

# MINI PROJECT - I



E – LEARNING PRESENTATION

**DEPARTMENT OF COMPUTER SCIENCE & APPLICATION**

**INSTITUTE OF ENGINEERING & TECHNOLOGY**



**GLA**  
**UNIVERSITY**  
**MATHURA**  
Established vide U.P. Act 21 of 2010.

**SUBMITTED TO:-**

MS. RUCHI TALWAR

(TECHNICAL TRAINER)

**SUBMITTED BY:-**

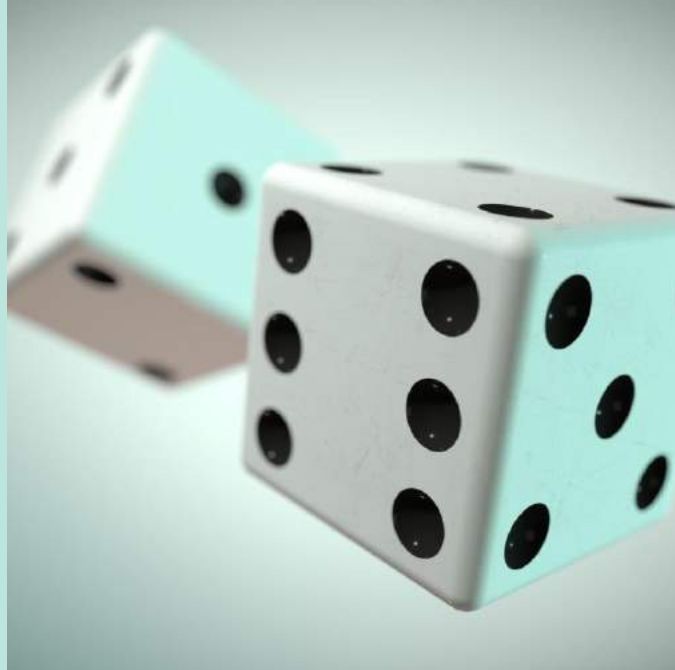
HARSHIT YADAV (201500287)

SACHIN GAHLOT (201500593)

JYOTI VASHISHTHA (201500319)

ASHUTOSH SINGH (201500170)

RITIKA MITTAL (201500578)



# INTRODUCTION

The aim of the Dice Throwing Game is to simulate a simple game for 2 players, where they take turn to each roll a dice twice, and score points according to the results of the dice rolls. The winner is the one who accumulates a pre-defined maximum score first. Your program will display a menu which allows the user of the program to select various options to simulate the various operations. Results of all the operations will be printed on the screen as plain text only (e.g. “Sachin rolled 5 + 3, and scored 8 points”).

# **SOFTWARE REQUIREMENTS**

- JDK16.0.2
- INTELLIJ 2021.1.3
- VISUAL STUDIO CODE
- ETHERNET ADAPTER
- WINDOW 11
- WEB TECHNOLOGIES: HTML, CSS, JAVASCRIPT

# **PROJECT DESCRIPTION**

The purpose of this project is that we will write a program that simulates a rather simplistic Dice Throwing Game. This section specifies the required functionality of the program. Only a text interface is required for this program; however, more marks will be gained for a program that is easy/intuitive to use, with clear information/error messages to the user.

The project is divided into 3 modules – Game, Player and Dice.



# GAME, PLAYER & DICE

The user selects from various options available. The user takes an option on a given menu. Each options have their multiple working. Users will be able to register themselves in the game as players. Only two players will be allowed in the Game. Players views their scores, which player is leading and which player wins the game. Player can also know more about the rules related to the game in the help section. Creating player name for the game, scores will reside in the draft area and it is saved by system. Show the results of the players while playing the rounds in the Game and also give a text response which player is in the leading position. Creating player name for the game, scores will reside in the draft area and it is saved by system. Show the results of the players while playing the rounds in the Game and also give a text response which player is in the leading position.



# WORKING

```
Game.java x
1 import java.util.Scanner;
2 import java.util.Random;
3 public class Game {
4     private Player p1;
5     private Player p2;
6     private Dice dice;
7     private int scoreToWin;
8     public class Player {
9         private final String name;
10        private int totalScore;
11        Player(String name){
12            this.name = name;
13        }
14        String getName(){
15            return name;
16        }
17        int getTotalScore(){
18            return totalScore;
19        }
20        void setTotalScore(int score){
21            totalScore += score;
22        }
23    }
```

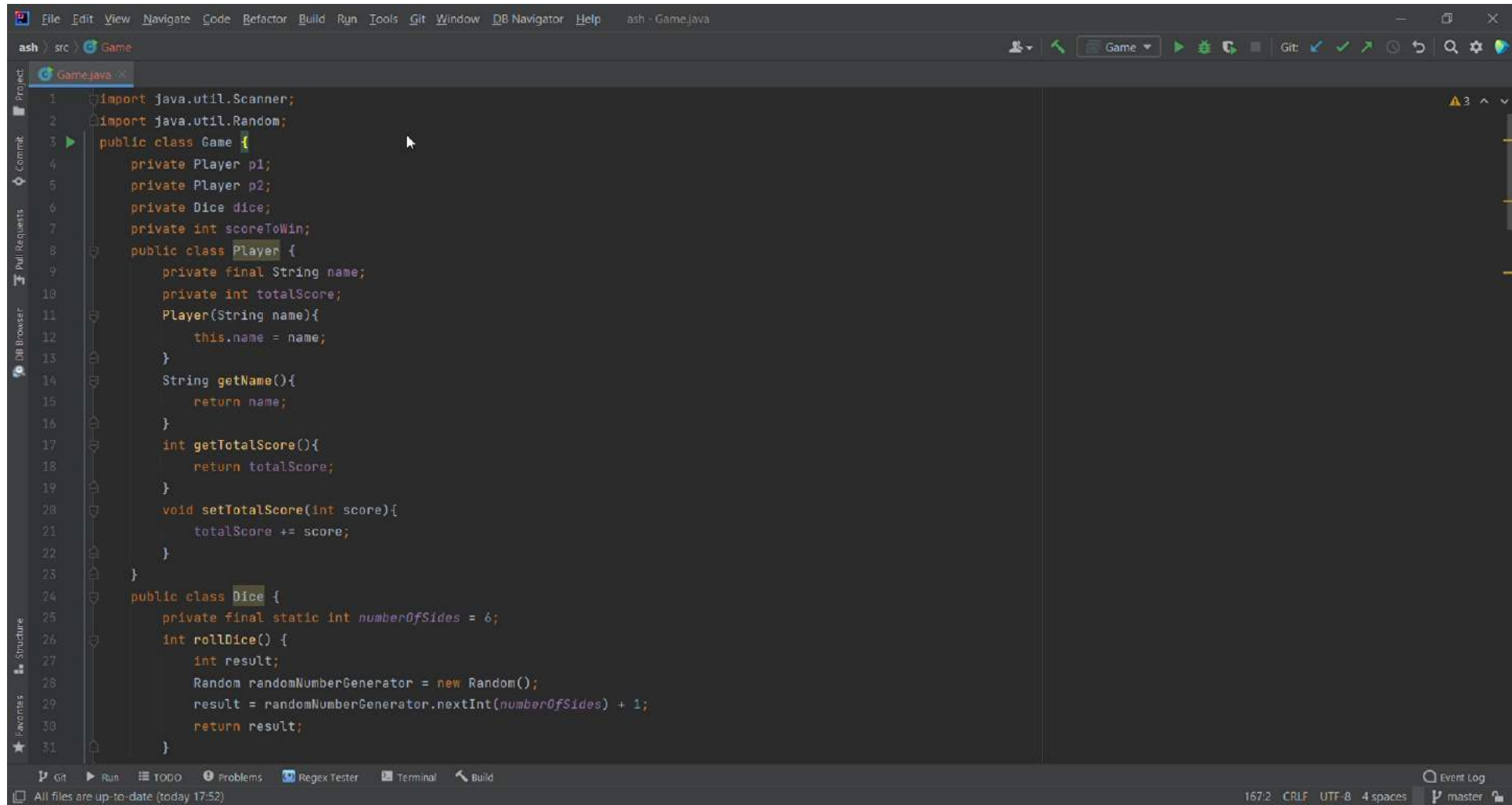
```
void startNewGame() {
    String p1Name;
    String p2Name;
    Scanner sc = new Scanner(System.in);
    System.out.print("Please enter player one name: ");
    p1Name = sc.nextLine();
    System.out.print("Please enter player two name: ");
    p2Name = sc.nextLine();
    System.out.print("Please enter the maximum score required to win: ");
    scoreToWin = sc.nextInt();
    p1 = new Player(p1Name);
    p2 = new Player(p2Name);
    dice = new Dice();
}
```

```
void playOneRound(Player p) {
    int result;
    int firstDiceRoll = dice.rollDice();
    int secondDiceRoll = dice.rollDice();
    if (firstDiceRoll == secondDiceRoll) {
        result = (firstDiceRoll + secondDiceRoll) * 2;
        p.setTotalScore(result);
        System.out.printf("%s rolled %d and %d, "
            + "and scored %d points(BONUS DOUBLE POINTS), "
            + "for a total of %d points",
            p.getName(), firstDiceRoll, secondDiceRoll,
            result, p.getTotalScore());
    } else {
        result = (firstDiceRoll + secondDiceRoll);
        p.setTotalScore(result);
        System.out.printf("%s rolled %d and %d, "
            + "and scored %d points, "
            + "for a total of %d points",
            p.getName(), firstDiceRoll, secondDiceRoll,
            result, p.getTotalScore());
    }
    System.out.println();
}
```

```
void whoIsLeading() {
    if (p1.getTotalScore() == p2.getTotalScore()) {
        System.out.format("Its currently a draw, "
            + "%s has %d, %s has %d",
            p1.getName(), p1.getTotalScore(),
            p2.getName(), p2.getTotalScore()
        );
    } else if (p1.getTotalScore() > p2.getTotalScore()) {
        System.out.printf("%s is leading, %s has %d points, "
            + "%s has %d points",
            p1.getName(), p1.getName(), p1.getTotalScore(),
            p2.getName(), p2.getTotalScore());
    } else if (p1.getTotalScore() < p2.getTotalScore()) {
        System.out.format("%s is leading, %s has %d points, "
            + "%s has %d points.",
            p2.getName(), p2.getName(), p2.getTotalScore(),
            p1.getName(), p1.getTotalScore()
        );
    }
}
```

```
boolean checkIfAnyoneHasWon() {
    if (p1.getTotalScore() >= scoreToWin && p2.getTotalScore() >= scoreToWin) {
        System.out.println("Its a draw! Both players have exceeded the score limit");
        return true;
    } else if (p1.getTotalScore() >= scoreToWin && p2.getTotalScore() < scoreToWin) {
        System.out.format("%s won", p1.getName());
        return true;
    } else if (p1.getTotalScore() < scoreToWin && p2.getTotalScore() >= scoreToWin) {
        System.out.format("%s won", p2.getName());
        return true;
    }
    return false;
}
```

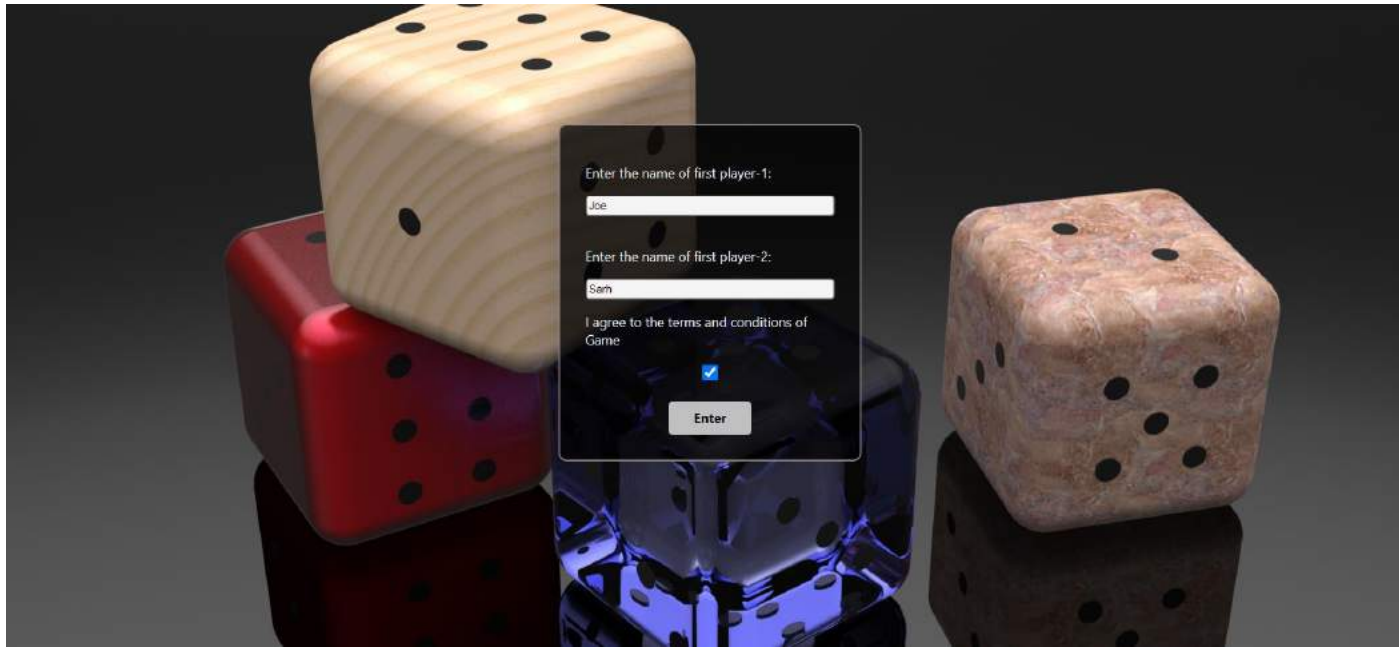




```
1 import java.util.Scanner;
2 import java.util.Random;
3 public class Game {
4     private Player p1;
5     private Player p2;
6     private Dice dice;
7     private int scoreToWin;
8     public class Player {
9         private final String name;
10        private int totalScore;
11        Player(String name){
12            this.name = name;
13        }
14        String getName(){
15            return name;
16        }
17        int getTotalScore(){
18            return totalScore;
19        }
20        void setTotalScore(int score){
21            totalScore += score;
22        }
23    }
24    public class Dice {
25        private final static int numberOfSides = 6;
26        int rollDice() {
27            int result;
28            Random randomNumberGenerator = new Random();
29            result = randomNumberGenerator.nextInt(numberOfSides) + 1;
30            return result;
31        }
32    }
```

The screenshot shows an IDE window titled 'ash - Game.java'. The code defines a 'Game' class with two inner classes: 'Player' and 'Dice'. The 'Player' class has attributes 'name' and 'totalScore', and methods 'getName()', 'getTotalScore()', and 'setTotalScore()'. The 'Dice' class has a static attribute 'numberOfSides' set to 6, and a 'rollDice()' method that uses a 'Random' object to generate a random number between 1 and 6. The IDE interface includes a menu bar, a toolbar, a sidebar with project and file views, and a status bar at the bottom showing file encoding and line count.

```
59
60 <div class="head">
61
62   <div class="form">
63     <form id='register' class='input-group-register'>
64       <p> Enter the name of first player-1:</p>
65       <input type='text' class='input-field' placeholder='Enter ur name' required id="p1">
66       <br>
67       <p> Enter the name of first player-2:</p>
68       <input type='text' class='input-field' placeholder='Enter ur name' required id="p2">
69
70       <p>I agree to the terms and conditions of Game</p>
71       <input type='checkbox' class='check-box'>
72       <br>
73       | <a href="./player.html" id="button" >Enter</a>
74     </form>
75   </div>
76 </div>
77 <script>
78
79
80 function js(){
81   var player1=document.getElementById(p1);
82   var player2=document.getElementById(p2);
83   alert("success");
84 }
85
86
87 </script>
88
89 </body>
```



```

player.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>Dice Game</title>
8    <link rel="stylesheet" href="./styles.css">
9    <script src="https://kit.fontawesome.com/81ef3afdd3.js" crossorigin="anonymous"></script>
10 </head>
11 <body>
12   <div class="sty">
13     <section class="section-1">
14       <h1>Refresh Me</h1>
15       <button onclick="myFunction()"><i class="fa-solid fa-arrow-rotate-right fa-2xl"></i></button>
16       <script src="./JS.js">
17         function myFunction();
18       </script>
19       <table>
20         <tr>
21           <th>
22             <div id="player-1">
23               <h2>Player-1</h2>
24               
25             </div>
26           </th>
27           <th>
28             <div id="player-2">
29               <h2>Player-2</h2>
30               
31             </div>
32           </th>
33         </tr>
34       </table>
35     </section>
36   </div>

```

```

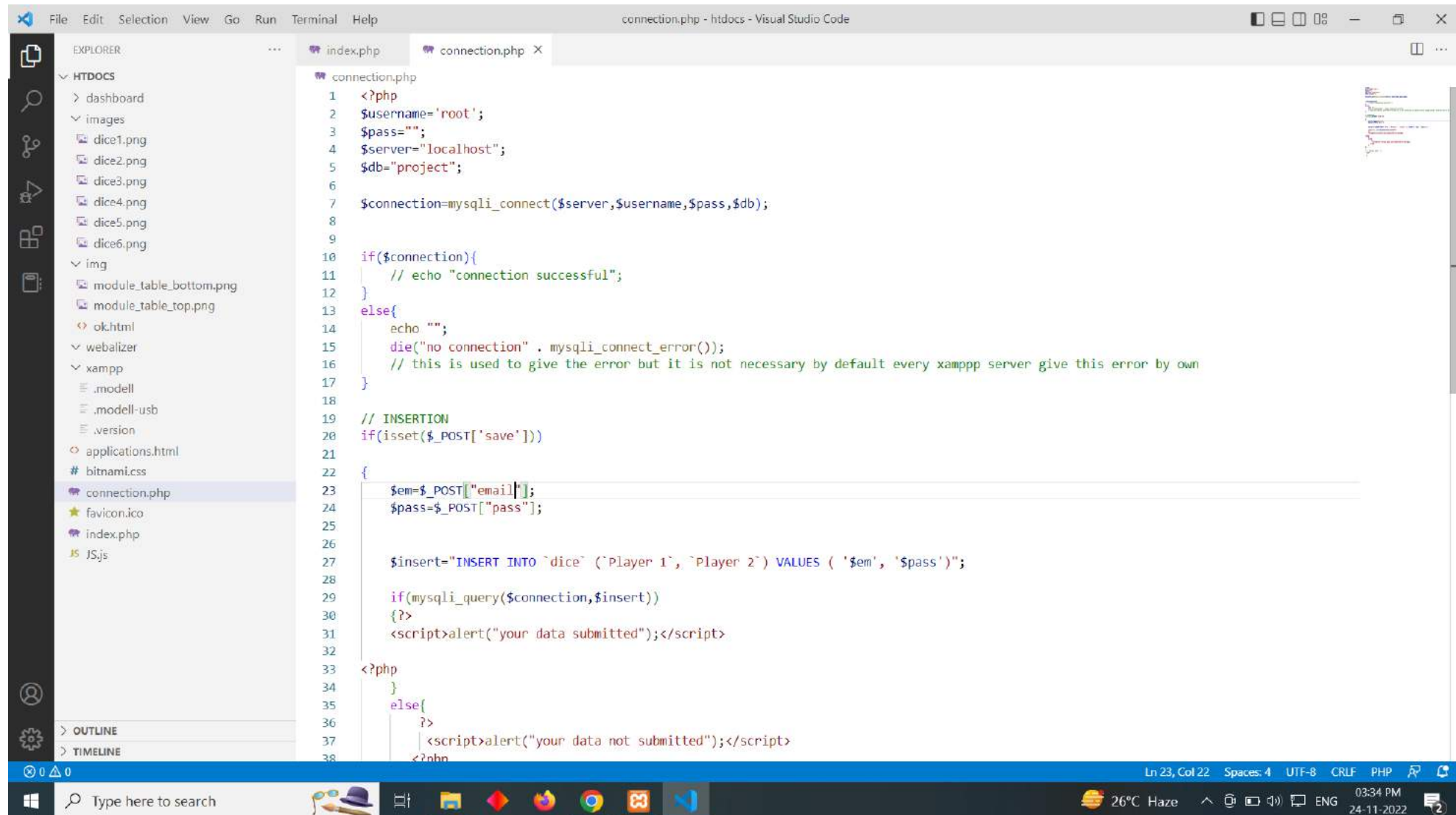
JS JS.js > myFunction
1  function myFunction(){
2    var r1=Math.random();
3    r1=Math.floor(r1*6);
4    var r2=Math.random();
5    r2=Math.floor(r2*6);
6    var URL=["./images/dice1.png","./images/dice2.png","./images/dice3.png",
7            "./images/dice4.png","./images/dice5.png","./images/dice6.png"]
8
9    var d1=document.getElementById("P1");
10   d1.setAttribute("src",URL[r1]);
11
12   var d2=document.getElementById("P2");
13   d2.setAttribute("src",URL[r2]);
14
15   if(r1>r2)
16   {
17     document.querySelector("h1").innerHTML="Player 1 Wins";
18   }
19   else if(r1<r2)
20   {
21     document.querySelector("h1").innerHTML="Player 2 Wins";
22   }
23   else if(r1==r2)
24   {
25     document.querySelector("h1").innerHTML="Draw!";
26   }
27 }

```

The screenshot shows the phpMyAdmin interface for a local MySQL database named 'project'. The 'dice' table is selected, and its contents are displayed. The table has three columns: 'S.no', 'Player 1', and 'Player 2'. The data is as follows:

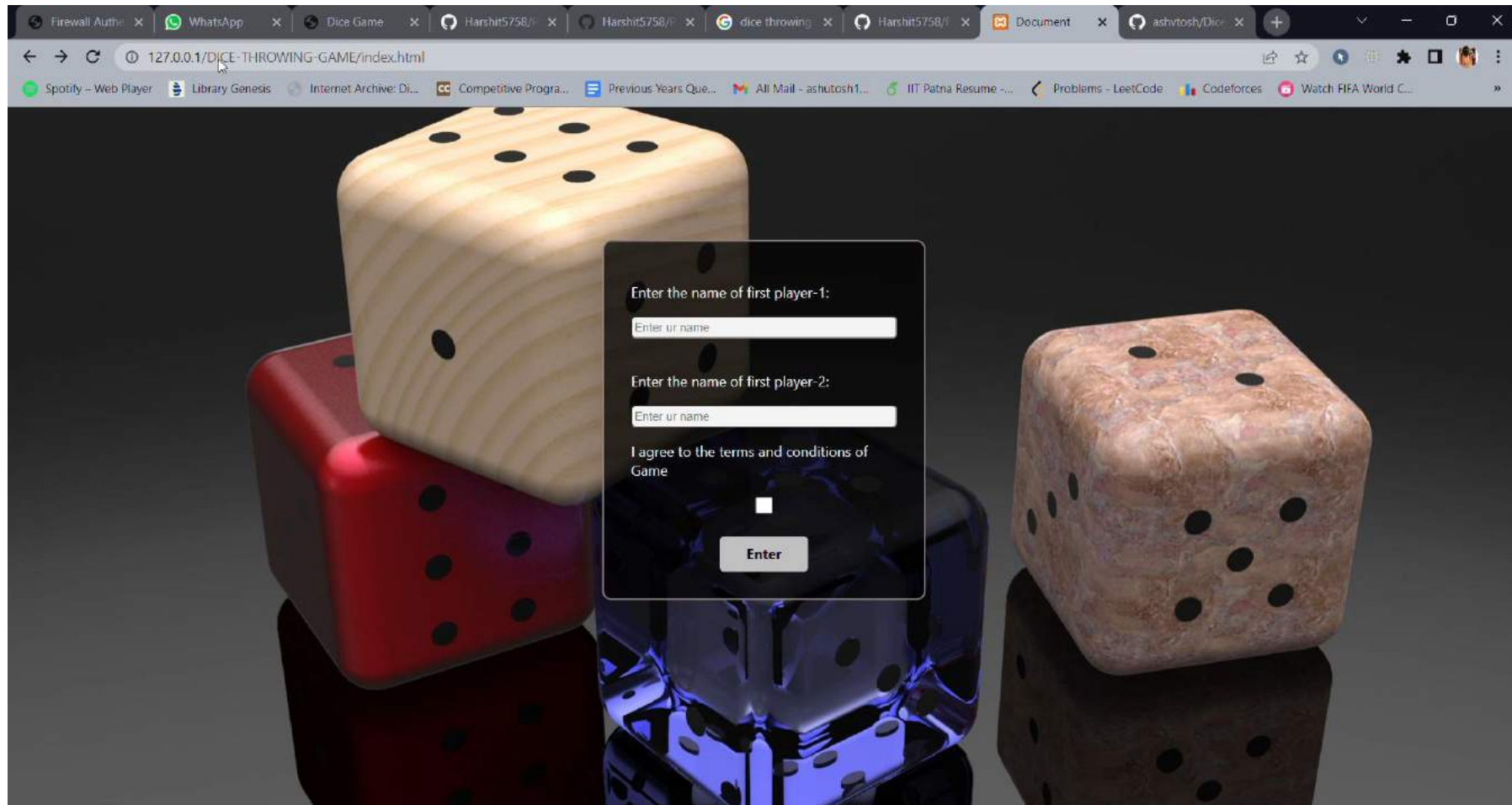
| S.no | Player 1 | Player 2 |
|------|----------|----------|
| 1    | HARSHIT  | SACHIN   |
| 4    | Gully    | Boy      |
| 5    | ansh     | samuel   |
| 6    | p1       |          |
| 7    |          |          |
| 8    |          | SACHIN   |
| 9    | HARSHIT  | SACHIN   |

The interface includes a left sidebar with a database tree, a top navigation bar with various tools (Browse, Structure, SQL, etc.), and a bottom status bar with system information and a search bar.



```
1  <?php
2  $username='root';
3  $pass="";
4  $server="localhost";
5  $db="project";
6
7  $connection=mysqli_connect($server,$username,$pass,$db);
8
9
10 if($connection){
11     // echo "connection successful";
12 }
13 else{
14     echo "";
15     die("no connection" . mysqli_connect_error());
16     // this is used to give the error but it is not necessary by default every xampp server give this error by own
17 }
18
19 // INSERTION
20 if(isset($_POST['save']))
21 {
22     $em=$_POST["email"];
23     $pass=$_POST["pass"];
24
25     $insert="INSERT INTO `dice` (`Player 1`, `Player 2`) VALUES ( '$em', '$pass')";
26
27     if(mysqli_query($connection,$insert))
28     {
29         <script>alert("your data submitted");</script>
30     }
31 }
32 <?php
33 }
34 else{
35     ?>
36     <script>alert("your data not submitted");</script>
37 }
38 <?php
```





**THANK YOU!!**