

## Assignment-6.3

2303a51595

B-10

### TASK-1

#### Prompt :

Write a Java Student class with attributes name, roll number, and branch. Include constructor and displayDetails() method.

#### Code :

```
import java.util.Scanner;

class Student {
    String name;
    int rollNo;
    String branch;
    Student(String name, int rollNo, String branch) {
        this.name = name;
        this.rollNo = rollNo;
        this.branch = branch;
    }
    void displayDetails() {
        System.out.println("\n--- Student Details ---");
        System.out.println("Name: " + name);
        System.out.println("Roll No: " + rollNo);
        System.out.println("Branch: " + branch);
    }
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter Name: ");
        String name = sc.nextLine();
```

```
System.out.print("Enter Roll Number: ");  
int rollNo = sc.nextInt();  
sc.nextLine(); // clear buffer
```

```
System.out.print("Enter Branch: ");  
String branch = sc.nextLine();
```

```
Student s1 = new Student(name, rollNo, branch);
```

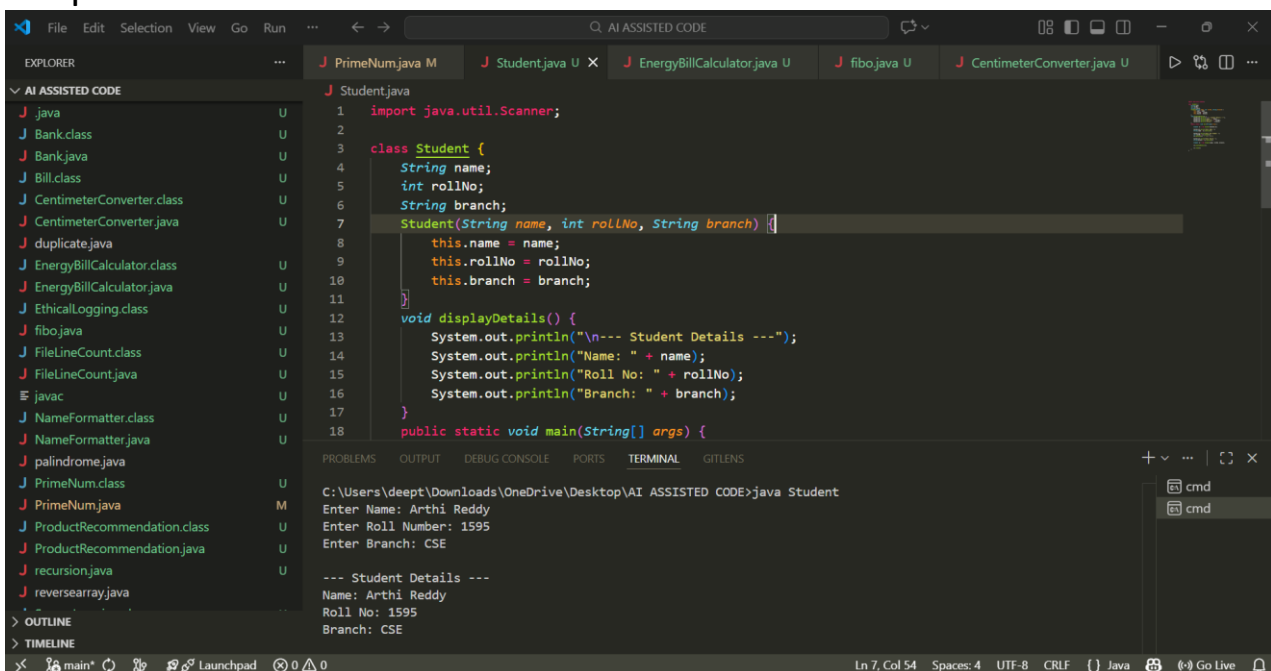
```
s1.displayDetails();
```

```
sc.close();
```

```
}
```

```
}
```

Output :



The screenshot shows an IDE with the following components:

- EXPLORER:** A list of files including java, Bank.class, Bank.java, Bill.class, CentimeterConverter.class, CentimeterConverter.java, duplicate.java, EnergyBillCalculator.class, EnergyBillCalculator.java, EthicalLogging.class, fibo.java, FileLineCount.class, FileLineCount.java, javac, NameFormatter.class, NameFormatter.java, palindrome.java, PrimeNum.class, PrimeNum.java, ProductRecommendation.class, ProductRecommendation.java, recursion.java, and reversearray.java.
- EDITOR:** The file Student.java is open, showing the following code:

```
1 import java.util.Scanner;  
2  
3 class Student {  
4     String name;  
5     int rollNo;  
6     String branch;  
7     Student(String name, int rollNo, String branch) {  
8         this.name = name;  
9         this.rollNo = rollNo;  
10        this.branch = branch;  
11    }  
12    void displayDetails() {  
13        System.out.println("\n--- Student Details ---");  
14        System.out.println("Name: " + name);  
15        System.out.println("Roll No: " + rollNo);  
16        System.out.println("Branch: " + branch);  
17    }  
18    public static void main(String[] args) {
```
- TERMINAL:** The output of the program is displayed:

```
C:\Users\deepthi\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java Student  
Enter Name: Arthi Reddy  
Enter Roll Number: 1595  
Enter Branch: CSE  
  
--- Student Details ---  
Name: Arthi Reddy  
Roll No: 1595  
Branch: CSE
```

## Explanation :

This program uses a Student class with constructor and methods to store and display student details.

It demonstrates object creation and basic Object-Oriented Programming concepts.

## TASK-2

### Prompt :

Write a Java program to take a number from the user and print its first 10 multiples.

Use both for loop and while loop to show the result.

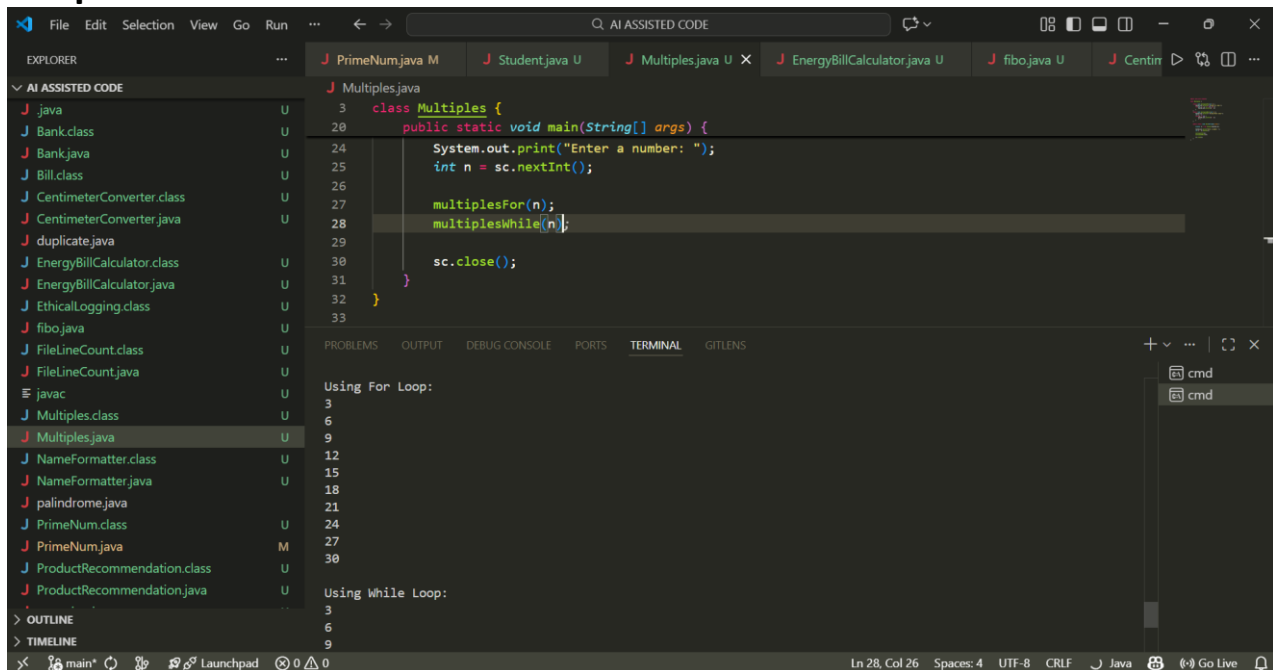
### Code :

```
import java.util.Scanner;
```

```
class Multiples {
```

```
    static void multiplesFor(int n) {  
        System.out.println("\nUsing For Loop:");  
        for(int i = 1; i <= 10; i++) {  
            System.out.println(n * i);  
        }  
    }  
    static void multiplesWhile(int n) {  
        System.out.println("\nUsing While Loop:");  
        int i = 1;  
        while(i <= 10) {  
            System.out.println(n * i);  
            i++;  
        }  
    }  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Enter a number: ");  
        int n = sc.nextInt();  
  
        multiplesFor(n);  
        multiplesWhile(n);  
  
        sc.close();  
    }
```

## Output :



```
class Multiples {  
    public static void main(String[] args) {  
        System.out.print("Enter a number: ");  
        int n = sc.nextInt();  
  
        multiplesFor(n);  
        multiplesWhile(n);  
  
        sc.close();  
    }  
}
```

Using For Loop:

3  
6  
9  
12  
15  
18  
21  
24  
27  
30

Using While Loop:

3  
6  
9

## Explanation :

This program takes a number from the user and prints its first 10 multiples using both for and while loops

## TASK-3

### Prompt :

Write a Java program to take age as input from the user and classify the person as child, teenager, adult, or senior using conditional statements.

### Code :

```
import java.util.Scanner;
```

```
class AgeClassification {
```

```
    static String classifyAge(int age) {
```

```
        if(age < 13)
```

```
            return "Child";
```

```
        else if(age < 20)
```

```
            return "Teenager";
```

```

    else if(age < 60)
        return "Adult";
    else
        return "Senior";
}

```

```

public static void main(String[] args) {

```

```

    Scanner sc = new Scanner(System.in);

```

```

    System.out.print("Enter age: ");
    int age = sc.nextInt();

```

```

    String result = classifyAge(age);

```

```

    System.out.println("Category: " + result);

```

```

    sc.close();

```

```

}

```

```

}

```

## Output :

The screenshot shows an IDE with the following components:

- EXPLORER:** A list of files on the left, including `AgeClassification.java`, `Bank.class`, `Bank.java`, `Bill.class`, `CentimeterConverter.class`, `CentimeterConverter.java`, `duplicate.java`, `EnergyBillCalculator.class`, `EnergyBillCalculator.java`, `EthicalLogging.class`, `fibonacci.java`, `FileLineCount.class`, `FileLineCount.java`, `javac`, `Multiples.class`, `Multiples.java`, `NameFormatter.class`, `NameFormatter.java`, `palindrome.java`, `PrimeNum.class`, and `PrimeNum.java`.
- EDITOR:** The main window showing the code for `AgeClassification.java`. The code is as follows:
 

```

1  import java.util.Scanner;
2
3  class AgeClassification {
4
5      static String classifyAge(int age) {
6
7          if(age < 13)
8              return "Child";
9          else if(age < 20)
10             return "Teenager";
11          else if(age < 60)
12             return "Adult";
13          else
14             return "Senior";
15      }
16
17      public static void main(String[] args) {
18

```
- TERMINAL:** The bottom panel showing the execution of the program. The commands and output are:
 

```

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>javac AgeClassification.java
C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java AgeClassification
Enter age: 19
Category: Teenager

```

**Explanation :**

This program takes age as input and checks conditions using if-else statements to decide the age group.

It demonstrates decision making and conditional logic in Java.

**TASK-4****Prompt :**

Write a Java program to take a number n from the user and calculate the sum of the first n natural numbers.

**Code :**

```
import java.util.Scanner;
```

```
class SumNumbers {  
    static int sumFor(int n) {  
        int sum = 0;  
        for(int i = 1; i <= n; i++) {  
            sum += i;  
        }  
        return sum;  
    }  
    static int sumWhile(int n) {  
        int sum = 0, i = 1;  
        while(i <= n) {  
            sum += i;  
            i++;  
        }  
        return sum;  
    }  
    static int sumFormula(int n) {  
        return n * (n + 1) / 2;  
    }  
}
```

```
public static void main(String[] args) {
```

```
    Scanner sc = new Scanner(System.in);
```

```
System.out.print("Enter n: ");  
int n = sc.nextInt();
```

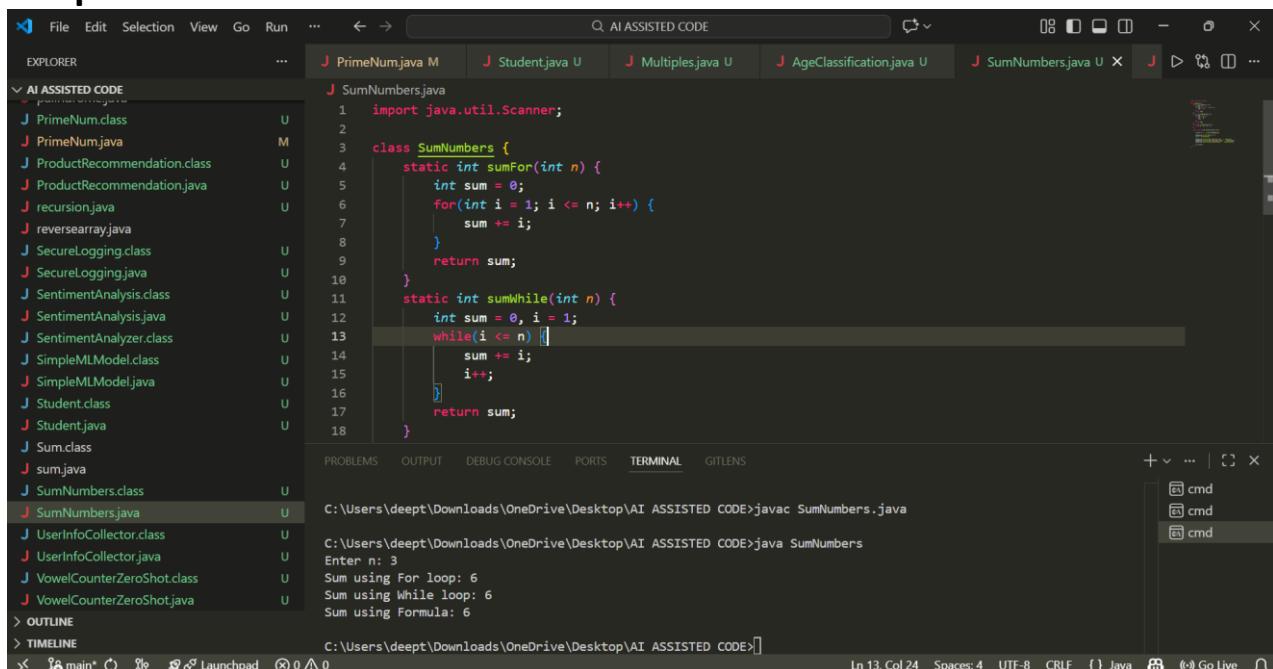
```
System.out.println("Sum using For loop: " + sumFor(n));  
System.out.println("Sum using While loop: " + sumWhile(n));  
System.out.println("Sum using Formula: " + sumFormula(n));
```

```
sc.close();
```

```
}
```

```
}
```

## Output :



The screenshot shows an IDE with the following components:

- EXPLORER:** A list of files on the left, including `SumNumbers.java` which is currently selected.
- EDITOR:** The main window displays the code for `SumNumbers.java`. It includes imports for `java.util.Scanner` and defines a class `SumNumbers` with three static methods: `sumFor` (using a for loop), `sumWhile` (using a while loop), and `sumFormula` (using a mathematical formula).
- TERMINAL:** The bottom panel shows the execution of the program. It displays the command `javac SumNumbers.java` followed by `java SumNumbers`. The output shows the prompt "Enter n: 3" and the results: "Sum using For loop: 6", "Sum using While loop: 6", and "Sum using Formula: 6".

## Explanation :

This program calculates the sum of first n numbers using loops and a mathematical formula.

## Task-5

### Prompt :

Write a Java program to create a BankAccount class with deposit, withdraw, and check balance methods.

Code :

```
import java.util.Scanner;
```

```
class BankAccount {
```

```
    String name;  
    double balance;
```

```
    // Constructor
```

```
    BankAccount(String name, double balance) {  
        this.name = name;  
        this.balance = balance;  
    }
```

```
    // Deposit money
```

```
    void deposit(double amount) {  
        balance += amount;  
        System.out.println("Amount Deposited: " + amount);  
    }
```

```
    // Withdraw money
```

```
    void withdraw(double amount) {  
        if(amount <= balance) {  
            balance -= amount;  
            System.out.println("Amount Withdrawn: " + amount);  
        } else {  
            System.out.println("Insufficient Balance");  
        }  
    }
```

```
    // Check balance
```

```
    void checkBalance() {  
        System.out.println("Current Balance: " + balance);  
    }  
    public static void main(String[] args) {
```



```
Scanner sc = new Scanner(System.in);
```

```
System.out.print("Enter Account Holder Name: ");
```

```
String name = sc.nextLine();
```

```
BankAccount acc = new BankAccount(name, balance);
```

```
System.out.print("Enter deposit amount: ");
```

```
acc.deposit(sc.nextDouble());
```

```
System.out.print("Enter withdraw amount: ");
```

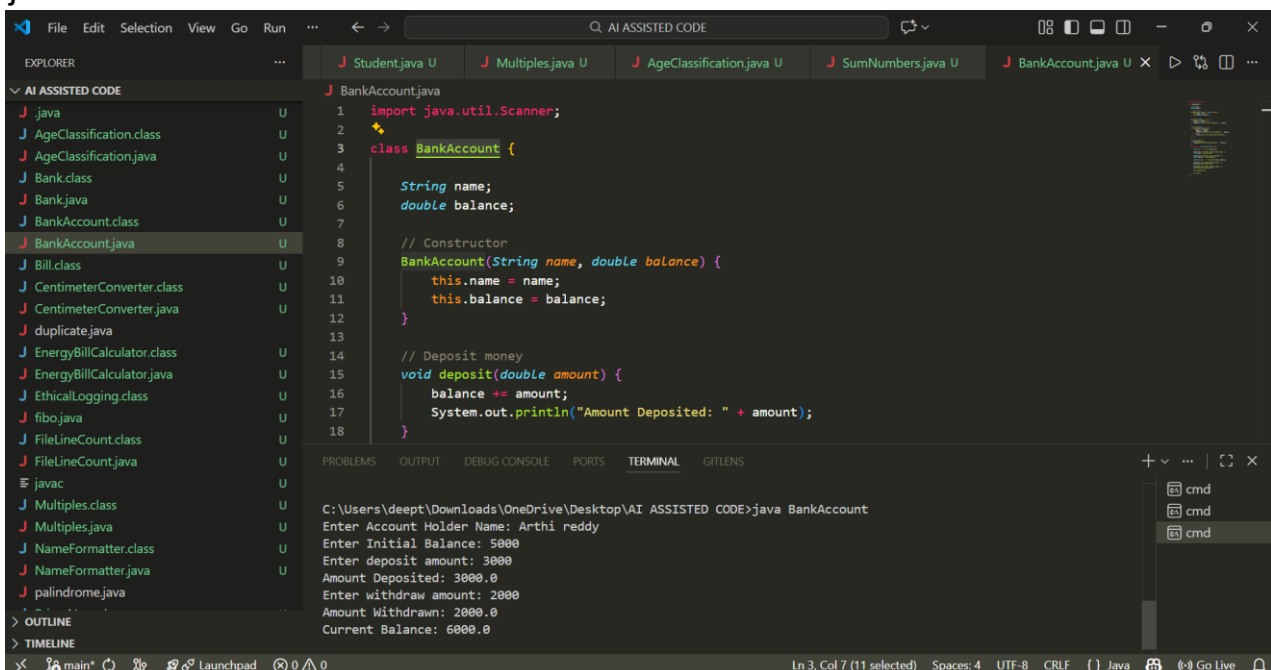
```
acc.withdraw(sc.nextDouble());
```

```
acc.checkBalance();
```

```
sc.close();
```

```
}
```

```
}
```



The screenshot shows an IDE with the following components:

- EXPLORER:** A list of files including `BankAccount.java`, which is currently selected.
- EDITOR:** Displays the code for `BankAccount.java`. The code includes an import statement for `Scanner`, a class definition with attributes `String name` and `double balance`, a constructor `BankAccount(String name, double balance)`, and two methods: `deposit(double amount)` and `withdraw(double amount)`.
- TERMINAL:** Shows the output of running the program. The output is as follows:

```
C:\Users\deep\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java BankAccount
Enter Account Holder Name: Arthi reddy
Enter Initial Balance: 5000
Enter deposit amount: 3000
Amount Deposited: 3000.0
Enter withdraw amount: 2000
Amount Withdrawn: 2000.0
Current Balance: 6000.0
```

## Explanation :

This program creates a `BankAccount` class to manage money transactions like deposit, withdraw, and balance checking.

