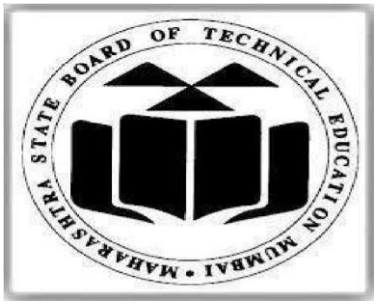


MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION

GOVERNMENT POLYTECHNIC, AWASARI (KH)

TAL-AMBEGAON, DIST – PUNE – 412405

ACADEMIC YEAR : 2021-22

- **COURSE NAME : JAVA PROGRAMMING**
- **COURSE CODE : 22412**
- **BRANCH : INFORMATION TECHNOLOGY (SY)**
- **TOPIC NAME : To develop a Game station (Tic_Tac_Toe And Guess The Number Game)**

DETAILS OF TEAM MEMBERS:

SR. NO.	NAMES OF GROUP MEMBERS	ROLL NO.	ENROLLMENT NO.
1	GHONGE KARISHMA RAJARAM	20IF213	2010510360
2	GITE AKSHAY SANGRAM	20IF214	2010510361
3	HULE GAYATRI KERBHAU	20IF215	2010510362
4	HULE SANIKA RAJENDRA	20IF216	2010510363
5	JADHAV KISHOR HARI	20IF217	2010510364
6	JADHAV PRANAV TANAJI	20IF218	2010510365

GUIDANCE BY :

MRS. P. S. GHODE

GROUP LEADER :

AKSHAY GITE

PART A – A MICRO PROJECT PROPOSAL

Aims / benefits of micro-project :

Aim: To develop a Game station (Tic_Tac_Toe And Guess The Number Game)

Benefits:

- First we understand concept of Java Language very well.
- We learn how to use Classes , Objects in Java Programming.
- We learn how to write java program and run on any machine.
- We learn how to apply concept of inheritance in Java.

Course Outcomes (COs) : -

CO1: Develop program using Object Oriented methodology in Java.
CO2: Apply concept of inheritance for code reusability.
CO3: Develop programs using multithreading.
CO4: Implement Exception Handling.
CO6: Develop programs for handling I/O and file streams.

Proposed Methodology: -

- First we select the topic Game station with the help of Teacher.
- Then we will install JDK and set JDK path using command prompt.
- We will discuss how to cover maximum Cos in our programme.
- Then we will start programming in Notepad.
- After that we will complete our programming and Run it properly.
- We will test our coding and games are proper work or not in game station.
- Then we will create part b of our project in soft copy and hard copy also and submit hard copy in college.

Action Plan: -

Sr. No.	details of activity	planned start date	Planned Finished date	Responsible Name of members
1	Group formation and allocation of Micro project title			All team members
2	Information search and required analysis.			All team members
3	Actual project / assembly project			All team members
4	Testing of project			All team Members
5	Acquire the printout and submit it.			All team members
6	Submission of project.			All team members

Resources Required: -

Sr.no.	Name of Resource	Specification	Qty.	Remarks
1	Software	Notepad. Visual Studio, JDK, Command Prompt	-	
2	Websites	Javatpoint,	-	
3	Book	**The Complete Reference : Java Seventh Edition By Herbert Schildt**	-	

Date :- / /

TEACHER	PRINCIPAL	H.O.D
MRS. P. S. GHODE	DR. D.R. NANDANWAR	DR.D.N.REWADKAR

DEPARTMENT OF INFORMATION TECHNOLOGY

GOVERNMENT POLYTECHNIC AWASARI (KHURD)

SEMESTER —IV (2021-22)

CERTIFICATE

This is to certify the following students of semester Fourth of Diploma in Information Technology of Institute : Government polytechnic, Awasari (kh) (code : 1051) has completed the micro project satisfactorily in subject- **JAVA PROGRAMMING** (**22412**) for the academic year 2021-22 as prescribed in the curriculum.

SR. NO.	NAMES OF GROUP MEMBERS	ROLL NO.	ENROLLMENT NO.
1	GHONGE KARISHMA RAJARAM	20IF213	2010510360
2	GITE AKSHAY SANGRAM	20IF214	2010510361
3	HULE GAYATRI KERBHAU	20IF215	2010510362
4	HULE SANIKA RAJENDRA	20IF216	2010510363
5	JADHAV KISHOR HARI	20IF217	2010510364
6	JADHAV PRANAV TANAJI	20IF218	2010510365

TEACHER	PRINCIPAL	H.O.D
MRS. P. S. GHODE	DR. D.R. NANDANWAR	DR.D.N.REWADKAR

Part – B Micro-Project Report

Aim: To develop a Game station (Tic_Tac_Toe And Guess The Number Game)

Rationale :-

Nowdays everyone needs to get entertained. Getting away from daily schedule, daily work is very important. For that purpose some people prefers Going out , playing Sports but most prefer Playing Computer Games.

That is why we created Game Station which is easy to understand and anyone can play it. This game station have Two games First one is Tic-Tac-Toe which is one of the most popular game and Second one is Number Guessing Game.

Tic-Tac Toe is very easy to understand and play. We will get assigned with One Character either X or O. Then we have to choose slot where we want to place our character. The main Aim of placing character is to align them either Horizontally, Vertically or diagonally.The first one to Do that win the game

Number Guessing Game is very Simple. Computer will generate any number between 1 to 100 and we just have guess the number and enter it. If we enter number greater / smaller than computer generated number, the game will warn us. And after Successfully Guessing Number it will count Turns took to guess the Number**.**

Course Outcomes (COs) : -

CO1: Develop program using Object Oriented methodology in Java.
CO2: Apply concept of inheritance for code reusability.
CO3: Develop programs using multithreading.
CO4: Implement Exception Handling.
CO6: Develop programs for handling I/O and file streams.

Proposed Methodology: -

- First we selected the topic Game station with the help of Teacher.
- Then we had installed JDK and set JDK path using command prompt.
- We had discussed how to cover maximum Cos in our program.
- Then we started programming in Notepad / Visual Studio.
- After that we completing our programming and we Ran it properly.
- We had tested our code and checked if everything is working properly or not.
- Then we created part b of our project in soft copy and Printed hard copy and submited hard copy in college.

➢ Actual Resources Required: -

Sr.no.	Name of Resource	Specification	Qty.	Remarks
1	Software	**Notepad. Visual Studio, JDK, Command Prompt**	-	
2	Websites	**Javatpoint**	-	
3	Book	**The Complete Reference : Java Seventh Edition By Herbert Schildt**	-	

INDEX

➢ Code

```
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;
import java.lang.Thread;

class BaseGame extends Thread
{
   char board[][]=new char[3][3];
   int winner=0;
   int slot;
   String name1,name2;
   int position=1;
   int correct=0;
   Scanner sc = new Scanner(System.in);
   File won = new File("Winners.txt"); //Develop program for handling io and file streams
   void welcome()
   {
      System.out.println("\n\n.....Welcome to Tic-Tac-Toe.....\n");
   }
   void takeInfo()
   {
      System.out.println("\n-----Please Enter the Name of First Player (X) : \n");
      name1=sc.next();
      System.out.println("\n-----Please Enter the Name of Second Player (O) : \n");
      name2=sc.next();
   }
   void printBoard()
   {
      System.out.println("\n");
      System.out.println("---------------------");
      for(int i=0;i<=2;i++)
      {
```

```
        for(int j=0;j<=2;j++)
        {
          System.out.print(" "+board[i][j]+"  |  ");
        }
        System.out.println("\n---------------------");
      }
    }
    void endGame()
    {
      System.out.println("\n-------------Thank you for Playing ;-)------------\n");
      System.out.println("\n-------------Have a Nice Day---------------\n");
    }
    void saveWinners(String winnerName,String loserName)
    {
      try
      {                                  //Implement Exception handling
        FileWriter fWriter = new FileWriter(won,true);      //Develope program for handling
io and file streams
        fWriter.write(winnerName+" vs "+loserName+"   -------     Winner :
"+winnerName+"\n");
        fWriter.close();
        System.out.println("\n.....Successfully Saved Statistics.....\n");
      }
      catch (IOException e)
      {
        System.out.println("An Error Occurred.");
        e.printStackTrace();
      }
    }
    void descBoard()
    {
      board[0][0]='1';
      board[0][1]='2';
      board[0][2]='3';
      board[1][0]='4';
      board[1][1]='5';
      board[1][2]='6';
      board[2][0]='7';
      board[2][1]='8';
```

```
      board[2][2]='9';
      printBoard();
   }
   public void run()                    //Develop programs using multithreading
   {
      System.out.println("\nGame Starting In : ");
      for(int i=5;i>=1;i--)
      {
         System.out.println(i);
         try
         {
            Thread.sleep(1000);
         }
         catch(InterruptedException e)
         {
            System.out.println(e);
         }
         if(i==1)
         {
            System.out.println("\nThank You For Waiting.....\n");
         }
      }
   }
}

class MainGame extends BaseGame
{                          //Apply Concept of inheritance for code reusabilty
   private void playTurn(String nameOfPlayer,char turn){
            System.out.println("\n"+nameOfPlayer);
            System.out.println("\nEnter Slot you want ("+turn+") in : \n");
            slot=sc.nextInt();
            switch(slot)
            {
               case 1: board[0][0]=turn;
                     correct=1;
                     break;
               case 2: board[0][1]=turn;
                     correct=1;
                     break;
```

```
            case 3: board[0][2]=turn;
                  correct=1;
                  break;
            case 4: board[1][0]=turn;
                  correct=1;
                  break;
            case 5: board[1][1]=turn;
                  correct=1;
                  break;
            case 6: board[1][2]=turn;
                  correct=1;
                  break;
            case 7: board[2][0]=turn;
                  correct=1;
                  break;
            case 8: board[2][1]=turn;
                  correct=1;
                  break;
            case 9: board[2][2]=turn;
                  correct=1;
                  break;
            default:
               System.out.println("\nYou've Entered Wrong Slot\nPlease Enter Between 1
to 9\n");
          }
   }
   void xTurn()
   {
        do
        {
           playTurn("-----------"+name1+"'s Turn--------",'X');
        }while(correct!=1);
   }
   void oTurn()
   {
      correct=0;
        do
        {
           playTurn("-----------"+name2+"'s Turn--------",'O');
```

```
            }while(correct!=1);
        }
    void checkX()
    {
       if((board[0][0]=='X' && board[0][1]=='X' && board[0][2]=='X') ||
(board[1][0]=='X'&&board[1][1]=='X'&&board[1][2]=='X') || (board[2][0]=='X' &&
board[2][1]=='X' && board[2][2]=='X') ||
          (board[0][0]=='X' && board[1][1]=='X' && board[2][2]=='X')  ||  (board[0][2]=='X'
&& board[1][1]=='X' && board[2][0]=='X') ||  (board[0][0]=='X' && board[1][0]=='X' &&
board[2][0]=='X') || (board[0][1]=='X' && board[1][1]=='X' && board[2][1]=='X') ||
(board[0][2]=='X' && board[1][2]=='X' && board[2][2]=='X'))
          {
            System.out.println("\nPlayer "+name1+" (O) Won The Game.....\n");
            winner=1;
            saveWinners(name1, name2);
          }
    }
    void checkO()
    {
          if((board[0][0]=='O' && board[0][1]=='O' && board[0][2]=='O') || (board[1][0]=='O'
&& board[1][1]=='O' && board[1][2]=='O') || (board[2][0]=='O' && board[2][1]=='O' &&
board[2][2]=='O') ||
            (board[0][0]=='O' && board[1][1]=='O' && board[2][2]=='O')  ||
(board[0][2]=='O' && board[1][1]=='O' && board[2][0]=='O') ||  (board[0][0]=='O' &&
board[1][0]=='O' && board[2][0]=='O') || (board[0][1]=='O' && board[1][1]=='O' &&
board[2][1]=='O') || (board[0][2]=='O' && board[1][2]=='O' && board[2][2]=='O'))
            {
              System.out.println("\nPlayer "+name2+ " (O) Won The
Game.....\n");
              winner=1;
              saveWinners(name2, name1);
            }
      }
      void checkTie()
      {
         if((board[0][0]=='X' || board[0][0]=='O') && (board[0][1]=='X' || board[0][1]=='O')
&& (board[0][2]=='X' || board[0][2]=='O') && (board[1][0]=='X' || board[1][0]=='O') &&
(board[1][1]=='X' || board[1][1]=='O') && (board[1][2]=='X' || board[1][2]=='O')
            && (board[2][0]=='X' || board[2][0]=='O') && (board[2][1]=='X' ||
board[2][1]=='O') && (board[2][2]=='X' || board[2][2]=='O'))
```

```
        {
          System.out.println("\n----------Game is Tie (Draw) Please Play Again----------
\n");
          winner=1;
        }
     }
}

class BaseGuessNumber
{
   int ComputerNumber,UserNumber,winner=0,turnsTook=0;
   String name;
   File turn = new File("Guess The Number Winners.txt");
   void welcome()
   {
      System.out.println("\n.....Welcome to Guess The Number
Game.....");
   }
   void generateNumber()
   {
      ComputerNumber=(int)(Math.random()*100);
   }
   void getName()
   {
      System.out.println("\nEnter Your Name : ");
      Scanner sc = new Scanner(System.in);
      name = sc.next();
   }
}
class MainGuessNumber extends BaseGuessNumber
{
   void runGame()
   {
      Scanner sc = new Scanner(System.in);
      do
      {
         System.out.println("\nEnter Your Number : ");
         UserNumber=sc.nextInt();
         if(UserNumber>100 || UserNumber<1)
```

```
        {
          System.out.println("\nPlease Enter Number Between 1 to 100\n");
        }
        else if(UserNumber>ComputerNumber)
        {
          System.out.println("\nThe Number You Entered is Bigger than Computer
Generated Number");
          turnsTook+=1;
        }
        else if(ComputerNumber>UserNumber)
        {
          System.out.println("\nThe Number You entered is Smaller than Computer
Generated Number");
          turnsTook+=1;
        }
        else
        {
          System.out.println("\nYou Guessed Right Number :)\n");
          winner=1;
          turnsTook+=1;
        }
      }while(winner!=1);
    }
    void calculateTurns()
    {
      System.out.println("\nYou Took "+turnsTook+" Turns to Guess Number Correctly\n");
    }
    void saveTurnStat()
    {
      try{                                    //Implement Exception handling
        FileWriter fWriter = new FileWriter(turn,true);      //Develope program for handling
io and file streams
        fWriter.write(name+" Took "+turnsTook+" Turns To win the Game\n");
        fWriter.close();
        System.out.println("\n.....Successfully Saved Statistics.....\n");
      }
      catch (IOException e)
      {
        System.out.println("An error occurred.");
```

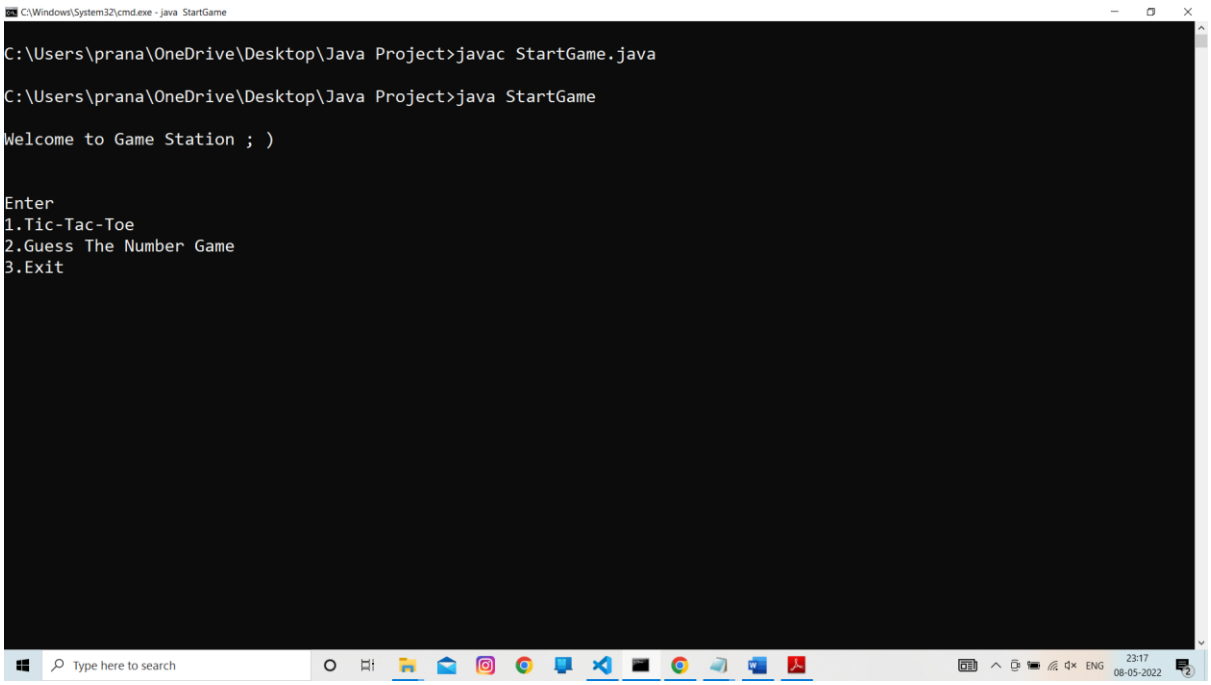
```
            e.printStackTrace();
        }
    }
}

class StartGame
{
    public static void main(String args[])
    {
        MainGame game = new MainGame();                    // Develop program using Object
Oriented methodology in Java
        MainGuessNumber guessGame = new MainGuessNumber();
        Scanner sc = new Scanner(System.in);
        System.out.println("\nWelcome to Game Station ; ) \n");
        System.out.println("\nEnter \n1.Tic-Tac-Toe  \n2.Guess The Number Game \n3.Exit
");
        int choiceGame;
        choiceGame = sc.nextInt();
        do{
            if(choiceGame==1 || choiceGame==2)
            {
                game.start();
                try
                {
                    game.join();
                }
                catch(InterruptedException e)
                {
                    System.out.println(e);
                }
            }
            switch(choiceGame)
            {
                case 1 :
                        game.welcome();
                        game.descBoard();
                        game.takeInfo();
                        do{
                            game.xTurn();
```

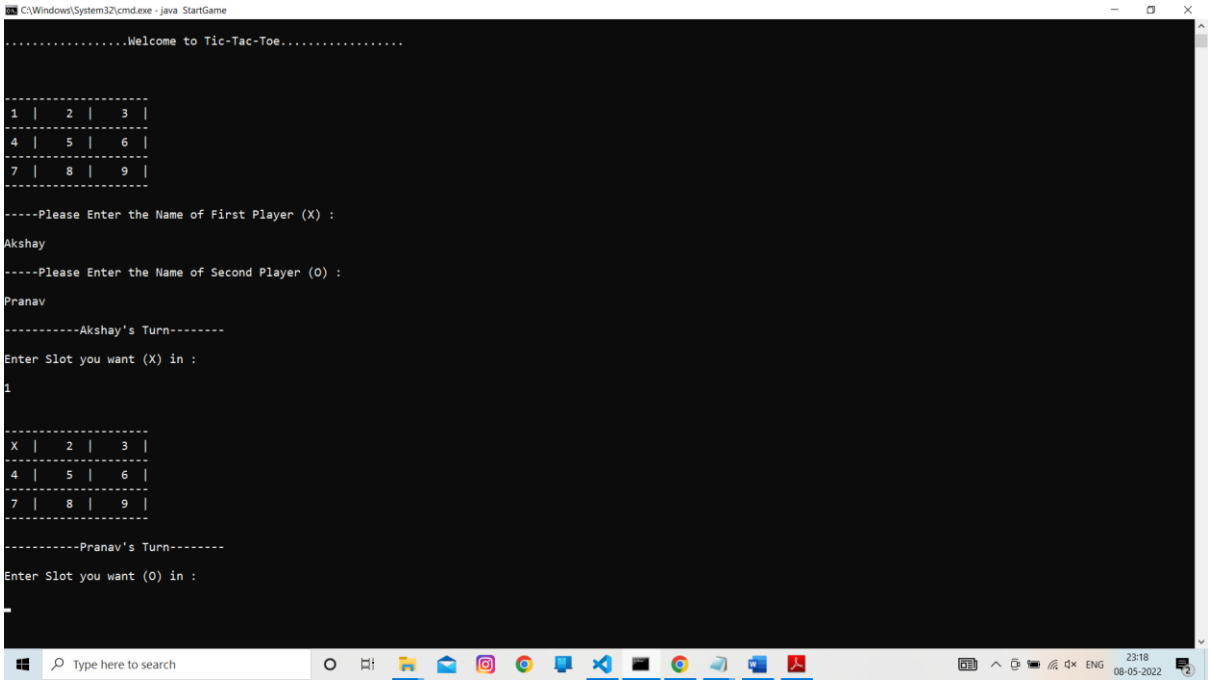
```
                game.printBoard();
                game.checkX();
                game.checkTie();
                if(game.winner!=1){
                  game.oTurn();
                  game.printBoard();
                  game.checkO();
                  game.checkTie();
                }
              }while(game.winner!=1);
              game.endGame();
              choiceGame=3;
          break;
          case 2 :
              guessGame.welcome();
              guessGame.generateNumber();
              guessGame.getName();
              guessGame.runGame();
              if(guessGame.winner==1)
              {
                guessGame.calculateTurns();
                guessGame.saveTurnStat();
              }
              game.endGame();
              choiceGame=3;
          break;
          case 3 :
              System.out.println("\nSuccessfully Exited The Game.....");
          break;
          default :
              System.out.println("\n Please Enter \n1.Tic-Tac-Toe  \n2.Guess The Number
Game \n3.Exit ");
              choiceGame=4;
        }
       }
     while(choiceGame!=3);
  }
}
```

➢ ScreenShots

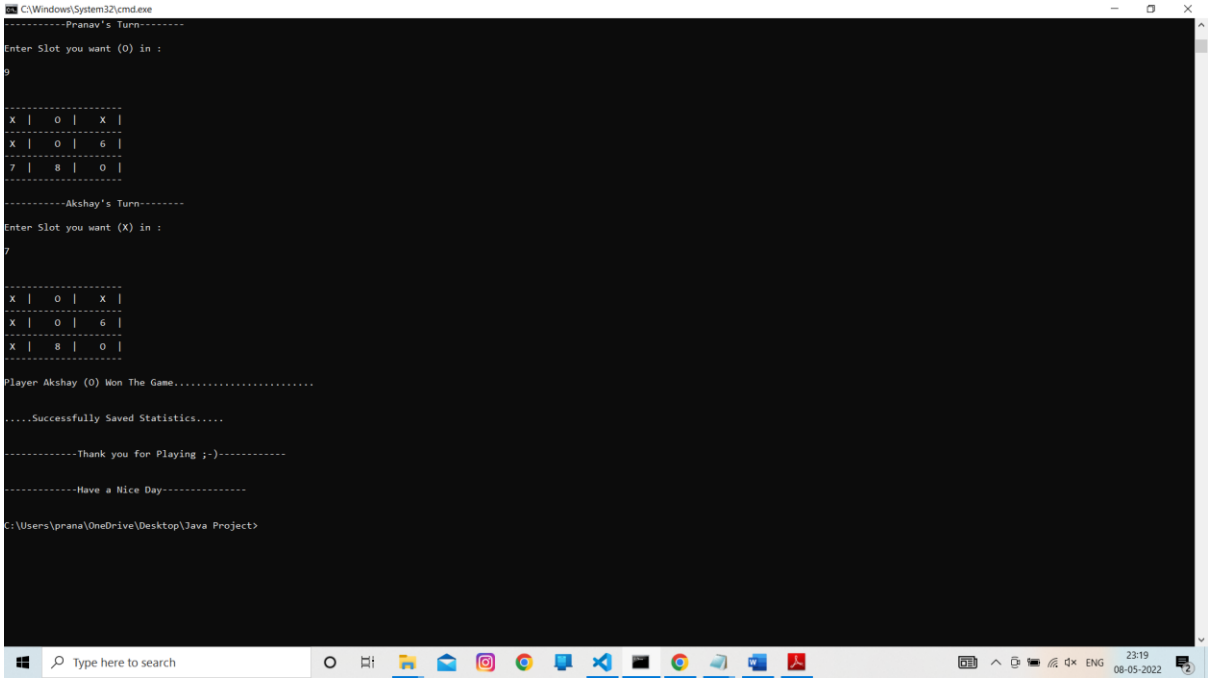
- **Starting Menu**



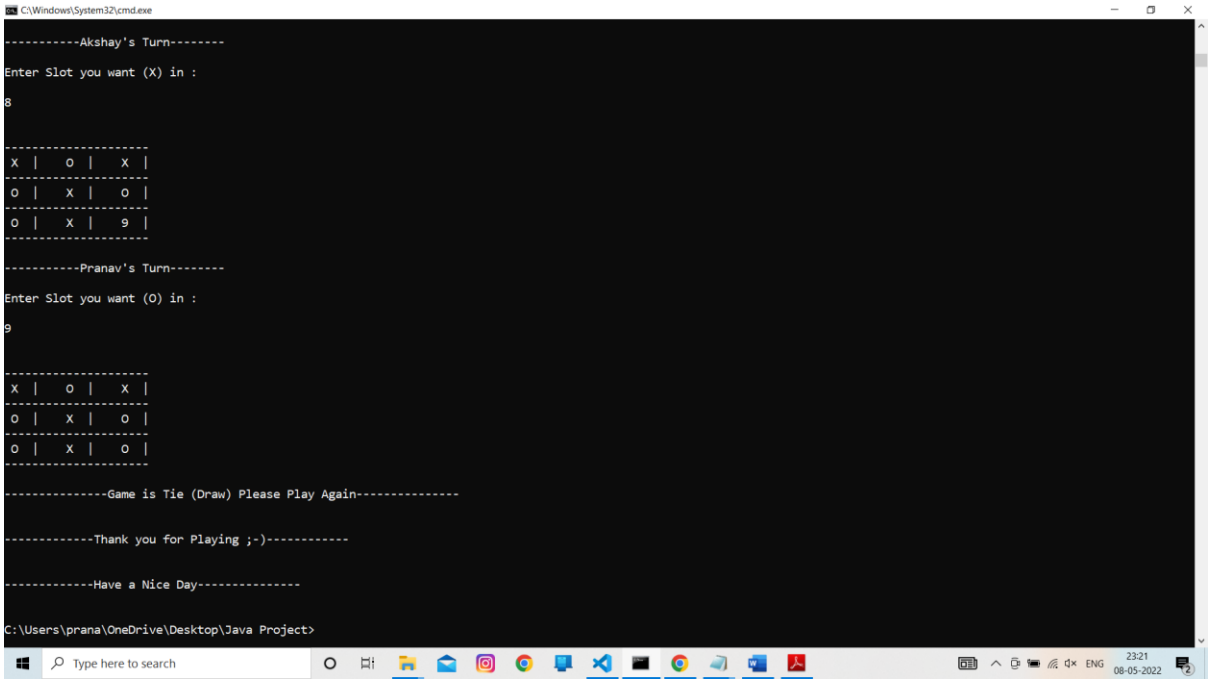
- **Tic-Tac-Toe**



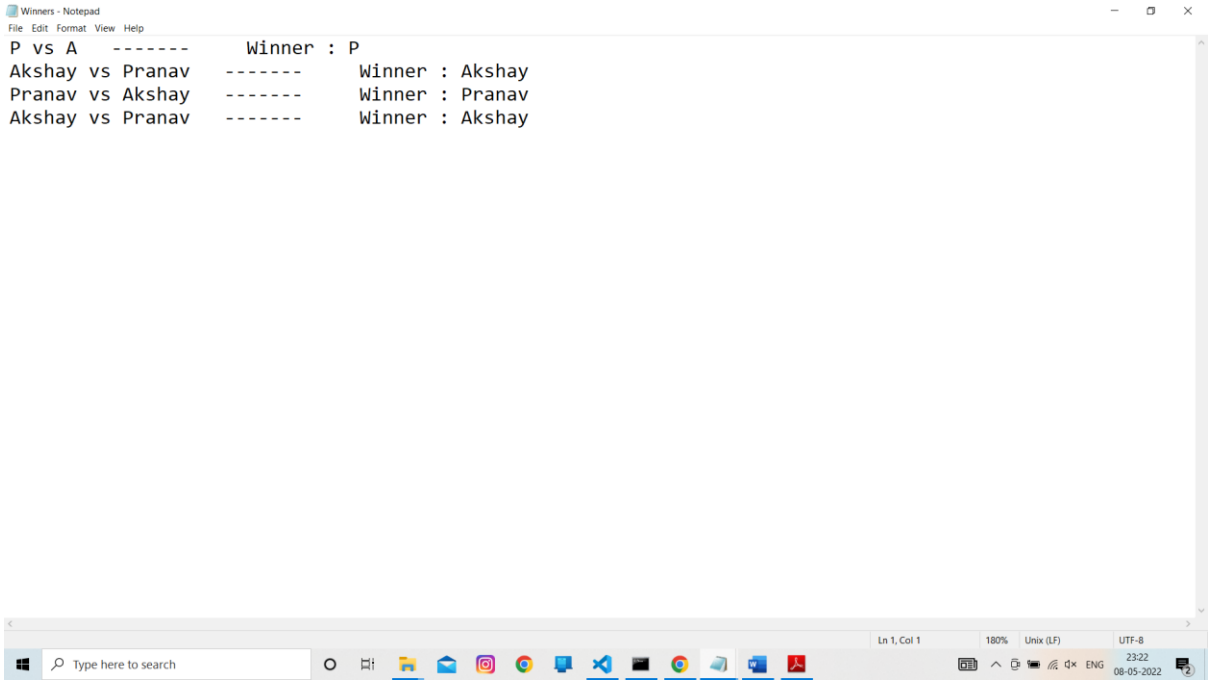
- **Winnig Tic-Tac-Toe**



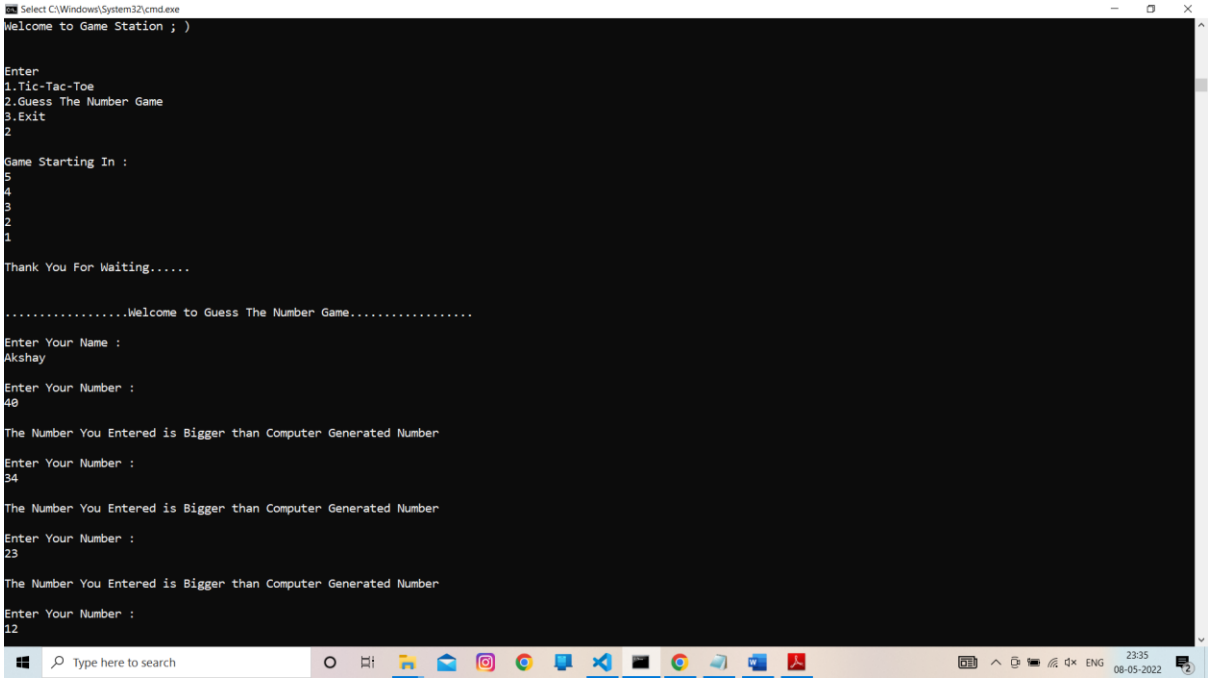
- **Tie Game**

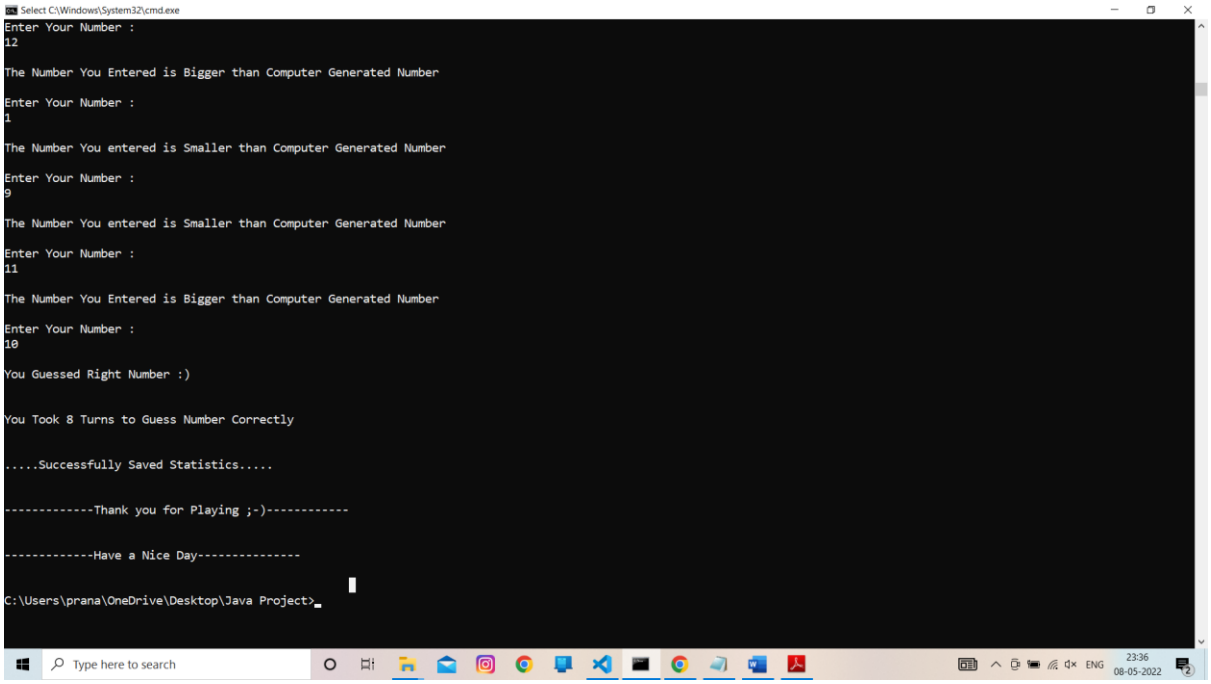


- **File Created**

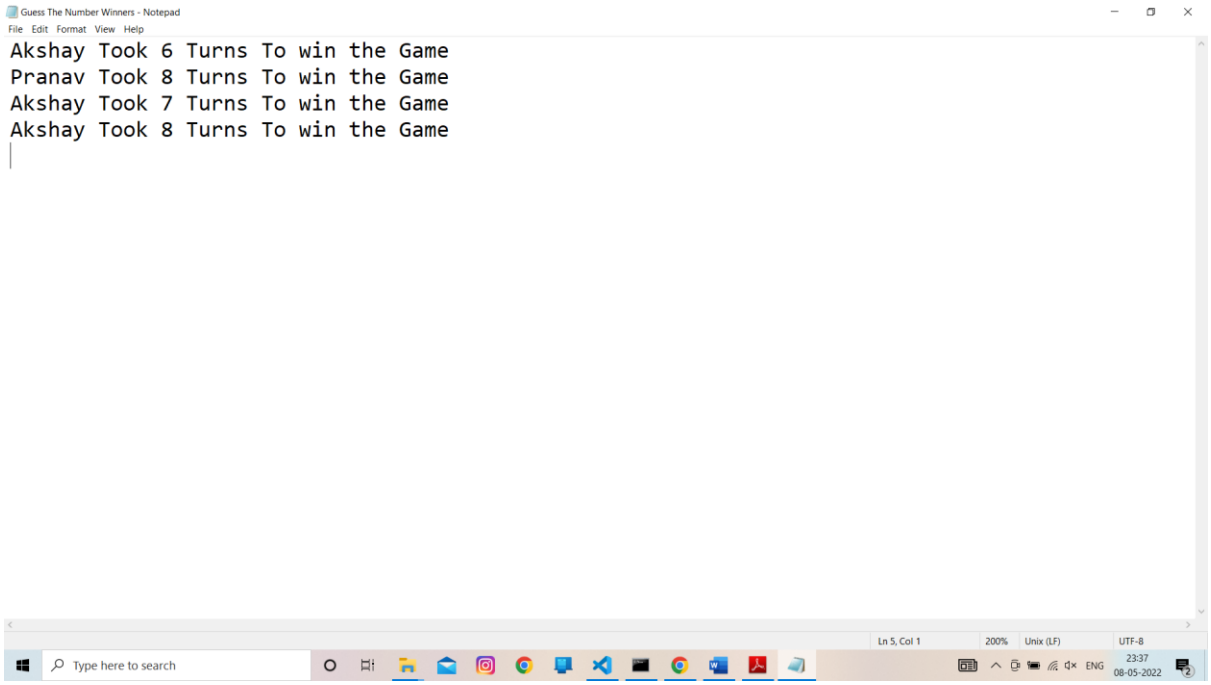


➢ Guess the Number

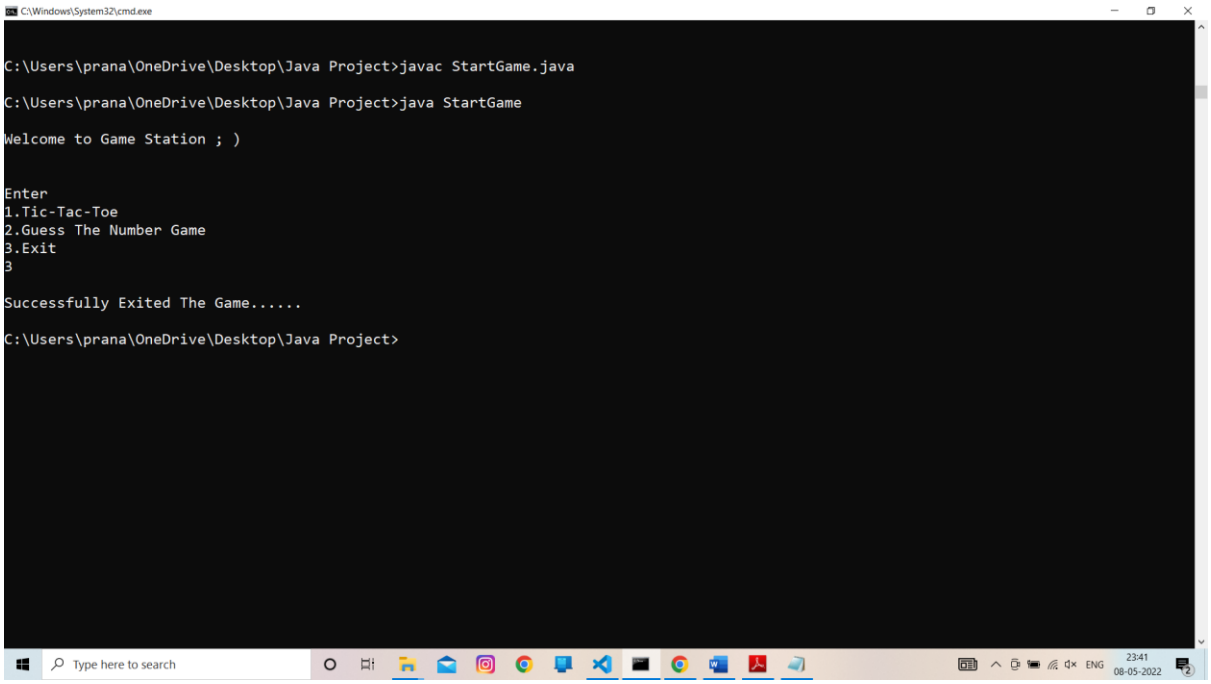




- **File created for Guess the Number**



➢ Exit Game Station



➢ Conclusion

We had created Game Station in Java .Which have Two Games ,First one is Tic-Tac-Toe and Second one is Guess The Number. We used OOP , Applied Inheritance concept ,Multithreading ,Exception Handling and File Handling. We Used File handling Make Sure We Store Every Game Result in TXT Format.

This Project helped us to understand 5 Course Outcomes and Helped us to build our Coding Knowledge.