in conjunction with cardinality, mandatory participation is where every record in the table must take part in the relationship, and optional participation is where entities are not required to participate in the relationship. Multiplicity combines cardinality and participation and sets 0 for optional participation and 1 for mandatory participation when constructing ER diagrams.

Entity relationship diagrams known as ER diagrams, design, construct and inspect the entities and relationships in a relational database. ER diagrams are useful for a non-technical audience to understand and inspect the construction of a database. ER diagrams can be built in different ways the two common notational styles are the Chen style and the unified modelling language (UML) style, for this assessment I will be constructing a UML style ER diagram.

The UML style is made up of a series of boxes used to represent the different tables, these boxes are given table names at the top of each box used to represent the entity types, the relationship names are written above the lines between the tables to represent their relationships with each other, the primary key is underlined, and the cardinality is shown at the end of each relationship line. The ER diagram below shows the different tables needed and the relationships between them, this ER diagram does not yet account for any decomposition of tables.

Diagram

Description automatically generated

The doctors and patients have a many to many relationships as a patient can have multiple doctors and a doctor can have multiple patients, this means that a joining table must be created to account for the primary keys of each table, the appointment table however acts as a natural joining table as the patient and doctor can only communicate through an appointment therefore, they both have a one-to-many relationship with appointments.

c)

In relational database design the database should store a give fact only once, this is desirable to achieve efficiency and consistency within the database, avoiding duplication therefore is an important goal, this means that decomposition of tables is needed to avoid such duplication. Normal forms are used in database design to achieve the criteria specified above, there are several different types of normal forms, but normal form one to three are the most prevalent and will be focused on in this assessment. A table satisfies first normal form if is arranged in a tabular format, this means that it is ordered correctly, and no table cells contain a null value. There must also be no duplicate rows or no table cells containing more than one value. Having no null values is often ignored in database design, this is a heavily debated topic as in some cases having null values are important for example if there is an empty space to be later filled when the data arrives. In the database provided in this assessment there are several null values relating to known allergies, current medication, and recent condition.

There are several issues in database provided. Many of these issues conform under first normal form, the first issue is in terms of the text, SQL must not have spaces in the naming columns, we must either use under scores or camel case to correct this issue and ensure that we are consistent throughout the database using the same form of writing. Secondly the date and time for the appointments are not written in the correct SQL format, the correct format is YYYY/MM/DD therefore must be changed in the database design. There are also duplicate entries with the same individual Frank Jones who is clearly the same person due to his address and phone number, therefore one entrance must be deleted, that one being the Frank with a birth date in 2008 as that is not consistent with the other age ranges. In terms of first normal form the main issues are that multiple table cells contain more than one value. There are also issues with there being several null values, these null values however are justified as those fields do not have to be populated if a patient does not have any information relating to them, it also gives opportunity to be filled once the patient has completed their appointment. When constructing this database there will also be some small name changes for ease of reading, the names will be broken down into first and last as that ensures that they are atomic values.

We will first convert this table into first normal form and decompose any necessary tables. We need to first add additional fields however to the table, this is in the form of primary keys, primary keys must be unique, therefore no other individual can have the same one, in the table the patients phone number may be suggested as a primary key, however this phone number may be a landline and not be unique to one individual. Furthermore as healthcare data is highly sensitive and must be secured due to patient confidentiality an additional column will be added in the form of patient id, a hybrid key could be used, however in healthcare none of the information on the key should relate to the patient due to confidentiality, an artificial key must therefore be used, the best approach for this is a random set of nine digits as this gives a upper limit of nearly one billion users, this is better than using an auto incremented system as every patient having the same number of digits in their id helps with consistency. We must also give the doctors an id, the only information we have of them is their name, for this assessment we will assume that there are no duplicate names.

Second normal form must have the criteria of first normal form to begin with, every attribute that is not part of the primary key must be fully functionally dependant on the primary key, ensuring that a table has full functional dependency removes the duplication of facts that can be defined by only part of the primary key. In the database provided in this assessment we will assume that there is a dependency between current medication and recent conditions, this means that we will need to create a composite primary key to conform to second normal form. We must also decompose the known allergies to a separate table which will remove the duplication of the primary key, it is important to separate these tables due to update, insert and delete anomalies, for if there are two of the same primary key and we need to update values in reference to that primary key how will SQL know which one to update.

Diagram

Description automatically generatedIn third normal form we must first have our tables in second normal form. every attribute that is not part of the primary key must be fully functionally dependant of the primary key alone. We could argue that our tables are already written in third normal form as all the records are dependent on the primary key alone. There could be an argument for current medication and recent conditions being functionally dependant, however there could be hundreds of medication options for the recent condition therefore it can’t depend on it.

The tables above are all constructed to the third normal form standard, every non key field is dependant on the key the whole key and nothing but the key, this also applies to the composite primary key as the non key field is fully functionally dependant on both the primary keys, achieving third normal form was an important goal as it represents that may database design is working in an efficient manner, with a larger set of variables to consider it may be difficult to achieve third normal form as the decomposition of many tables adds to the complexity of coding and may introduce some consistency errors. These normal forms will be more easily recognised once the tables have been constructed and there is clear indication of normal form criteria being met.

d)

Text

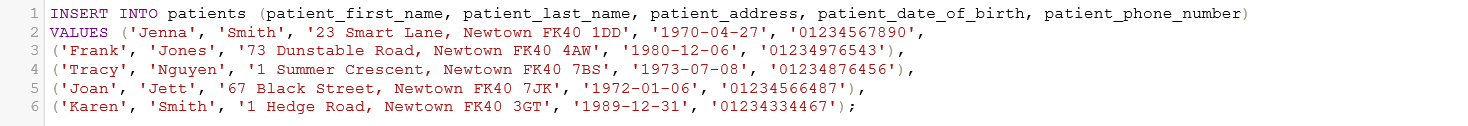
Description automatically generatedText

Description automatically generatedMySQL is one of the layers used to run a database management system, I will be using MySQL and PHP myadmin that is a web page interface that uses MySQL and is common to building websites in this question. We use MySQL for a variety of tasks, it passes information to databases, allows you to add alter and delete data and allows you to query it. Below shows the construction of the database as per the database design specifications.



The patient’s table was later altered as missing columns were added, as shown above.

The primary keys are almost all ID’s this ensures that the primary key will always be a unique value, I also ensured that certain fields can not be left null, this was important to add as missing key data for a patient’s identification could cause identification errors, this is especially important due to the fact medical data must be confidential.

e)

Above shows one example of the insertion of data into the patients table, I chose to display the patients table as it is the largest one and stores the most information. The primary key did not need to be inserted into the table as it is auto incremented and therefore automatically assigned to the data, each row shows a different indicidual and there relevant patient data. The other tables in this database had data inserted into them following the same structure above, some were very small where only foreign keys needed to be matched.

f) This section will feature a series of screenshots containing the relevant queries for each question.

i)

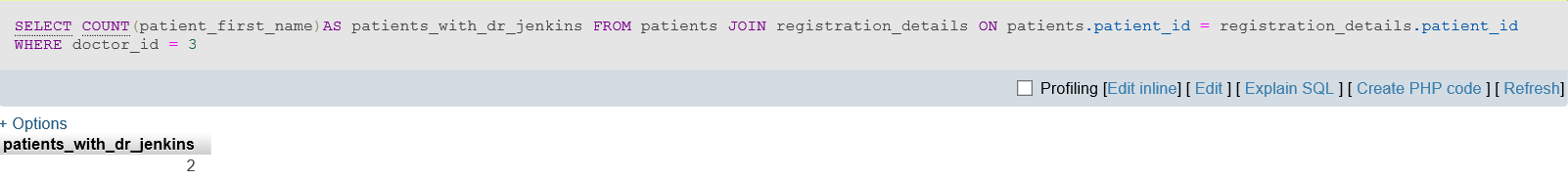
Table

Description automatically generated

ii)

Graphical user interface

Description automatically generated with medium confidence

iii)

Graphical user interface, text, application

Description automatically generatediv)

Graphical user interface, application

Description automatically generatedv)

Graphical user interface, application

Description automatically generatedvi)

Graphical user interface, application, Word

Description automatically generatedvii)

Graphical user interface, application

Description automatically generatedviii)

ix)

Graphical user interface, application

Description automatically generated