MEMORY GAME

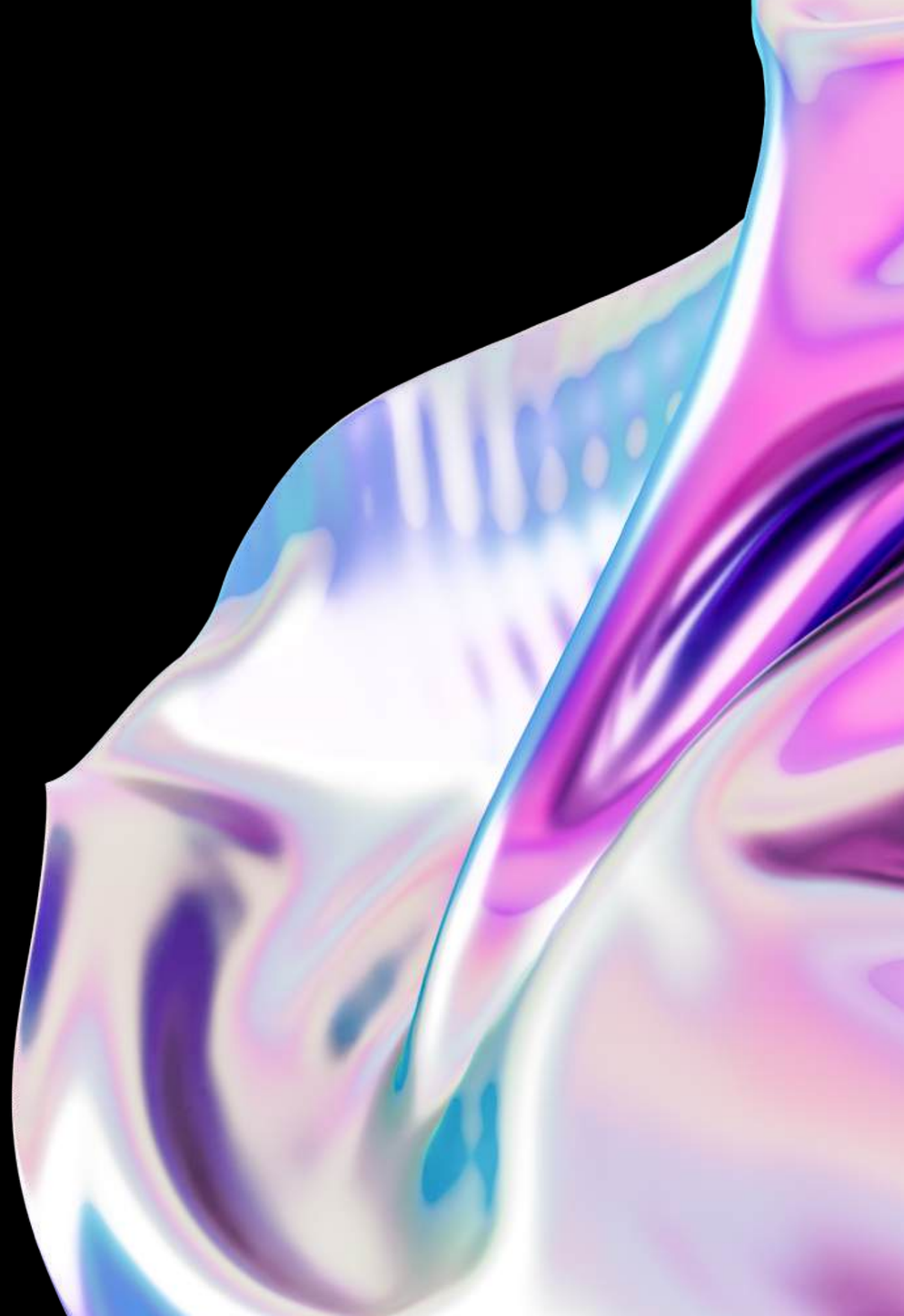
java programming

The Memory Game is a game where the player has to flip over pairs of cards with the same symbol:

- We create two arrays to represent the game board: board and flipped. The board is an array of strings that represents the state of the game board at any given time.
- When a player flips a card, we replace the corresponding element of the board array with the value of the card.
- Flipped is a boolean array that keeps track of which cards have been flipped over.
- All elements of the flipped array are false. When a player flips a card, we set the corresponding element of the flipped array to true.

Code Explanation

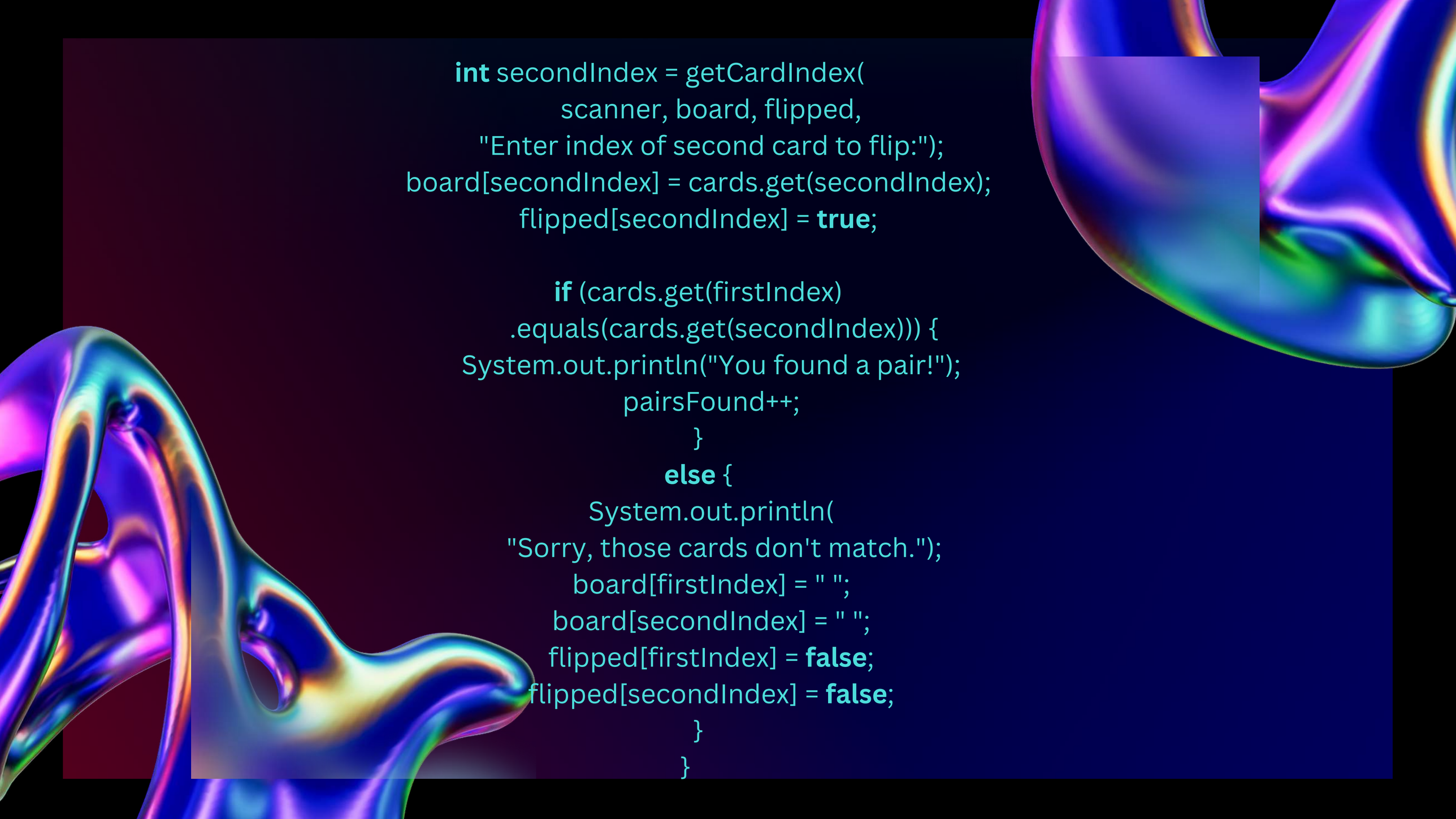
the MemoryGame class represents the game logic. The constructor takes an argument number which specifies the number of pairs of cards in the game. The play method runs the game loop, prompting the user to choose two cards and reveal them, checking if they match, and hiding them again if they don’t. The complete method checks if all the cards have been shown. The primary method creates a new MemoryGame object with 6 pairs of cards and starts the game by calling the play method.



Program:

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Scanner;

// class name : memorygame
public class MemoryGame {

public static void main(String[] args)
{

Scanner scanner = new Scanner(System.in);
ArrayList<String> cards = new ArrayList<>();
cards.add("A");
cards.add("A");
cards.add("B");
cards.add("B");
cards.add("C");
```

```
cards.add("C");
cards.add("D");
cards.add("D");
Collections.shuffle(cards);

String[] board = new String[cards.size()];
boolean[] flipped = new boolean[cards.size()];
int pairsFound = 0;

while (pairsFound < 4) {
printBoard(board);
int firstIndex = getCardIndex(
scanner, board, flipped,
"Enter index of first card to flip:");
board[firstIndex] = cards.get(firstIndex);
flipped[firstIndex] = true;
printBoard(board);
```



```
int secondIndex = getCardIndex(
        scanner, board, flipped,
    "Enter index of second card to flip:");
board[secondIndex] = cards.get(secondIndex);
flipped[secondIndex] = true;

if (cards.get(firstIndex)
    .equals(cards.get(secondIndex))) {
System.out.println("You found a pair!");
    pairsFound++;
    }
    else {
    System.out.println(
    "Sorry, those cards don't match.");
    board[firstIndex] = " ";
    board[secondIndex] = " ";
    flipped[firstIndex] = false;
    flipped[secondIndex] = false;
    }
}
```

```
        // win
        System.out.println("Congratulations, you won!");
    }

    public static int getCardIndex(Scanner scanner,
                                   String[] board,
                                   boolean[] flipped,
                                   String prompt)
    {
        int index;
        while (true) {
            System.out.println(prompt);
            index = scanner.nextInt();
            if (index < 0 || index >= board.length) {
                System.out.println(
                    "Invalid index, try again.");
            }
```

```
else if (flipped[index]) {
            System.out.println(
"Card already flipped, try again.");
}
else {
 break;
}
}
return index;
}

public static void printBoard(String[] board)
{
for (int i = 0; i < board.length; i++) {
System.out.print("| " + board[i] + " ");
}
System.out.println("|");
}
}
```

OUTPUT;

```
| null | null | null | null | null |
er index of first card to flip:

| null | null | null | null | null | null | null |
er index of second card to flip:

found a pair!
| C | null | null | null | null | null | null |
er index of first card to flip:

| C | B | null | null | null | null | null |
er index of second card to flip:

ry, those cards don't match.
| C |   |   | null | null | null | null |
er index of first card to flip:

| C |   |   | D | null | null | null |
er index of second card to flip:

ry, those cards don't match.
| C |   |   |   |   | null | null |
er index of first card to flip:

| C |   |   |   |   | A | null |
er index of second card to flip:
```

Thank you!

Team Members:

Victoria -AA1-19
Josephin-AA1-50
Keerthana-AA1-29
Steffie shreya-AA1-49
Hemanath.M-AA1-54