# Information Management II

## Database Design Project

## John Sinclair - 16325734

# Golfing Union of Ireland

# Overview

I chose to design a database for the Golfing Union of Ireland (GUI). The GUI is the governing body for male amateur golf in Ireland and Northern Ireland. It represents 400 golf clubs and over 160,000 members. The entity tables in this database are as follows: Province, Golf Club, Employee, Competition, Player and Round.

The **Province** table represents each of the sections of the Golfing Union of Ireland. The Union is split into five sections, the National level and then the four provinces of Ireland. The table contains info such as the province’s unique ID and name, as well as its contact info, Phone Number and Email.

The **Golf Club** table is a list of all the golf clubs in the Golfing Union of Ireland. This table contains the club’s unique ID, name, address and the province’s ID that it is a member of.

The **Employee** table is a list of all the employees employed directly or indirectly by the Golfing Union of Ireland. It contains all relevant employee information such as their unique employee ID, their first and last names, their role, salary, date of birth (DOB) and the ID of the club or province that they work for.

The **Competition** table is a list of all the past and upcoming competitions in the Union. Each element in the table represents a single competition with information like the competitions unique ID, the name of the competition, the date it takes place and the ID of the golf club that is hosting it.

The **Player** tables is a list of all the golfers that are indirectly (via golf clubs) members of the Golfing Union of Ireland. It has information columns for their unique player ID, their first and last names, their date of birth (DOB), their official world golf ranking (OWGR), their handicap and the ID of the golf club to which they are a member.

The **Round** table represents all the rounds played within the Union, logged for reasons such as handicap calculation. The table contains information such as the rounds unique ID, the date the round was played, the gross and net score, the ID of the golfer who played the round and the ID of the golf course where the round was played.

The last table in the database is an intermediary table called **Competition Entry**. This table is a list of all the competition entries in the Union. It contains a timestamp of when the entry was made as well as the players ID and the ID of the competition that the player is entering.

# Entity Diagram

|  |
| --- |
| A picture containing graphical user interface  Description automatically generated |

# Entity Relationship Diagram

|  |
| --- |
| Diagram  Description automatically generated |

# Relational Schema

|  |
| --- |
| Diagram  Description automatically generated |

# Functional Dependency Diagram

|  |
| --- |
| Diagram  Description automatically generated |

# Constraints

I implemented a number of constraints when designing this database. Firstly, each entities table in this database was initialised with a not null, unique, automatically generated primary key. The primary keys are created as not null to insure every entry in the table can be uniquely identified, thus avoiding integrity constraint violations, creating an entity integrity constraint.

Another integrity constraint I implement was that all foreign keys within the entity tables was created as not null, this ensures referential integrity.

During the table creation process I implemented constraints and checks ensuring that valid inputs were made to a tables attributes. This was done so to ensure data integrity was intact within the database.

The first data integrity constraint I implemented was making sure an employee worked at either club or at provincial level. In this database an employee is either employed directly by the province or by a specific golf club. I did this with the following code:

|  |
| --- |
| Graphical user interface, text, application  Description automatically generated |

The second constraint I implemented was that no golf club could be a member of the National level province. Instead they must be a part of one of the other four provinces (Leinster, Munster, Connacht and Ulster).

|  |
| --- |
|  |

The last constraint implemented in the design of this database ensured that no player on the register had a handicap greater than the limit of 54.

|  |
| --- |
| A picture containing graphical user interface  Description automatically generated |

# Security

The constraints I implemented ensure the databases integrity is not compromised by any accidental corruption of the data but it is no defence against intentional corruption of the databases data. This is where the security of the database is essential.

With this database access control is its primary form of security. This prevents any user from accessing the database without the proper authorisation. This ensures that an unauthorised user cannot view data they should not be able to view nor will they be able to edit data unless they have the authority to do so.

Access to this database is organised by accounts which have differing levels of access. There are administrators who have the right to create new accounts and assign these new accounts with a certain level of database access. Access can also be split into two categories, read and write privileges and then another two levels of access to the database. Read and write privileges are important as some users of the database can have a read only view of certain tables and views without the ability to change or modify the data within. The higher level of privilege is having the ability to write changes to the database, again this can be split into two main groups, the first can modify the data contained within certain tables whereas the higher access users can modify the tables and views themselves.

The first of the two levels of database access is that the administrator can grant certain access privileges separately to the relations in the database and the second level is that the administrator can grant certain access privileges to a user in regards to exact tables or view the user can have viewing or modification access to.

# Views

Within the database I created five views.

The first view was a simple club directory. Its purpose was to be easier to navigate and understand when searching for a particular golf club. It is organised by province first then club name making it easier for the user.

A picture containing graphical user interface

Description automatically generated

Club Directory (club\_directory)

Graphical user interface, text, application

Description automatically generated

The next view I created was a more detailed view of the competition entries made. This was because the Competition Entry table has functional data but it is difficult to understand information. With this view you can clearly see when the entry was made, the players information, such as their ID and name, the name of the competition they entered (versus the competition ID) and the golf clubs name that is hosting the competition.

|  |
| --- |
| Timeline  Description automatically generated |
| Competition Entries (detailed\_comp\_entries)Table  Description automatically generated |

Another view in the database is a detailed view of the competitions past and upcoming in the entire Union. The Competition table only shows the name and date of the competition along with the clubs ID that is hosting it. With this more detailed view the user gets information such as the province, golf club, competition name, date and the phone number for the club in order to make a reservation. This view is ordered by province, club and then by date.

|  |
| --- |
| A picture containing text, person, screenshot  Description automatically generated  Competitions (detailed\_competitions) |

The next view the database required was a ranked players table. The Player table had the players in order of addition to the database so this view displays all the players in the Union sorted into provinces and clubs, then ranked in ascending order by their handicap.

Graphical user interface, text, application

Description automatically generated

Players (detailed\_players)

Table

Description automatically generated

The final view I created within the database was a list of all the rounds played, sorted into provinces and golf clubs then by date played from newest to oldest. I thought this view was required because the standard Round table only contained the basic information such as the gross score, net score, date and the IDs of the player and golf club where the round was played. This more detailed rounds view provided a better way for a player to see their rounds. The view is sorted into provinces and clubs and then by the player’s ID, this keeps their scores all together. The view is then further sorted by date so the newest score would be shown first.

Graphical user interface, text, application

Description automatically generated

Rounds (detailed\_rounds)

Table

Description automatically generated

# SQL Commands

### Creating the tables

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

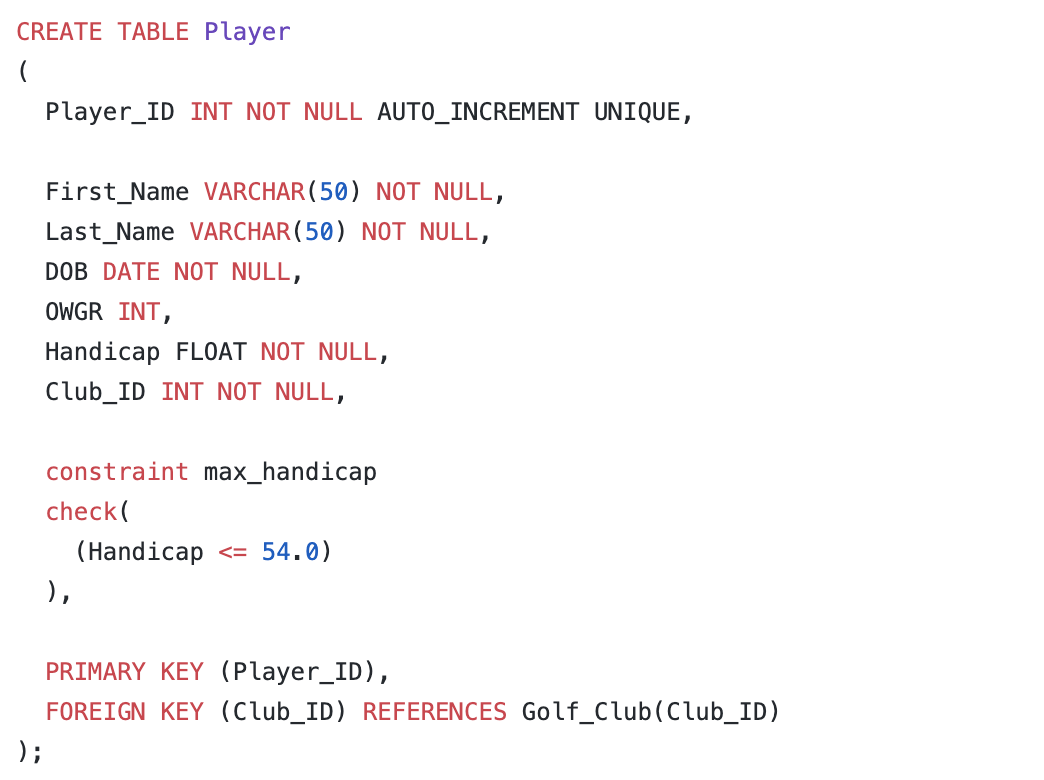
Description automatically generated

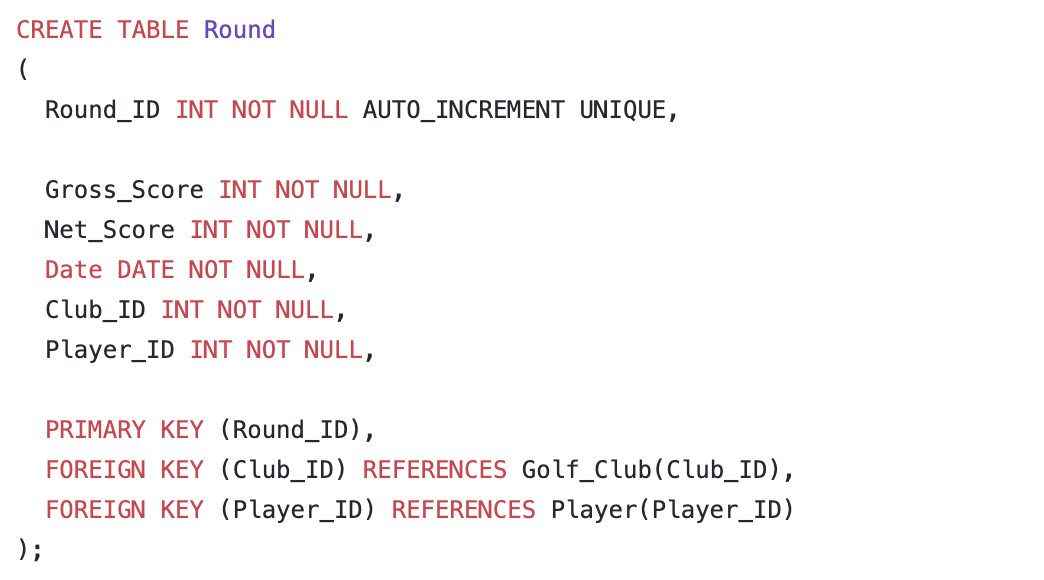
Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated





Graphical user interface, text, application

Description automatically generated

### Populating the databases

A picture containing text, screenshot, person

Description automatically generatedText

Description automatically generated

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated