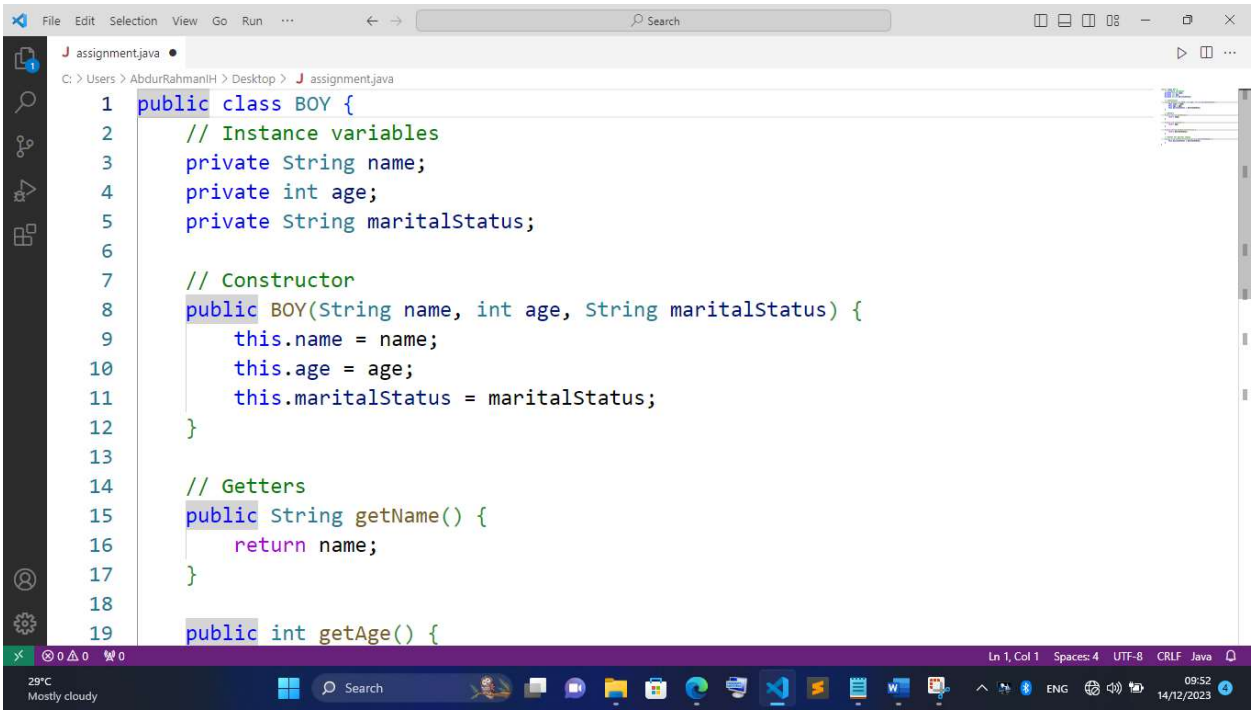
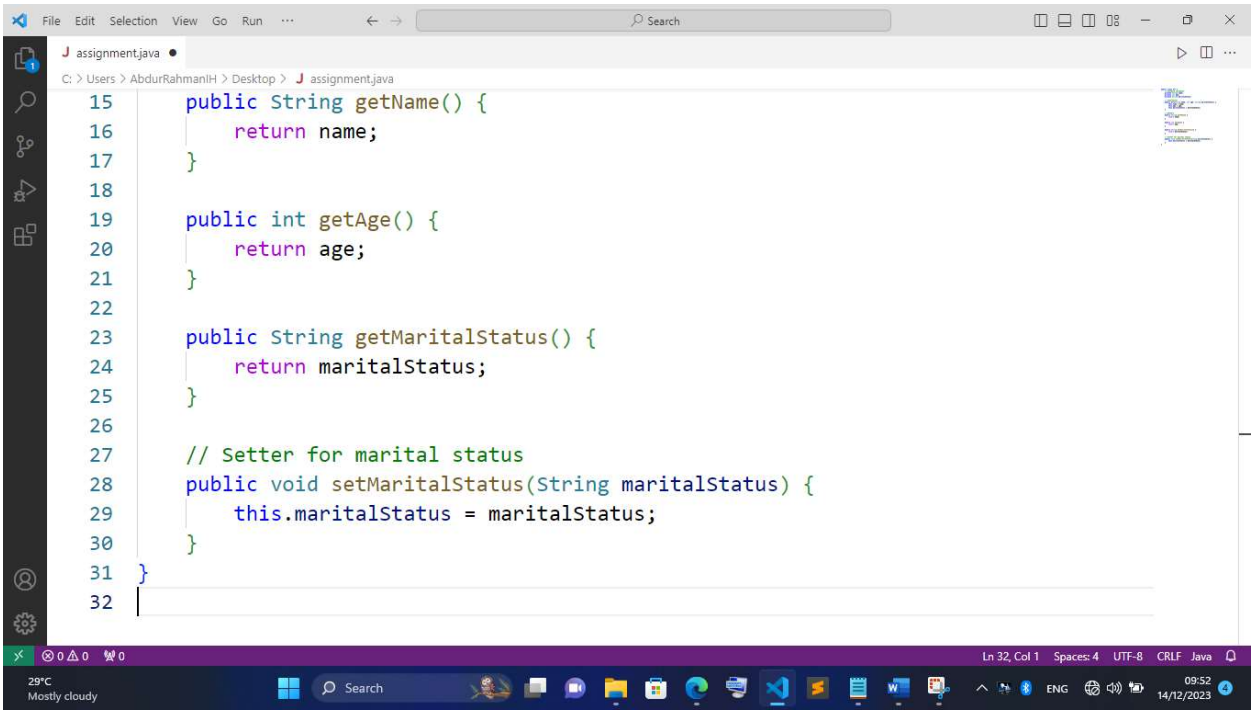
1. Create a class 'BOY' that has the following details:

Three instance variables: name age, marital status, A constructor that initialize the three instance variables

A getter for all the instance variables

A setter for the marital status

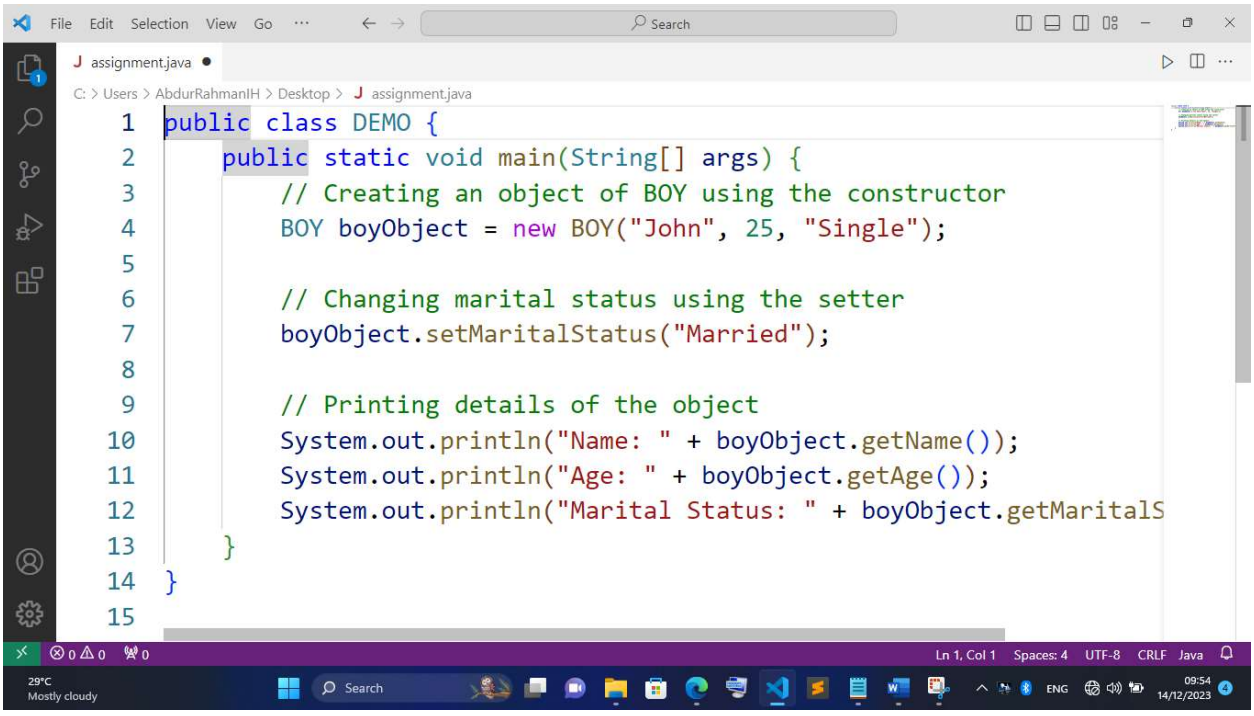




```
15      public String getName() {
16          return name;
17      }
18
19      public int getAge() {
20          return age;
21      }
22
23      public String getMaritalStatus() {
24          return maritalStatus;
25      }
26
27      // Setter for marital status
28      public void setMaritalStatus(String maritalStatus) {
29          this.maritalStatus = maritalStatus;
30      }
31  }
32
```

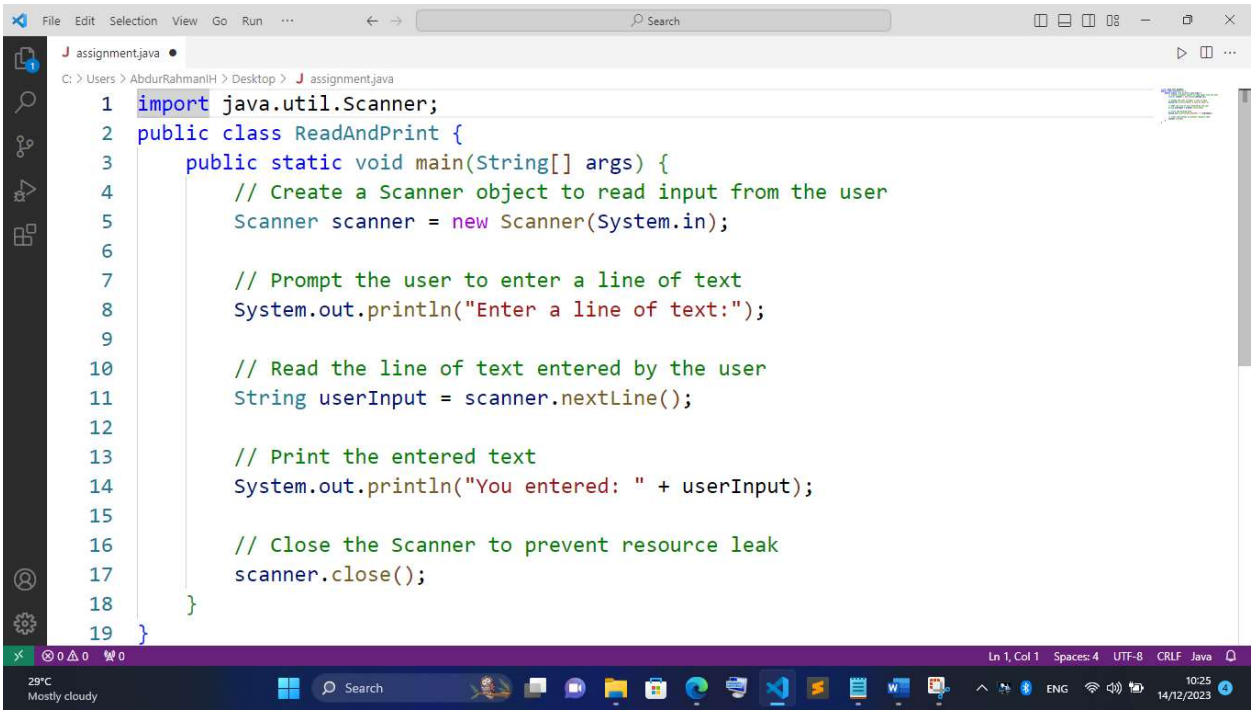
Define a DEMO class that create an 'object' of BOY using the constructor define. Change the

marital status of the object you created from single to married. Print the details of the object.



```
1   public class DEMO {
2       public static void main(String[] args) {
3           // Creating an object of BOY using the constructor
4           BOY boyObject = new BOY("John", 25, "Single");
5
6           // Changing marital status using the setter
7           boyObject.setMaritalStatus("Married");
8
9           // Printing details of the object
10          System.out.println("Name: " + boyObject.getName());
11          System.out.println("Age: " + boyObject.getAge());
12          System.out.println("Marital Status: " + boyObject.getMaritalS
13      }
14  }
15
```

Write a program Read a line of text from the user using Scanner and print it.



```java
import java.util.Scanner;
public class ReadAndPrint {
    public static void main(String[] args) {
        // Create a Scanner object to read input from the user
        Scanner scanner = new Scanner(System.in);

        // Prompt the user to enter a line of text
        System.out.println("Enter a line of text:");

        // Read the line of text entered by the user
        String userInput = scanner.nextLine();

        // Print the entered text
        System.out.println("You entered: " + userInput);

        // Close the Scanner to prevent resource leak
        scanner.close();
    }
}
```

Question 2

2. a. Consider the following Java program. Trace through the code and answer the questions that follows.

public class Tracing

public static void main(Stringl args) {

int x = 5;

int y = 10;

System.out.printin("Before methodCall: x = " +x+"y = "+ y);

methodCall(x, y):

System.out,.printin("After methodCall: x ="+x+"y="+y);

public static void method Call(int a, int b) {

a t=5;

b *=2:

System.out.printin("Inside method Call: a = "+a+", b = "+ b);

i. What are the initial values of x and y before the method call?

```
Before methodCall: x = 5, y = 10
```

ii. What values are passed to the method Call method for parameters a and b?

```
Inside method Call: a = 10, b = 20
```

i. Inside the methodCall method, what are the values of a and b after the operations?

```
After methodCall: x = 5, y = 10
```

iv. After the methodCall method has been executed, what are the values of x and y in the main method?

X = 5 and y = 10

V. Provide the final output that will be printed to the console.

```
Before methodCall: x = 5, y = 10
Inside method Call: a = 10, b = 20
After methodCall: x = 5, y = 10
```

Question 3

3 . Trace the output of each of the following programs.

a. public class Computation{

public static void main(String [] args){

int x = 3, y = 5, z = 15*y/x+35%3;

```
            System.out.println("Z = " +z);
            }
        }
B public static void main(Stringl args) {
int x =3, y=10, z
z= 15"y/x +35%3;
System.outprintin('z =+z);
public class GreetingWorld (
public static void main(Stringl args) {
String s1 = "Distace learning Centre ";

String s2 = "Computer Science";
String s3 = "DISTANCE LEARNING CENTRE ";

System. out printin(s1.equals(s3));
System. out printin(s2.compareTo(s3));
System. out printin(s1.compareTolgnoreCase(s3));
System. outprintin(s1.indexOf('e');
System.out printin(s2.substring(9));
System.outprintn(s3.charAt(2));
```

Q3 output

a. Z = 50

b. -13

3

Learning

a

Question 4

Difference between a constructor and a method:

A constructor is a special method used to initialize the state of an object when it's created.

It has the same name as the class and does not have a return type.

It is automatically called when an object is instantiated.

A method, on the other hand, is a block of code that performs a specific task.

It can be called multiple times and may or may not return a value.

Question 5: 5. Write a java program to SUM the elements in an aTay. Assume any aray of te

n (10) integer values.

ASSUME: Int] array={87, 68, 94, 100, 83, 78, 85, 91, 76, 87 }

```
public class ArraySum {
   public static void main(String[] args) {
      int[] array = {87, 68, 94, 100, 83, 78, 85, 91, 76, 87};

      int sum = 0;
      for (int num : array) {
         sum += num;
      }
      System.out.println("Sum of array elements: " + sum);
```

```
    }
}
```

Question 6: Java program for Instructor class:

Write a JAVA program that create an instructor. The instructor class should ha

Four instance variables name, rank, courseTitle and department as instance variables. Two constructors to initialize the instance variables. The first constructor will have four parameters to represent the instance variables. The second constructor should have three paameters with courseTitle set to it default value. Use the this' keyword to call the first constructor from the second constructor.

Create getters for all instance variables as well as setters for the appropriate

e variable(s).

Define a TEST class that will create THREE instructors and assign a course to

each of the instructors. The title of the course for the first instructor is "0bject

Oriented Programming I" and that of the second instructor is "Introduction to programming Languages". The third instructor should have no course title (meaning you use the constructor with three paramaters). Test all the methods and use the setters to change the COURSETITLE for the two instructors to "Compiler Construction and "Finite Automata. (NOTE: Use SCANNER to get the values for name, rank courseTitle and department for the first instructor only. You can assign values for the other instructors]

```
import java.util.Scanner;


public class Instructor {
    private String name;
    private int rank;
    private String courseTitle;
    private String department;
```

```
    // Constructor with four parameters
    public Instructor(String name, int rank, String courseTitle, String department) {
        this.name = name;
        this.rank = rank;
        this.courseTitle = courseTitle;
        this.department = department;
    }

    // Second constructor with three parameters, calling the first constructor
    public Instructor(String name, int rank, String department) {
        this(name, rank, "Default Course", department);
    }

    // Getters and Setters for all instance variables
    // ...

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

        // Getting values for the first instructor
```

```
        System.out.println("Enter name for the first instructor: ");
        String name = scanner.nextLine();

        System.out.println("Enter rank for the first instructor: ");
        int rank = scanner.nextInt();
        scanner.nextLine(); // Consume the newline character

        System.out.println("Enter department for the first instructor: ");
        String department = scanner.nextLine();

        // Creating the first instructor object
        Instructor instructor1 = new Instructor(name, rank, department);

        // Setting course title for the first instructor
        instructor1.setCourseTitle("Object Oriented Programming I");

        // Creating the second instructor object
        Instructor instructor2 = new Instructor("Instructor2", 2,
"Computer Science");

        // Setting course title for the second instructor
        instructor2.setCourseTitle("Introduction to Programming
Languages");
```

```
        // Creating the third instructor object
        Instructor instructor3 = new Instructor("Instructor3", 3,
"Mathematics");

        // Testing methods and setters
        // ...

        // Changing COURSETITLE for the two instructors
        instructor1.setCourseTitle("Compiler Construction");
        instructor2.setCourseTitle("Finite Automata");

        scanner.close();
    }
}
```