# ../../../../../Desktop/Screen%20Shot%202017-11-14%20at%202.Programming assignment:

# A video game with a database-backed leaderboard

I’m sure you are familiar with the classic video game Space Invaders. I have developed my own twist on this game: It’s like the original, but the player chooses a hero to fight the invaders. The heroes are the classic video game archetypes: strong fighter, rogue archer, and intelligent wizard. It took me about 10 hours to write the game in Java and about 3 more hours to get the HUD (the buttons and labels) to work. You can download the game from here:  
<http://github.com/jvanderhyde/space-invaders/archive/master.zip>  
It is a NetBeans project. Open it up and run it. When you lose, you can close the game window.

Your goal in this assignment is to create a database to store the scores of each game played. You will also write a Java program to display various leaderboards and lists of high scores that are derived from the data stored in the database. For this task (Task 1), you will only work on figuring out the SQL queries. The Java code will be in later tasks.

We will set up the database with only one table, where each row is a record of a game. Here is an example:

**INVADERS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **player** | **hero** | **gametime** | **score** |
| 1 | crono | 1 | 2017-11-14 12:17:01 | 50 |
| 2 | crono | 2 | 2017-11-14 12:17:01 | 470 |
| 3 | marle | 3 | 2017-11-14 12:17:01 | 320 |
| 4 | lucca | 2 | 2017-11-14 12:17:01 | 5630 |
| 5 | crono | 1 | 2017-11-14 12:17:01 | 480 |

## Task 1: Design the queries

Task 1 is the planning stage. Design the SQL SELECT queries that you will need for each of the problems below. The only table involved in each query will be INVADERS, because it is the only table in the database.

1. Note that Crono has played the game 3 times. What is his high score? Write a SELECT query that will find Crono’s high score.
2. We will need a high score list. This is a list of the highest-scoring games, with the score, the name of the player, and the hero used for each game. Write a SELECT query that creates the high score list, in order from greatest to least score. Example result:

**HIGH SCORE LIST**

|  |  |  |
| --- | --- | --- |
| **score** | **player** | **hero** |
| 5630 | lucca | 2 |
| 480 | crono | 1 |
| 470 | crono | 2 |
| 320 | marle | 3 |
| 50 | crono | 1 |

1. There are three heroes, numbered 1, 2, and 3. Their names are Acton, Ellis, and Currer, respectively. We will need a leaderboard for each hero (a total of 3). A leaderboard is a list of players, ranked by score. Each hero leaderboard lists the top 10 players who have played with that hero, their high score with that hero, their average score with that hero (rounded to a whole number), and the number of times they have played with that hero. The whole table should be in order by high score. Write the 3 SELECT queries you will need to build these 3 leaderboards from the existing data. Example result:

**ELLIS LEADERBOARD**

|  |  |  |  |
| --- | --- | --- | --- |
| **player** | **high score** | **average score** | **times played** |
| lucca | 5630 | 5630 | 1 |
| crono | 470 | 470 | 1 |

1. Finally, we need a leaderboard for the whole game. It is a list of the top 10 players, in order by highest score achieved in a game, with the hero that game was played with and the date that game was played. Create a query to generate this list for you, or use multiple queries and explain how you will combine the results in Java code. Example result:

**GAME LEADERBOARD**

|  |  |  |  |
| --- | --- | --- | --- |
| **player** | **score** | **hero** | **date** |
| lucca | 5630 | Ellis | Nov. 14, 2017 |
| crono | 480 | Acton | Nov. 14, 2017 |
| marle | 320 | Currer | Nov. 14, 2017 |

Put all your queries and other notes into a document and upload it to Canvas by the posted due date.