**QUESTION 1**

Describe some realistic constraints, including both business-related ones and nonbusiness-related ones such as primary keys, foreign keys, check constraints, and not null constrains, for the tables and attributes, etc..

* **PRIMARY KEY CONSTRAINTS**

This constraints one or a set of columns to mandatory and unique values. Meaning that no primary key column can be null. For an example, a table for Students, we must be able to store and identify all data associated with each student. Primary key enables us to clearly identify each student’s modules, course, tuition fees, marks, etc. They are also an integral part of the relationships that encourage referential integrity, that is the parent-child relationship. If you insert or update a column value forcing a duplicate, the update or insert statement will not be executed, it will fail with an error because primary key has to be unique. Alter table command is used to add, drop, disable or enable primary key constraint.

* **FOREIGN KEY CONSTRAINTS**

This refers to a set of columns or column that is used to establish a link between two tables to control the data. A link is created when the column that has the primary key for one table is referenced by a column in another table. Primary key from the first table becomes the foreign key in the second table. For an example, the modules table is linked to the student table through StudentID column which is the primary key in the student table and foreign key in the modules table. By creating this relationship, it means that the value of StudentID cannot be inserted into the modules table if it does not exist in the student table. Foreign key constraint guarantees that changes cannot be made to data in primary key table if they invalidate the link to data in the foreign key table. If there is need to delete or alter a row, you need to first either delete or change the foreign key data in foreign key table which connects the foreign key to different primary key data. Values that make up the foreign key are set to null when the corresponding row in parent table is altered or deleted.

* **UNIQUE CONSTRAINT**

This constraint is used to make sure that the values entered in specific column that is not primary key are not duplicate. Although primary key and unique constraint both ensures uniqueness, unique constraint is used when you need to enforce uniqueness in a specific column or set of columns that is/ are not primary key. It allows null values but only one null value is allowed per column. It can be referenced by foreign key. When you add a unique constraint to an already existing column, database engine automatically examines the existing data to ensure that all values are unique. If there are duplicate values, the database engine creates returns an error. The database engine creates unique index to enforce uniqueness of this constraint. If you attempt to insert duplicate, it will return an error message.

* **CHECK CONSTRAINT**

It enforces domain integrity by limiting values that are accepted by column(s). Check constraints can be created with any logical expression that returns a True or False value depending on the logical operators. For an example, the range of tuition fees column can be limited by creating a check constraint that only allow the data to range from R10 000 to R50 000. This helps to ensure that students are not charged fees that is beyond the regular range. Single column can have multiple check constraints and also a single check constraint can be applied to many column is it was created at the table level. These constraints determine the validity of values from logical expressions. This type of constraints rejects values that evaluate to false. Null values evaluate to unknown so their presence may override the constraint. For the check constraints to work, the table must have at least one row.

* **NOT NULL CONSTRAINTS**

This constraint prevents unknown values from being added into the column. Built-in data type support the presence of null values but some of the business rules may require that data must be provided. For example, every student must have next of kin contact details in case of emergency. This constraint is used to make sure that data in a specific column is never assigned the unknown value.

BUSINESS RULES

Each course belongs to a department, and a department can have multiple courses.

Each course is taught by an instructor, and an instructor can teach multiple courses.

Each instructor belongs to a department, and a department can have multiple instructors.

Each section belongs to a course and is taught by an instructor.

A student can only be enrolled in a course, while a course can have multiple students.

ERD

A picture containing text, diagram, plan, parallel

Description automatically generated