|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Period: Finals** | **Subject:** 6OOPLANG | | **Class Code:** \_\_\_\_\_\_ | | | **Score:** \_\_\_\_/50 |
| **Activity:** *Final Examination* | | **Date:** 12/03/2020 | | **Day/Time:** TH(10:15-1:15) | | |
| **Topic:** | | | **Instructor:** JR D. Casingal | | | |
|  | | |  | |  | |
| **Students Name(s):** Sumang,Michael Dave C. | | | | | | |

1. OBJECTIVES:

* To be able to collaborate and create an authentic solution to a specific problem using OOP.

1. INSTRUCTIONS:
   * Create a simple Java program that will solve the assigned problem from the group. Then, copy the Java source code and paste it. The grading will be based on the provided rubrics.

**Example:**

**Computer Problem:**

Create a simple Java program that will require a number of games, then, within each game is a number of gamers. Each gamer per game are required to store their corresponding game details consisting of username, game score and game status. Game status should depend on the game scores, as follows:

|  |  |
| --- | --- |
| **Game Scores** | **Game Status** |
| 1 – 50 | *Noob* |
| 51 – 100 | *Savage* |
| 101 – 150 | *Expert* |
| 151 – 200 | *Master* |
| 201 – 250 | *Veteran* |
| 251 – 300 | *Legendary* |
| 300 above | *Mythical* |

The program then will identify the game status based on the game score. Finally, display all of the details per gamers on every games.

MINIMUM PROGRAM REQUIREMENTS:

NOTE: Not following this requirements will nullify the point on the rubrics.

* 1 constructor
* 2 fields (private)
* 1 Enumeration
* it should have at least 4 methods with corresponding specifiers (except the main method)
* it should be in a 2 separate Java file (main and an invoking class)
* Follow the standard program structure (INPUT, PROCESS, OUTPUT)
* Observe an Object-oriented approach.
* SAMPLE OUTPUT:
* *Enter number of games: 2*
* *Enter the Game#1: C.O.D*
* *Enter the Game#2: PUBG*
* *Enter number of gamers each games: 3*
* Game#1: C.O.D
* Gamer1: Juan
* Score: 200
* Gamer2: Pedro
* Score: 50
* Game3: Juana
* Score: 350
* Game#2: PUBG
* Gamer1: Rowan
* Score: 100
* Gamer2: Allan
* Score: 250
* Game3: Ben
* Score: 300
* DISPLAY:
* **Game#1: C.O.D**

|  |  |  |
| --- | --- | --- |
| **Gamer** | **Score** | **Status** |
| 1 -  Juan | 200 | Master |
| 2 -  Pedro | 50 | Noob |
| 3 -  Juana | 350 | Mythical |

* **Game#2: PUBG**

|  |  |  |
| --- | --- | --- |
| **Gamer** | **Score** | **Status** |
| 1 -  Rowan | 100 | Savage |
| 2 -  Allan | 250 | Veteran |
| 3 -  Ben | 300 | Legendary |

**Filename: GameMain.java**

***Source code:***

*import java.util.\*;*

*public class GameMain*

*{*

*static Scanner input = new Scanner(System.in);*

*public static void main(String[] args)*

*{*

*//Input*

*int numGames, numPlayers, playerScore;*

*String gameName,playerName;*

*GameInvoked enumObj = new GameInvoked();*

*System.out.print("Enter number of games: ");*

*numGames = input.nextInt();*

*GameInvoked[] gameArr = new GameInvoked[numGames];*

*input.nextLine();*

*for (int i=0;i<numGames;i++)*

*{*

*int gCount = i+1;*

*System.out.print("Enter the Game#"+gCount+": ");*

*gameName = input.nextLine();*

*gameArr[i] = new GameInvoked(gameName);*

*}*

*System.out.print("\nEnter number of gamers each games:");*

*numPlayers = input.nextInt();*

*String[][] playerArr = new String[numGames][numPlayers];*

*int[][] scoreArr = new int[numGames][numPlayers];*

*for (int i=0; i<numGames;i++)*

*{*

*int gCount = i+1;*

*System.out.println("\nGame#"+gCount+":"+gameArr[i].getName());*

*for (int j=0;j<numPlayers;j++)*

*{*

*int pCount = j+1;*

*input.nextLine();*

*System.out.print("Gamer#"+pCount+": ");*

*playerName = input.nextLine();*

*playerArr[i][j] = playerName;*

*System.out.print("Score: ");*

*playerScore = input.nextInt();*

*scoreArr[i][j] = playerScore;*

*}*

*}*

*//Output*

*m\_Printer(enumObj, gameArr, playerArr, scoreArr, numGames, numPlayers);*

*}*

*public static void m\_Printer(GameInvoked enumObj,GameInvoked[] gameArr,String[][] playerArr,int[][] scoreArr, int x, int y)*

*{*

*System.out.println("\n----------------OUTPUT-------------");*

*for (int i = 0;i < x;i++ )*

*{*

*int gCount = i+1;*

*System.out.println("\nGame#"+gCount+":"+gameArr[i].getName());*

*for (int j = 0;j < y ;j++ )*

*{*

*int pCount = j+1;*

*System.out.println(pCount+" - "+playerArr[i][j]);*

*System.out.println(" Score: "+scoreArr[i][j]);*

*enumObj.setStatus(scoreArr[i][j]);*

*System.out.println("Status: "+enumObj.getStatus());*

*}*

*}*

*}*

*}*

**Filename: *GameInvoked.java***

***Source code:***

*public class GameInvoked*

*{*

*private int score;*

*private String name;*

*enum Status*

*{*

*DEFAULT, NOOB, SAVAGE, EXPERT, MASTER, VETERAN, LEGENDARY, MYTHICAL;*

*}*

*GameInvoked()*

*{*

*}*

*GameInvoked(String x)*

*{*

*name = x;*

*}*

*public String getName()*

*{*

*return name;*

*}*

*public void setStatus(int y)*

*{*

*score = y;*

*}*

*//Main Process*

*public Status getStatus()*

*{*

*Status z = Status.DEFAULT;*

*if (score>=1 && score<=50)*

*{*

*z = Status.NOOB;*

*}*

*else if (score>=51 && score<=100)*

*{*

*z = Status.SAVAGE;*

*}*

*else if (score>=101 && score<=150)*

*{*

*z = Status.EXPERT;*

*}*

*else if (score>=151 && score<=200)*

*{*

*z = Status.MASTER;*

*}*

*else if (score>=201 && score<=250)*

*{*

*z = Status.VETERAN;*

*}*

*else if (score>=251 && score<=300)*

*{*

*z = Status.LEGENDARY;*

*}*

*else if (score>=301)*

*{*

*z = Status.MYTHICAL;*

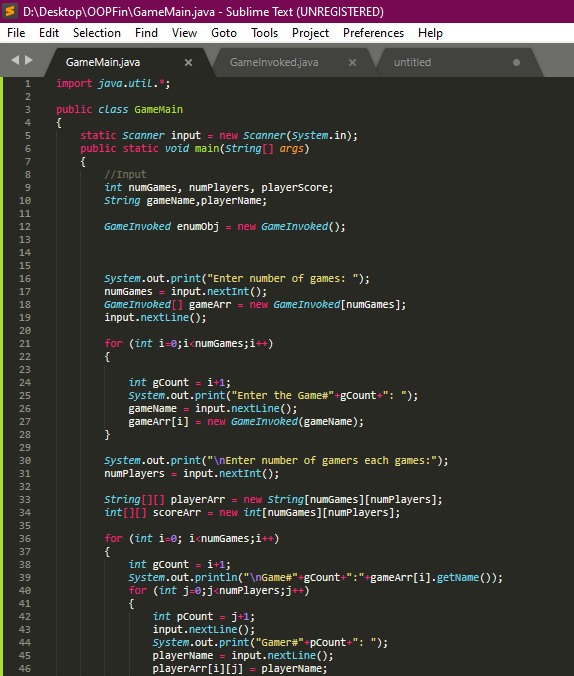
*}*

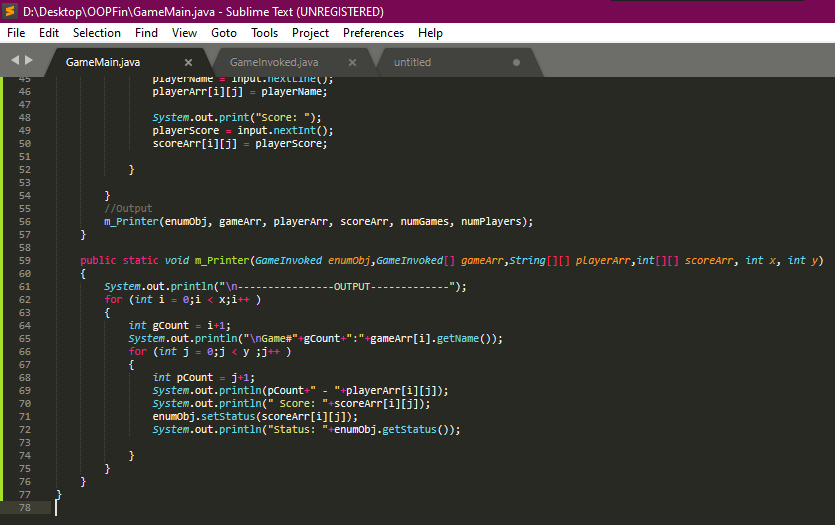
*return z;*

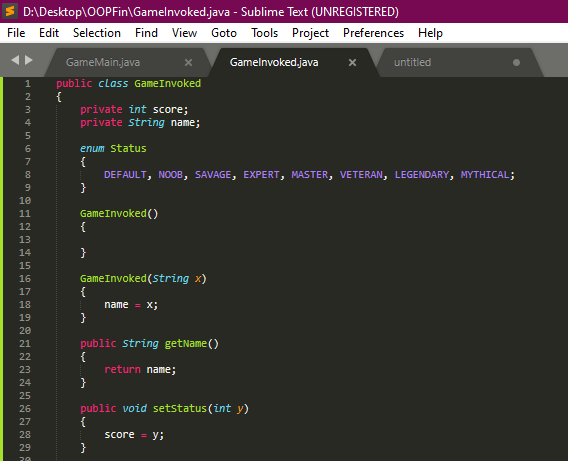
*}*

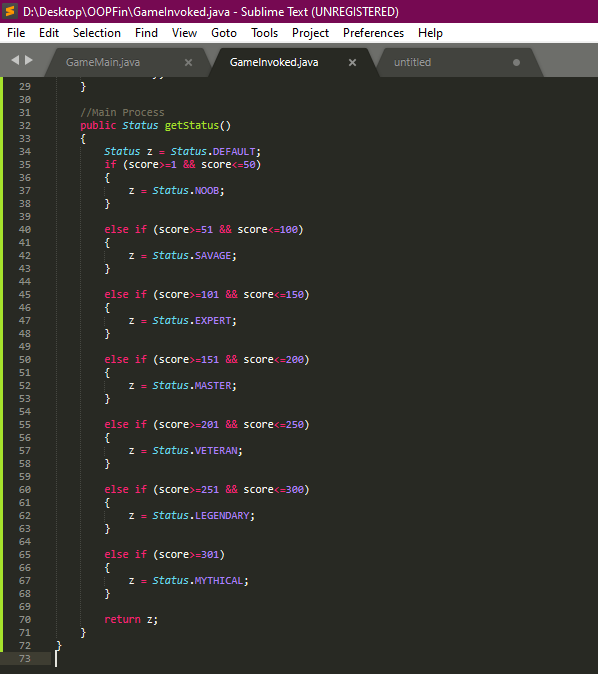
*}*

**Code snip:**

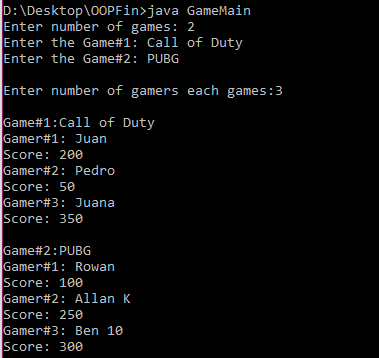
****

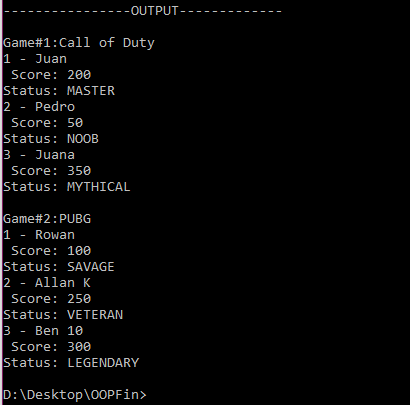
****

****

****

***OUTPUT SNIP:***

******

******