# **Checkpoint 1 (due date 6/21)**

### **Title of the project:** Video Game Sales Database

### **Group members:** Tae Yong Namkoong, Nate Sackett, Hyukjoon Yang

# Motivation and Project Description

### (I) The application domain:

Our database will be useful for analyzing and predicting sales trends in the game market. It will also provide meaningful information about demographic (regional) data in the gaming industry.

### (II) The type of application (web-based, standalone, or app):

We require a web-based application because the data should be accessible by multiple parties. The video game market is rapidly growing and expanding, so the application should be updatable and viewable in real-time. Since new games can be released on any day, and their sales data can change rapidly, our database needs to support concurrent access/modification of data as well as scalability to large datasets.

### (III) Why it is essential to have a database in this domain application:

Millions of games are sold every year; to track the success of a particular game, platform, genre, or company, you must look at it in the context of its competition.

### (IV) Description of the functionality of the application:

Users will be able to add new games (and their associated information) to the database as well as update sales data for the various regions. The information for each game will be editable and deletable, including removing/changing a publisher, genre, or the name. The database will handle user queries.

# Dataset

Our dataset contains the names of video games, the platform on which they are released, the year of publication, their genre, publisher, and the sales data for North American, European, Japanese, Other, and Global regions.

We chose this dataset because our group is knowledgeable about the topic and it met the minimum requirements for the project, having 11 columns of data and 16600 rows. The dataset is well-organized, requiring little, if any, pre-processing, and comes from a single source, so no multiple source integration will be required.

*Dataset taken from:* <https://www.kaggle.com/gregorut/videogamesales>

# Entity Sets, Relationship Sets, & ER Diagram

### Entity sets (Key underlined):

1. Game (GameName: String, Year: Date, Genre: String, Rank: Integer)
   * GameName - Unique String; < 255 Characters
   * Year - Date; 1900 < Year < 2200
   * Genre - String; choose from:
     + Action, Adventure, Fighting, Misc, Platform, Puzzle, Racing, Role-Playing, Shooter, Simulation, Sports, Strategy
   * Rank - Unique positive integer; 1 < Rank < 20000
2. Platform (PlatformName: String)
   * PlatformName - Unique String; < 255 Characters
3. Sales (Region: String, UnitsSold (in millions): Float)
   * Region - Unique String; < 255 Characters
   * UnitsSold - Positive float; 0.00 < UnitsSold < 1000.00; no digits smaller than the hundredths place
4. Publisher (PublisherName: String)
   * PublisherName - Unique String; < 255 Characters

### Relationship Sets:

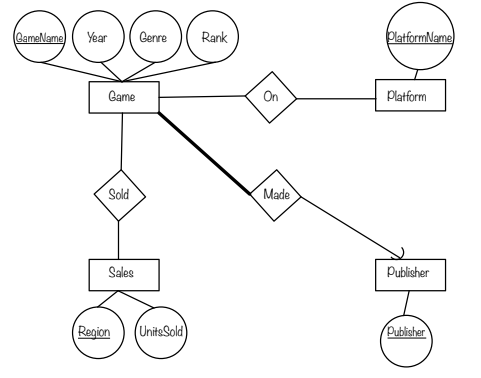
1. On (GameName: String, PlatformName: String)
   * Game ㅡ♦ㅡ Platform (many to many)
2. Sold (GameName: String, Region: String)
   * Game ㅡ♦ㅡ Sales (many to many)
3. Made (GameName: String, PublisherName: String)
   * Game **ㅡ**♦ㅡ) Publisher (many to many, total participation, referential)

### ER Description:

Our ER diagram is designed to describe a video game from our dataset. Video game is its own entity, which is described by the characteristics of year, its year of release, its genre, and rank, its current position in the bestselling of all time rankings. Three additional entities describe, and are related to the game entity: the platform it was released On (PlayStation 4, Nintendo Gameboy, etc.), the publisher that Made the game (Microsoft, Sega, etc.), and how the game Sold, or its Sales in a particular region and the number of units sold.

* A Platform is the system the Game was released on. It’s key is the name of the Platform. Platform and Game are related by the relationship “On”. A Game may be released on many Platforms and a Platform may have many Games.
* Sales is the data about how many millions of units have been sold of a Game. The key for Sales is the Region to which the sales data is referring, and it has the characteristic UnitsSold, the number of millions of units sold in the specified Region. A Game is related to Sales by the relationship “Sold”. A Game may have Sales in many Regions, and the Sales in those Regions may consist of many Games.
* A Publisher is the studio that creates and/or releases Games. Publisher is given the key PublisherName. Publisher is related to Game by the relationship “Made”. Every Game must have a single Publisher, and every Publisher can make multiple Games.

### ER Diagram:



# **Technical Details**

We will use the Java language to implement the DBMS interface. We chose Java because every member of the group is comfortable with the language and it is required for Checkpoint 3. The DBMS that will be used will be MySQL. We chose MySQL because we wanted a relational, web-based database, that can be viewed and edited simultaneously.

# **Project Management**

|  |  |  |
| --- | --- | --- |
| Week | Work | In charge |
| **2** | **Checkpoint 1**   1. Motivation & Project Description 2. Dataset Selection & Explanation 3. ER Diagram | Hyukjoon  Nate  Tae |
| **3** | **Checkpoint 2 (06/28)**   1. Revise ER Diagram 2. Relational Schema 3. Non-trivial Functional Dependencies 4. Update Changes to DB | Hyukjoon  Nate  Tae  All |
| **4** | 1. Revise Relational Schema 2. Normalization Process 3. Data Standardization 4. Plan Implementation | Hyukjoon  Nate  Tae  Hyukjoon |
| **5** | **Checkpoint 3 (07/12)**   1. Refine Relational Schema 2. Normalization Process 3. Data Standardization 4. Implement Database | Hyukjoon  Nate  Tae  All |
| **6** | **Checkpoint 4 (07/19)**   1. SQL Queries 2. Interface 3. Stored Procedures 4. Complete Implementation 5. Evaluation 6. Write Final Report 7. Prepare Final Presentation 8. Complete Presentation Video | All  All  All  All  All  All  All  All |
| **7** | **Presentations** | All |