The following report deals with our Database Systems Project where we must scrape data, create a database, and create an interface to interact with that database.

Database Project

Media Viewing Model

Jocelyn Atwell 201800014

Keamogetse Mooki 201901580

Joshua Martin 201902378

# Table of Contents

[Table of Contents 1](#_Toc56279063)

[Table of Figures 1](#_Toc56279064)

[Introduction 2](#_Toc56279065)

[Design and Methodology 2](#_Toc56279066)

[Web scrapping 2](#_Toc56279067)

[Data Testing 3](#_Toc56279068)

[Data Base Application 3](#_Toc56279069)

[Results and Discussion 4](#_Toc56279070)

[Conclusion and Recommendations 5](#_Toc56279071)

[References 5](#_Toc56279072)

[Appendices 5](#_Toc56279073)

# Table of Figures

[Figure 1: Movies Query 4](#_Toc56279235)

[Figure 2: Series Query 4](#_Toc56279236)

[Figure 3: Top Rated Media 4](#_Toc56279237)

[Figure 4:Lowest Rated Media 4](#_Toc56279238)

[Figure 5: Adding the server-based data set 3](#_Toc56279239)

[Figure 6: Created the Data base 4](#_Toc56279240)

[Figure 7: The User Interface Showing all functions 4](#_Toc56279241)

# Introduction

We live in a world with unlimited access to Big Data for almost anything and everything. In this project we will be focusing on data from movie and tv series websites from the entertainment industry. We chose this specific data, because the entertainment industry grows exponentially each year especially during this pandemic. There are numerous active websites housing data for these movies and tv series.

# Design and Methodology

## Web scrapping

Before we scraped our data, we added the text “robot.txt” to verify if we had permission to scrape the site, which was granted.

The initial step we took to gather our data was to scrape data from a movie and tv series website. We used web scraping in Python to do so. By definition, web scraping is a process in which data is extracted from websites in its unstructured raw form, cleaned and placed into a structured format such as a table (Wikipedia, 2020).

The website we scraped from was the popular IMDB website. The data we scraped from comes from the webpage Top Action Rated Movies and TV Shows. The reason for this is that the website only allows you to view the media through a specific genre in which each main genre has its own URL. We chose this website as we initially wanted to scrape from many different genres to create a large database. After we scraped the action genre, we obtained 500 entries. However, after we cleaned the data, we were left with 50 entries only. When we tried scraping more data from other genres, but our code gave errors (Anon., 2020). Due to this we have decided to only use the data we scraped from the action genre.

We realised that the year was a string object, and we could not convert it into a numerical value due to the series having a dash between the years as a series broadcasts over a duration. We then used this dash in the years to create and append a “Type” column classifying the media into a movie or a tv series. We also received problems when extracting our cleaned table into an excel file. The dash between the years turned into two weird letters. However, we just used the find and replace function to convert it back into a dash. The code for the web scrapping process will be in the appendices below.

## Data Testing

To test and clean the data we analysed the data and normalised the table which then after we used Microsoft Access to test the data before creating the database application. To do this we just imported our excel file containing processed data. Thereafter, we used access to create test data queries.

The first query Movies query displays all the movies only from the main table Action\_T. There were 48 entries in this query alone. Compared to the number of entries in our finalised dataset, we can see that movies are more popular in the action entertainment industry than series are. This was the SQL code to obtain the table below as well.

**SQL code:**  
SELECT Action\_T.ID, Action\_T.Media, Action\_T.Year, Action\_T.IMDB, Action\_T.Votes, Action\_T.Restriction, Action\_T.Genre1, Action\_T.Genre2, Action\_T.Genre3

FROM Action\_T

WHERE (((Action\_T.Type)="Movie"));

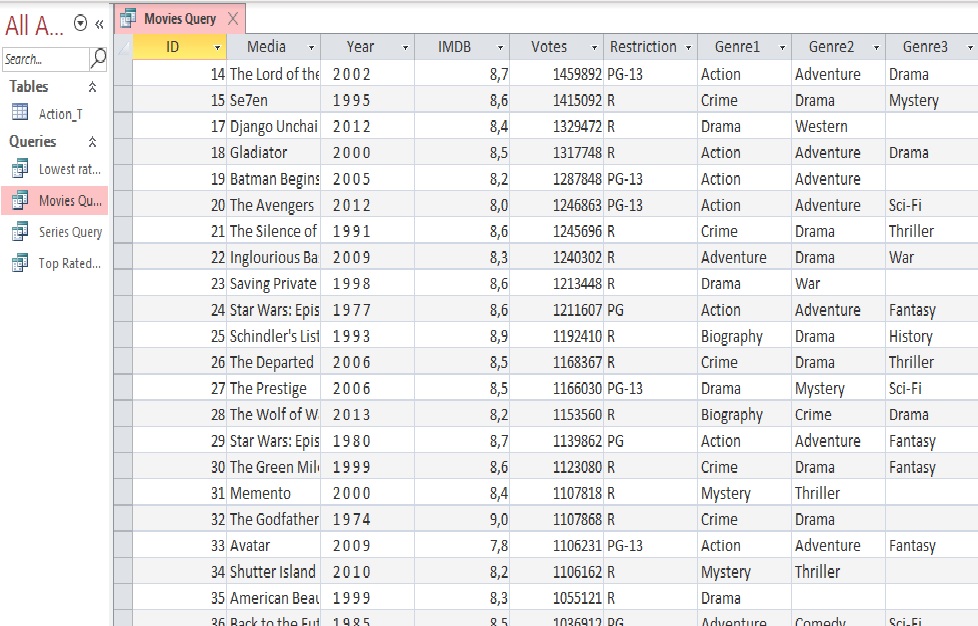


Figure : Movies Query

The following query Series Query displays all the series excluding the movies. There are only two entries in this query. We can see that from this query, series are not popular in the entertainment industry based on the action genre. The SQL code for obtaining the table is below.

**SQL code:**

SELECT Action\_T.ID, Action\_T.Media, Action\_T.Year, Action\_T.IMDB, Action\_T.Votes, Action\_T.Restriction, Action\_T.Genre1, Action\_T.Genre2, Action\_T.Genre3

FROM Action\_T

WHERE (((Action\_T.Type)="Series"));

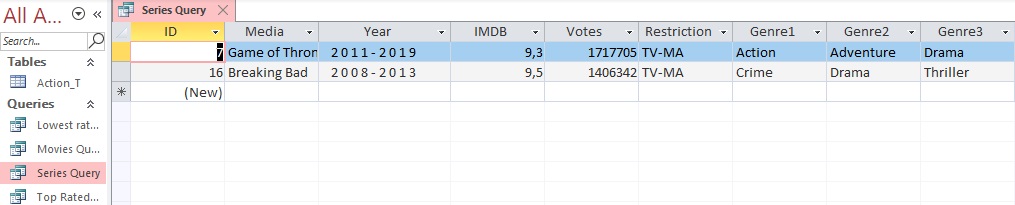


Figure : Series Query

This query Top Rated Media displays all the movies and tv series with a rating between 9 and 10. There are only six entries in this query where four are movies and two are tv series. The SQL code for this query is displayed below.

**SQL code:**

SELECT Action\_T.ID, Action\_T.Media, Action\_T.Year, Action\_T.IMDB, Action\_T.Restriction, Action\_T.Genre1, Action\_T.Genre2, Action\_T.Genre3, Action\_T.Type

FROM Action\_T

WHERE (((Action\_T.IMDB) Between 9 And 10));

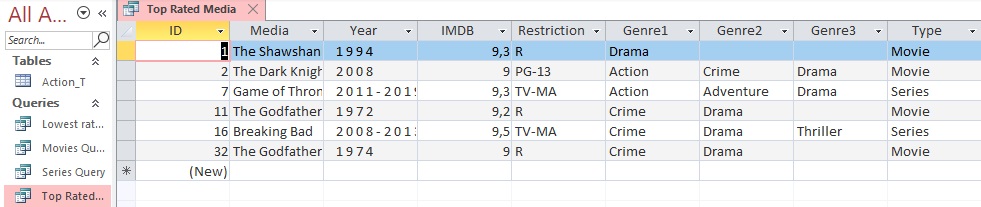


Figure : Top Rated Media

The final query we created is called Lowest Rated Media. This table displays all the media with a rating between 1 and 6.9. As you will see below, there are no entries in this table. This means that majority of the media data has a rating between 7 and 8.9 and that there are not movies or tv series with a low rating. The SQL code for the table is displayed below.

**SQL code:**

SELECT Action\_T.ID, Action\_T.Media, Action\_T.Year, Action\_T.IMDB, Action\_T.Restriction, Action\_T.Genre1, Action\_T.Genre2, Action\_T.Genre3, Action\_T.Type

FROM Action\_T

GROUP BY Action\_T.ID, Action\_T.Media, Action\_T.Year, Action\_T.IMDB, Action\_T.Restriction, Action\_T.Genre1, Action\_T.Genre2, Action\_T.Genre3, Action\_T.Type

HAVING (((Action\_T.IMDB) Between 1 And 6));

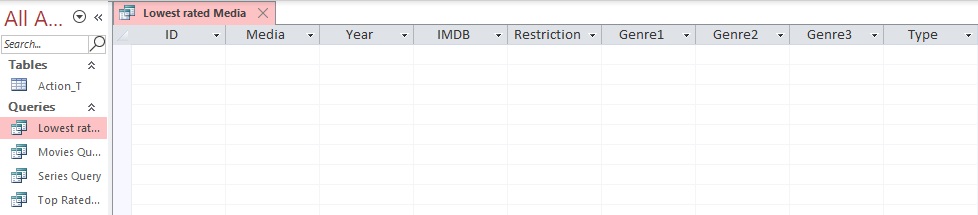


Figure :Lowest Rated Media

## Data Base Application

Creating a database application required that we set up a database which the application would access the data from. Using visual c# we created a server-based dataset and created the data source with two tables, movies, and series, and populated the data. Next was designing the user interface, the application allows the user to interact with the database via the several functions. These functions include viewing the data within the database, inserting data, and deleting data within the database. Each function in the backend works with several programming classes, such as the windows. sql. client package, and algorithms which incorporate various sql queries such as selects, unions, delete and inserts which connect to the database to produce the required results.

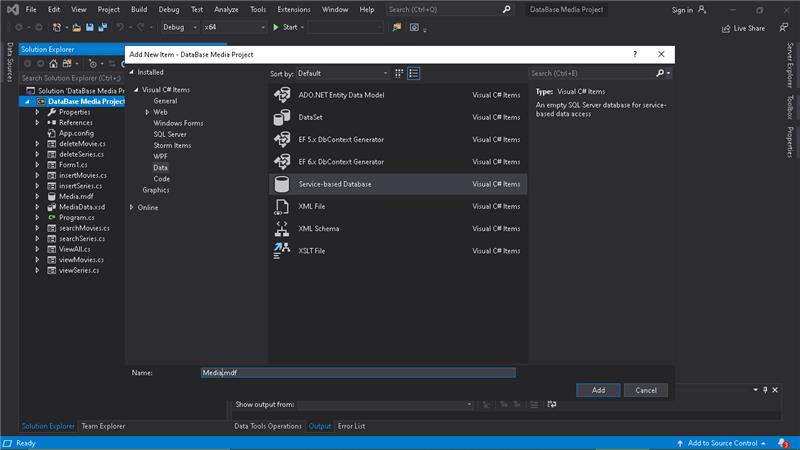
[](#_Table_of_Figures)

Figure 5: Adding the server-based data set.

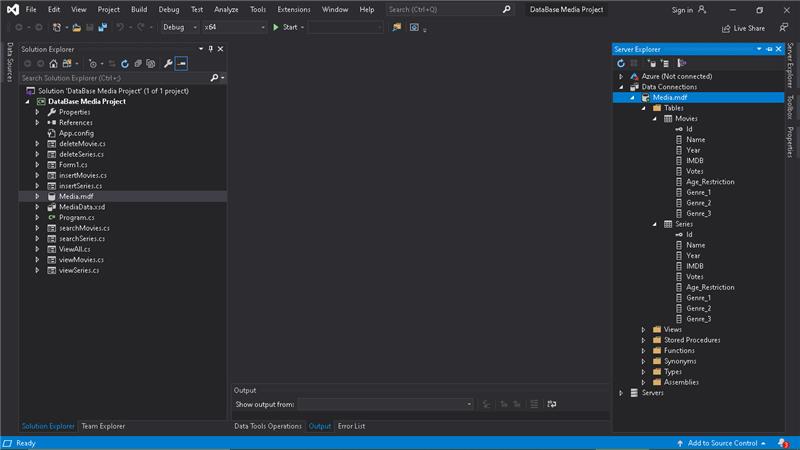
[](#_Table_of_Figures)

Figure 6: Created the Data base

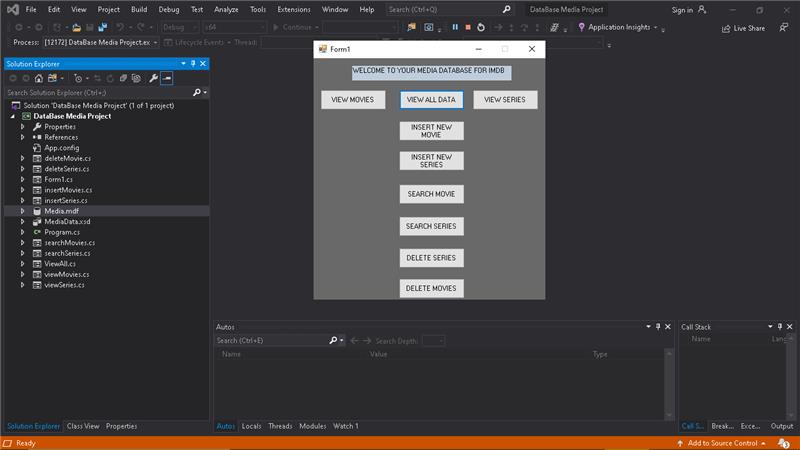
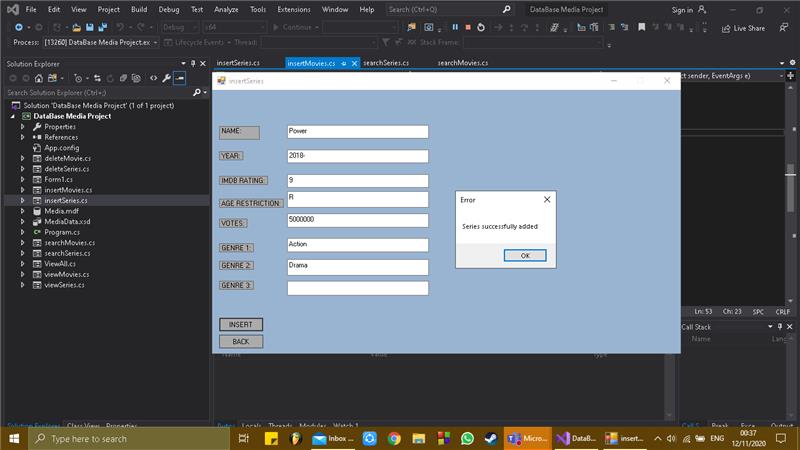
[](#_Table_of_Figures)

Figure 7: The User Interface Showing all functions.

# Results and Discussion

Our model works with 100 percent accuracy, allowing the user to fully interact with the data using the user interface via the functions within the application.



# Conclusion and Recommendations

The application could be improved by scraping more data to enlarge the data source, Improving the user interface, and provide the user with much more functions to interact with the data.

This data application in future if developed further could be used by IMDB as a desktop application as right now they only have a mobile application.

# References

Anon., 2020. *IMDB.* [Online]   
Available at: https://www.imdb.com/  
[Accessed 10 October 2020].

Anon., 2020. *Top 50 Action Movies and TV Shows.* [Online]   
Available at: https://www.imdb.com/search/title/?genres=action&explore=title\_type,genres&pf\_rd\_m=A2FGELUUNOQJNL&pf\_rd\_p=e0da8c98-35e8-4ebd-8e86-e7d39c92730c&pf\_rd\_r=8KERVMZHP819ZFVGEEA2&pf\_rd\_s=center-2&pf\_rd\_t=15051&pf\_rd\_i=genre&ref\_=ft\_gnr\_pr2\_i\_2  
[Accessed 11 October 2020].

Anon., 2020. *Web Scraping returns empty values.* [Online]   
Available at: https://stackoverflow.com/questions/48193590/web-scraping-returns-empty-values  
[Accessed 7 November 2020].

Wikipedia, 2020. *Wikipedia, the free encyclopedia.* [Online]   
Available at: https://en.wikipedia.org/wiki/Web\_scraping  
[Accessed 11 November 2020].

# Appendices

1. [Action Python Code.ipynb](file:///C:\Users\Student\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Files\S0\402\Attachments\Action%20Python%20Code.ipynb)
2. [Action\_Table.xlsx](file:///C:\Users\Student\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Files\S0\402\Attachments\Action_Table.xlsx)
3. [Media.accdb](file:///C:\Users\Student\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Files\S0\402\Attachments\Media.accdb)
4. [Data base Application](file:///C:\Users\Student\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Files\S0\503\Attachments\DataBase%20Media%20Project\DataBase%20Media%20Project.sln)