slip7

```
import javax.swing.*;
import java.awt.*;

public class s7q1 {
    public static void main(String[] args) {
        JFrame frame = new JFrame("Text with Label");
        frame.setSize(400, 400);

        JPanel panel = new JPanel();
        panel.setBackground(Color.RED);

        JLabel label = new JLabel("Dr. D Y Patil College");
        label.setFont(new Font("Serif", Font.BOLD, 20));

        panel.add(label);
        frame.add(panel);
        frame.setVisible(true);
    }
}
```

```
import java.util.Scanner;

class CricketPlayer {
    int pid;
```

```
    String pname;
    int totalRuns;
    int inningsPlayed;
    int notOutTimes;

    // Constructor to initialize player details
    CricketPlayer(int pid, String pname, int totalRuns, int inningsPlayed,
int notOutTimes) {
        this.pid = pid;
        this.pname = pname;
        this.totalRuns = totalRuns;
        this.inningsPlayed = inningsPlayed;
        this.notOutTimes = notOutTimes;
    }

    // Method to calculate the average of a player
    double calculateAverage() {
        if (inningsPlayed - notOutTimes == 0) {
            return 0; // Avoid division by zero
        }
        return (double) totalRuns / (inningsPlayed - notOutTimes);
    }

    // Method to display player details
```

```
    void displayDetails() {
        System.out.println("Player ID: " + pid);
        System.out.println("Player Name: " + pname);
        System.out.println("Total Runs: " + totalRuns);
        System.out.println("Innings Played: " + inningsPlayed);
        System.out.println("Not Out Times: " + notOutTimes);
    }
}
```

```
public class s7q2 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of players: ");
        int n = scanner.nextInt();

        CricketPlayer[] players = new CricketPlayer[n];

        // Accept player details
        for (int i = 0; i < n; i++) {
            System.out.println("Enter details of player " + (i + 1));
            System.out.print("Enter player ID: ");
            int pid = scanner.nextInt();
            System.out.print("Enter player name: ");
            String pname = scanner.next();
```

```
        System.out.print("Enter total runs: ");
        int totalRuns = scanner.nextInt();
        System.out.print("Enter innings played: ");
        int inningsPlayed = scanner.nextInt();
        System.out.print("Enter not out times: ");
        int notOutTimes = scanner.nextInt();

        players[i] = new CricketPlayer(pid, pname, totalRuns,
inningsPlayed, notOutTimes);
    }

    // Calculate and display average of all players
    double totalAverage = 0;
    for (int i = 0; i < n; i++) {
        double average = players[i].calculateAverage();
        totalAverage += average;


    }



    // Find and display player with maximum average
    CricketPlayer maxAveragePlayer = players[0];
    for (int i = 1; i < n; i++) {
        if (players[i].calculateAverage() >
maxAveragePlayer.calculateAverage()) {
```

```
            maxAveragePlayer = players[i];
        }
    }
    System.out.println("Player with maximum average:");
    maxAveragePlayer.displayDetails();
    System.out.println("Maximum average: " +
maxAveragePlayer.calculateAverage());
  }
}
```

```
class complexNumber:
  def __init__(self,real,imag):
    self.real=real
    self.imag=imag

  def __add__(self,other):
    if isinstance(other,complexNumber):
      realp=self.real+other.real
      imagp=self.imag+other.imag
      return complexNumver (realp,imagp)
    else:
      raise TypeError(" The operand must be an instance of
ComplexNumber")
  def __str__(self):
    return f"{self.real}+{self.imag}i"
```

```
c1=complexNumber(3,2)
c2=complexNumber(1,7)

result = c1+c2

print("First Complex Number:", c1)
print("Second Complex Number:", c2)
print("Sum:", result)
```

```
import tkinter as tk
import random
import string
deff Pass():
    length=12
    ch=string.ascii_letters
    password=''.join(random.choice(ch)for _ in range(length))
    pass_entry.delete(0, tk.END)
    pass_entry.insert(0, password)

root=tk.Tk()
root.title("password generator")

titlel=tk.Label(root,text="random passwprd generator",font=("Arial",12))
titlel.pack(pady=10)
```

```
pass_entry=tk.Entry(root, font=("Arial", 14), width=24)

pass_entry.pack(pady=20)


generate_button = tk.Button(root, text="Generate Password",
command=Pass, font=("Arial", 14))

generate_button.pack(pady=10)


root.mainloop()
```