```java
import java.io.*;
import java.util.*;

//HOW MY PROGRAM CHOOSES ITS MOVES:
/*
STEP 0
The program starts by choosing a random int between 0 and 8 and makes sure that the
space of the number is empty.

STEP 1
The program then checks to see if any moves will instantly win the AI player the
game. It does this by
using a function called checkDanger. Each space on the board has two variables
defined in the Jerry class called
isDangerO and isDangerX. The checkDanger function will check to see if each space
is dangerous for a specific character (X or O) and
update the space's variable.

The AI player uses whatever character the player has selected (X or O) on the
checkDanger function and then
checks each space to make sure no spots are dangerous for the player. If this is
the case, it moves on to step 2.
If this is not the case, the program checks if the chosen spot (the random int
mentioned above) is dangerous to the player
if it is not, it goes back to step 0. If it is, then that space is selected as the
AIs move.

STEP 2
If no spaces are dangerous to the player, the program checks to see if any spaces
are dangerous to the AI. It does the
same method as above but replaces the character to check with the AIs character.

If no spaces are dangerous to the AI, it moves on to step 3. If any spaces are
dangerous to the AI, then it checks to see
if the space at the random int is dangerous to the AI. If it is not, then it goes
back to step 0. If it is, then
that space is selected as the AIs move.


STEP 3
If no spaces are dangerous to either player, then the AI goes with the random int
as the space to mark.

*/

class JerryTacToe
{
      public static void displayBoard(Jerry [] board){
            System.out.println("| " +board[0].getChar() + " |---| " +
board[1].getChar() + " |---| " + board[2].getChar() + " |");
            System.out.println("");
            System.out.println("| " +board[3].getChar() + " |---| " +
board[4].getChar() + " |---| " + board[5].getChar() + " |");
            System.out.println("");
            System.out.println("| " +board[6].getChar() + " |---| " +
board[7].getChar() + " |---| " + board[8].getChar() + " |");
      }
      public static void playerturn(Jerry [] board, char t){
            char notT = 'O';
            if(t == 'O'){notT = 'X';}
            int playerNum = 0;
            Scanner cin=new Scanner(System.in);
            boolean a = true;
            System.out.println("Pick a number between 1 and 9");
            while(a){
                  playerNum = cin.nextInt();
                  if(playerNum < 10 && playerNum >0){
                        a = false;
                  }else{System.out.println("ERROR: invalid number. Please pick
another.");}
                  if(board[playerNum-1].getChar() != 'e'){
                        a = true;
                        System.out.println("ERROR: invalid number. Please pick
another.");
                  }
            }

            board[playerNum-1].setMark(t);
            board[playerNum-1].setDanger(notT, false);

      }


      public static void checkDanger(Jerry[] board, char t){
            char notT = 'O';
            if(t == 'O'){notT = 'X';}
            //check 1
            if(board[0].getChar() == 'e'){
                  //123
                  if((board[1].getChar() == notT)&&(board[2].getChar() == notT))
{board[0].setDanger(t, true);}
                  //148
                  if((board[3].getChar() == notT)&&(board[7].getChar() == notT))
{board[0].setDanger(t, true);}
                  //159
                  if((board[4].getChar() == notT)&&(board[8].getChar() == notT))
{board[0].setDanger(t, true);}
            }
            //check 2
            if(board[1].getChar() == 'e'){
                  //247
                  if((board[3].getChar() == notT)&&(board[6].getChar() == notT))
{board[1].setDanger(t, true);}
                  //269
                  if((board[5].getChar() == notT)&&(board[8].getChar() == notT))
{board[1].setDanger(t, true);}
                  //123
                  if((board[0].getChar() == notT)&&(board[2].getChar() == notT))
{board[1].setDanger(t, true);}
            }
            //check 3
            if(board[2].getChar() == 'e'){
                  //357
                  if((board[4].getChar() == notT)&&(board[6].getChar() == notT))
{board[2].setDanger(t, true);}
```

```
                  //368
                  if((board[5].getChar() == notT)&&(board[7].getChar() == notT))
{board[2].setDanger(t, true);}
                  //123
                  if((board[0].getChar() == notT)&&(board[1].getChar() == notT))
{board[2].setDanger(t, true);}
            }
            //check 4
            if(board[3].getChar() == 'e'){
                  //456
                  if((board[4].getChar() == notT)&&(board[5].getChar() == notT))
{board[3].setDanger(t, true);}
                  //148
                  if((board[0].getChar() == notT)&&(board[7].getChar() == notT))
{board[3].setDanger(t, true);}
                  //247
                  if((board[1].getChar() == notT)&&(board[6].getChar() == notT))
{board[3].setDanger(t, true);}
            }
            //check 5
            if(board[4].getChar() == 'e'){
                  //456
                  if((board[3].getChar() == notT)&&(board[5].getChar() == notT))
{board[4].setDanger(t, true);}
                  //159
                  if((board[0].getChar() == notT)&&(board[8].getChar() == notT))
{board[4].setDanger(t, true);}
                  //357
                  if((board[2].getChar() == notT)&&(board[6].getChar() == notT))
{board[4].setDanger(t, true);}
            }
            //check 6
            if(board[5].getChar() == 'e'){
                  //456
                  if((board[3].getChar() == notT)&&(board[4].getChar() == notT))
{board[5].setDanger(t, true);}
                  //368
                  if((board[2].getChar() == notT)&&(board[7].getChar() == notT))
{board[5].setDanger(t, true);}
                  //269
                  if((board[1].getChar() == notT)&&(board[8].getChar() == notT))
{board[5].setDanger(t, true);}
            }
            //check 7
            if(board[6].getChar() == 'e'){
                  //789
                  if((board[7].getChar() == notT)&&(board[8].getChar() == notT))
{board[6].setDanger(t, true);}
                  //247
                  if((board[1].getChar() == notT)&&(board[3].getChar() == notT))
{board[6].setDanger(t, true);}
                  //357
                  if((board[2].getChar() == notT)&&(board[4].getChar() == notT))
{board[6].setDanger(t, true);}
            }
            //check 8
            if(board[7].getChar() == 'e'){
                  //789
                  if((board[6].getChar() == notT)&&(board[8].getChar() == notT))
```

```
{board[7].setDanger(t, true);}
                  //148
                  if((board[0].getChar() == notT)&&(board[3].getChar() == notT))
{board[7].setDanger(t, true);}
                  //368
                  if((board[2].getChar() == notT)&&(board[5].getChar() == notT))
{board[7].setDanger(t, true);}
            }
            //check 9
            if(board[8].getChar() == 'e'){
                  //789
                  if((board[6].getChar() == notT)&&(board[7].getChar() == notT))
{board[8].setDanger(t, true);}
                  //159
                  if((board[0].getChar() == notT)&&(board[4].getChar() == notT))
{board[8].setDanger(t, true);}
                  //269
                  if((board[1].getChar() == notT)&&(board[5].getChar() == notT))
{board[8].setDanger(t, true);}
            }
      }

      public static void AIturn(Jerry [] board, char t){
            char notT = 'O';
            if(t == 'O'){notT = 'X';}

            checkDanger(board, notT);
            checkDanger(board, t);
            boolean validOption = false;
            int chosenSpot = 0;


            while(validOption == false){
                  chosenSpot = (int)(9*Math.random());
                  //is the option empty?
                  if(board[chosenSpot].getChar()!='e'){
                        //do nothing try again
                  }else{
                  //are there any options dangerous to the PLAYER
                        if(board[0].isDanger(notT) || board[1].isDanger(notT) ||
board[2].isDanger(notT) || board[3].isDanger(notT) ||
                              board[4].isDanger(notT) || board[5].isDanger(notT) ||
board[6].isDanger(notT) || board[7].isDanger(notT) ||
                              board[8].isDanger(notT)){
                                    //if yes, then it NEEDS to pick one that is
dangerous to the player.
                                    if(board[chosenSpot].isDanger(notT))
{validOption = true;}
                  }else{
                        //there are NOT any that are dangerous to the player.
                        //check if there are any that are dangerous to the AI.
                        if(board[0].isDanger(t) || board[1].isDanger(t) ||
board[2].isDanger(t) || board[3].isDanger(t) ||
                              board[4].isDanger(t) || board[5].isDanger(t) ||
board[6].isDanger(t) || board[7].isDanger(t) ||
                              board[8].isDanger(t)){
                                    //if yes, it NEEDS to pick one that is
dangerous to the AI.
                                    if(board[chosenSpot].isDanger(t)){validOption =
true;}
                }else{
                    validOption = true;
                }

            }
            }








        }


        board[chosenSpot].setMark(t);
        board[chosenSpot].setDanger(t, false);
    }



    //checks to see if a player has won the game.
    public static boolean checkForWin(Jerry [] board, char t){

        //check 1
        if(board[0].getChar() == t){
            //123
            if((board[1].getChar() == t)&&(board[2].getChar() == t)){
                System.out.println(t + " Player won the game!");
                return true;}
            //148
            if((board[3].getChar() == t)&&(board[7].getChar() == t)){
                System.out.println(t + " Player won the game!");
                return true;}
            //159
            if((board[4].getChar() == t)&&(board[8].getChar() == t)){
                System.out.println(t + " Player won the game!");
                return true;}
        }
        //check 2
        if(board[1].getChar() == t){
            //247
            if((board[3].getChar() == t)&&(board[6].getChar() == t)){
                System.out.println(t + " Player won the game!");
                return true;}
            //269
            if((board[5].getChar() == t)&&(board[8].getChar() == t)){
                System.out.println(t + " Player won the game!");
                return true;}
        }
        //check 3
        if(board[2].getChar() == t){
            //357
            if((board[4].getChar() == t)&&(board[6].getChar() == t)){
                System.out.println(t + " Player won the game!");
                return true;}
```

```
                  //368
                  if((board[5].getChar() == t)&&(board[7].getChar() == t)){
                        System.out.println(t + " Player won the game!");
                        return true;}
            }
            //check 4
            if(board[3].getChar() == t){
                  //456
                  if((board[4].getChar() == t)&&(board[5].getChar() == t)){
                        System.out.println(t + " Player won the game!");
                        return true;}
            }
            //check 7
            if(board[6].getChar() == t){
                  //789
                  if((board[7].getChar() == t)&&(board[8].getChar() == t)){
                        System.out.println(t + " Player won the game!");
                        return true;}
                  //368
            }

            if(board[0].getChar() != 'e' && board[1].getChar() != 'e' && 
board[2].getChar() != 'e' && board[3].getChar() != 'e' &&
            board[4].getChar() != 'e' && board[5].getChar() != 'e' && 
board[6].getChar() != 'e' && board[7].getChar() != 'e' &&
            board[8].getChar() != 'e'){

                  System.out.println("----------");
                  System.out.println("Draw match!");
                  System.out.println("----------");
                  return true;
            }
            return false;
      }





public static void main(String [] args) throws IOException
  {
      char userChar = 'a';
      boolean gameEnded = false;
      Jerry [] board =new Jerry[9];



      for(int i=0; i<9; i++)
    {
      board[i]= new Jerry();
    }

      while(userChar != 'Q'){
            System.out.println("Welcome to Jerry Tac Toe! Enter R for rules or 
enter S to start.");
            Scanner cin=new Scanner(System.in);
            userChar = cin.nextLine().charAt(0);
            if(userChar == 'R'){
            System.out.println("The rules to Jerry tac toe are similar to Tic Tac 
Toe.");
            System.out.println("");
            System.out.println("The player who starts first uses X and the player
who starts second uses O");
            System.out.println("");
            System.out.println("The first player to get a straight line filled with
their 3 letters wins the game.");
            System.out.println("");
            System.out.println("The only difference between Jerry tac toe and Tic
tac toe is the board.");
            System.out.println("");
            System.out.println("The board for Jerry tac toe can be found at: ");
            System.out.println("https://blue.butler.edu/~jsorenso/oop/jtt.gif");
            System.out.println("");
            System.out.println("Only straight lines count as a win, so for
example:");
            System.out.println("148 is a winning combination. 147 is not.");
            System.out.println("");
            }

            if(userChar == 'S'){

                  //sets the board ready for the game.
                  for(int i=0; i<9; i++)
                        {
                        board[i].resetBoard();
                        }


                  boolean a = false;
                  //start game
                  int playerNum = 0;
                  int AINum = 0;
                  while(a == false){
                  playerNum = 0;
                  AINum = 0;

                  System.out.println("Would you like to be the first player or
second player?");
                  System.out.println("Type 1 for player 1 and type 2 for player
2");
                  playerNum = cin.nextInt();
                  if(playerNum == 1){
                  System.out.println("You chose player 1. You will be X's");
                        AINum = 2;
                        a = true;}
                  else if(playerNum == 2){
                  System.out.println("You chose player 2. You will be O's");
                        AINum = 1;
                        a = true;
                  }
                  else{System.out.println("ERROR: not a valid option.");}
                  }





                  gameEnded = false;
                  while(gameEnded == false){

                        if(playerNum == 1){
                        System.out.println("Player turn: ");
                        System.out.println("----------");

                        playerturn(board, 'X');
                        displayBoard(board);
                        if(checkForWin(board, 'X')){
                              gameEnded = true;}
                        else{
                              System.out.println("AI turn: ");
                              System.out.println("----------");

                              AIturn(board, 'O');
                              displayBoard(board);
                              if(checkForWin(board, 'O')){
                              gameEnded = true;}
                              }



                        }else{
                        System.out.println("AI turn: ");
                        System.out.println("----------");


                        AIturn(board, 'X');
                        displayBoard(board);
                        if(checkForWin(board, 'X')){
                              gameEnded = true;}
                        else{
                        System.out.println("Player turn: ");
                        System.out.println("----------");
                        playerturn(board, 'O');
                        displayBoard(board);
                        if(checkForWin(board, 'O')){
                              gameEnded = true;}
                        }


                        }

                  }


                  //end of game
                  Scanner b=new Scanner(System.in);
                  System.out.println("Type Q to exit the program or anything else
to play again.");

                  userChar = b.nextLine().charAt(0);

            }
      }
```

```
  }
}
```