**King Saud University**

**College of Computer & Information Science**

**CSC111 – Project**

**All Topics Section:**

**31209**

**Instructions:**

**1- Due Date: Saturday \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**2- Only one or two students for the project.**

1. **Submit your Project by email only (\*\*\*\*\*\*\*\*\*\*).**
2. **I will grade only the first submission, if there is a group of two students submits only one copy.**
3. **Turn in two files (Java) only (Player.java, Team.java); do not send me zip files.**
4. **The grade is five Marks.**
5. **Make sure you put your id and name of all students inside both files.**

**Problem Description**

**You need to write a program for a system to manage the players in a team. Your system should be able to add players to the team, retrieve a player by a number or name, delete a Player given number and return the total number of players that have the same rank.**

**Do not change the names given in the UML, you must use the keyword “this” when it is needed, If you change the name, you will get zero for that part.**

|  |
| --- |
| **Player** |
| * **name : String** * **number : int** * **position : String** * **rank : String** * **goals : int** * **count : int** |
| **+ Player()**  **+ Player(name : String, position : String, rank : String , goals : int)**  **+ print() : void**  **+ setters for name, position, rank, goals**  **+ getters for name, position, rank, goals, number, count**  **+ addGoals(g : int) : void** |

**Number: is the number on the player’s shirt, must be 1 - 150, unique.**

**Count: is the number of players created.**

**Rank: is the player’s level according to age, must be one of the following: “Kids”, “Youth”, “First” (Ignore case).**

**Position: is the player position in the field, must be one of the following: “Keeper”, “defense”, “middle” and “attack” (ignore case).**

**Player (): calls the other constructor with the following: “NA”,**

**“middle”,”kids”,0**

**addGoals (int g): add number of goals in g to goals.**

|  |
| --- |
| **Team** |
| **+ name : String**  **+ city : String**  **+ arrPlayers : player[]**  **+ noPlayers : int** |
| **+ Team(name : String , c : String)**  **+ Team(name : String , c : String , size : int)**  **+ searchPlayer(num : int) : int**  **+ searchPlayer(n : String) : int**  **+ searchPlayer(num : int, rank : String) : int**  **+ addPlayer(p : Player) : boolean**  **+ deletePlayer(number : int) : boolean**  **+ goalsTotal() : int**  **+ goalsNonAttack() : int**  **+ getKids() : Player[]**  **+ numberOfRank(String r) : int**  **+ print() : void**  **+ printRank(r : String) : void**  **+ sort() : void** |

**Description of Team class:**

|  |
| --- |
| **Team (name: String, c: String): This constructor will create an array of players with size 150.** |
| **Team (name: String, c: String, size: int): This constructor will create an array of players, if the size is negative, make it 150.** |
| **searchPlayer (num : int): This method receives the number of the Player and return its location or -1 if not found.** |
| **searchPlayer (n : String) : This method receives the name of the Player and return its location or -1 if not found.** |
| **searchPlayer (num : int, rank : String): This method receives the number of the Player and it’s rank then returns its location or -1 if not found.** |
| **addPlayer (p : Player): This method adds a new player, the number must be unique, if the player added return true otherwise return false.** |
| **deletePlayer (number : int): This method receives the player number and delete this player, if deleted return true otherwise return false.** |
| **goalsTotal (): This method returns the total number of goals.** |
| **goalsNonAttack ():This method returns the total number of goals for player not attackers.** |
| **getKids (): This method return an array of players with rank is Kids** |
| **numberOfRank (r : String): This method returns number of players having rank equals to r.** |
| **print (): This method prints all information about all players.** |
| **printRank (r : String): This method prints all information about players having rank equals to r.** |
| **sort (): This method sorts the array in ascending order according to the players numbers.** |

**------------------------- FINSHED -------------------------**