Course Name:	**Object Oriented Programming**	**Semester:**	**III**
Date of Performance:	**28 / 07 / 2025**	**Batch No:**	**Batch A1**
Faculty Name:	**Mrs. Amrita Naiksatam**	**Roll No:**	**20**
Faculty Sign & Date:		**Grade/Marks:**	**___/15**

Experiment No: 2
Title: First Java code use of Class and Object

Aim and Objective of the Experiment:
Learn the concept of encapsulation in Java

COs to be achieved:
CO1: Understand concepts of Object Oriented Programming and basic characteristics of Java.

Tools used:
Notepad, Command Prompt

Theory:
(About all modifier used while declaring methods) (constructors vs methods)

Code:

1. Write a program to perform addition, subtraction, multiplication and division of two complex numbers.
   - Variation 1: Implementation with One class only
   - Variation 2: Implementation with Two classes (One will be main class and Another will be class for complex) without constructors
   - Variation 3: variation2 with use of constructors (But without this keyword)
   - Variation 4: variation2 with use of constructors (But with use of this keyword)
   - Variation 5: Add one more class which performs following mathematical operation on complex number.
     - Inverse
     - Complex conjugate
     - Polar form

NOTE: use the concept of parameterized method, non-parameterized method, different constructors (passing all arguments, passing few arguments, passing object and without passing arguments)

Program:

Variation 1 –

```
import java.util.Scanner;

public class var1
{
	public static void main(String[] args)
	{
		Scanner sc = new Scanner(System.in);

		System.out.print("Enter real and imaginary parts of first complex number: ");
		double r1 = sc.nextDouble();
		double i1 = sc.nextDouble();

		System.out.print("Enter real and imaginary parts of second complex number: ");
		double r2 = sc.nextDouble();
		double i2 = sc.nextDouble();

		System.out.println("Addition: " + (r1 + r2) + " + " + (i1 + i2) + "i");
		System.out.println("Subtraction: " + (r1 - r2) + " + " + (i1 - i2) + "i");
		System.out.println("Multiplication: " + (r1 * r2 - i1 * i2) + " + " + (r1 * i2 + i1 * r2) + "i");

		double denom = r2 * r2 + i2 * i2;
		System.out.println("Division: " + ((r1 * r2 + i1 * i2) / denom) + " + " + ((i1 * r2 - r1 * i2) / denom) + "i");


		sc.close();
	}
}
```

Variation 2 –

```
import java.util.Scanner;

class Complex
{
	double r, i;

	void vals(double real, double imag)
	{
		r = real;
		i = imag;
	}

	void add(Complex c)
	{
		System.out.println("Addition: " + (r + c.r) + " + " + (i + c.i) +
"i");
	}

	void subtract(Complex c)
	{
		System.out.println("Subtraction: " + (r - c.r) + " + " + (i - c.i)
+ "i");
	}

	void multiply(Complex c)
	{
		System.out.println("Multiplication: " + (r * c.r - i * c.i) + " + "
+ (r * c.i + i * c.r) + "i");
	}

	void divide(Complex c)
	{
		double denom = c.r * c.r + c.i * c.i;
		System.out.println("Division: " + ((r * c.r + i * c.i) / denom) + "
+ " + ((i * c.r - r * c.i) / denom) + "i");
	}
}
```

```
public class var2
{
	public static void main(String args[])
	{
		Scanner sc = new Scanner(System.in);

		Complex c1 = new Complex();
		Complex c2 = new Complex();

		System.out.print("Enter real and imaginary parts of first complex
number: ");
		double r1 = sc.nextDouble();
		double i1 = sc.nextDouble();
		c1.vals(r1,i1);

		System.out.print("Enter real and imaginary parts of second complex
number: ");
		double r2 = sc.nextDouble();
		double i2 = sc.nextDouble();
		c2.vals(r2,i2);

		c1.add(c2);
		c1.subtract(c2);
		c1.multiply(c2);
		c1.divide(c2);


		sc.close();
	}
}
```

Variation 3 –

```
import java.util.Scanner;

class Complex
{
	double r, i;

	Complex(double real, double imag)
	{
		r = real;
		i = imag;
	}

	void add(Complex c)
	{
		System.out.println("Addition: " + (r + c.r) + " + " + (i + c.i) +
"i");
	}

	void subtract(Complex c)
	{
		System.out.println("Subtraction: " + (r - c.r) + " + " + (i -
c.i) + "i");
	}

	void multiply(Complex c)
	{
		System.out.println("Multiplication: " + (r * c.r - i * c.i) + " + "
+ (r * c.i + i * c.r) + "i");
	}

	void divide(Complex c)
	{
		double denom = c.r * c.r + c.i * c.i;
		System.out.println("Division: " + ((r * c.r + i * c.i) / denom) + "
+ " + ((i * c.r - r * c.i) / denom) + "i");
	}
}
```

```
public class var3
{
	public static void main(String args[])
	{
		Scanner sc = new Scanner(System.in);

		System.out.print("Enter real and imaginary parts of first complex
number: ");
		double r1 = sc.nextDouble();
		double i1 = sc.nextDouble();
		Complex c1 = new Complex(r1, i1);

		System.out.print("Enter real and imaginary parts of second complex
number: ");
		double r2 = sc.nextDouble();
		double i2 = sc.nextDouble();
		Complex c2 = new Complex(r2, i2);

		c1.add(c2);
		c1.subtract(c2);
		c1.multiply(c2);
		c1.divide(c2);

		sc.close();
	}
}
```

Variation 4 –

```
import java.util.Scanner;

class Complex
{
      double r, i;

      Complex(double r, double i)
      {
            this.r = r;
            this.i = i;
      }

      void add(Complex c)
      {
              System.out.println("Addition: " + (r + c.r) + " + " + (i + c.i) +
"i");
      }

      void subtract(Complex c)
      {
              System.out.println("Subtraction: " + (r - c.r) + " + " + (i -
c.i) + "i");
      }

      void multiply(Complex c)
      {
            System.out.println("Multiplication: " + (r * c.r - i * c.i) + " + "
+ (r * c.i + i * c.r) + "i");
      }

      void divide(Complex c)
      {
            double denom = c.r * c.r + c.i * c.i;
            System.out.println("Division: " + ((r * c.r + i * c.i) / denom) + "
+ " + ((i * c.r - r * c.i) / denom) + "i");
      }
}
```

```
public class var4
{
	public static void main(String args[])
	{
		Scanner sc = new Scanner(System.in);

		System.out.print("Enter real and imaginary parts of first complex
number: ");
		double r1 = sc.nextDouble();
		double i1 = sc.nextDouble();
		Complex c1 = new Complex(r1, i1);

		System.out.print("Enter real and imaginary parts of second complex
number: ");
		double r2 = sc.nextDouble();
		double i2 = sc.nextDouble();
		Complex c2 = new Complex(r2, i2);

		c1.add(c2);
		c1.subtract(c2);
		c1.multiply(c2);
		c1.divide(c2);

		sc.close();
	}
}
```

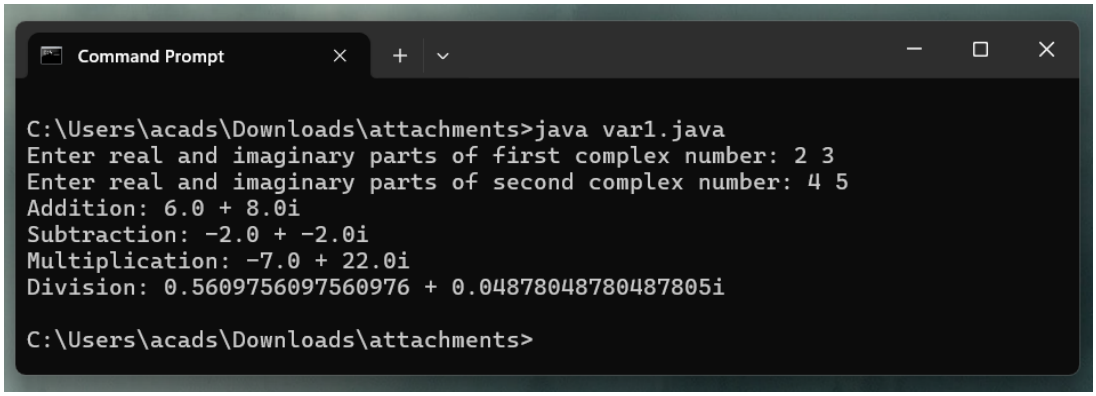
Variation 5 –

```
import java.util.Scanner;

class Complex
{
	double r, i;

	Complex(double r, double i)
	{
		this.r = r;
		this.i = i;
	}

	void add(Complex c)
	{
		 System.out.println("Addition: " + (r + c.r) + " + " + (i + c.i) + "i");
	}

	void subtract(Complex c)
	{
		 System.out.println("Subtraction: " + (r - c.r) + " + " + (i - c.i) +
"i");
	}

	void multiply(Complex c)
	{
		System.out.println("Multiplication: " + (r * c.r - i * c.i) + " + " + (r *
c.i + i * c.r) + "i");
	}

	void divide(Complex c)
	{
		double denom = c.r * c.r + c.i * c.i;
		System.out.println("Division: " + ((r * c.r + i * c.i) / denom) + " + " +
((i * c.r - r * c.i) / denom) + "i");
	}
}

class advComplex
{
	void inverse(Complex c)
	{
		double denom = c.r * c.r + c.i * c.i;
		System.out.println("Inverse: " + (c.r / denom) + " - " + (c.i / denom) +
"i");
```

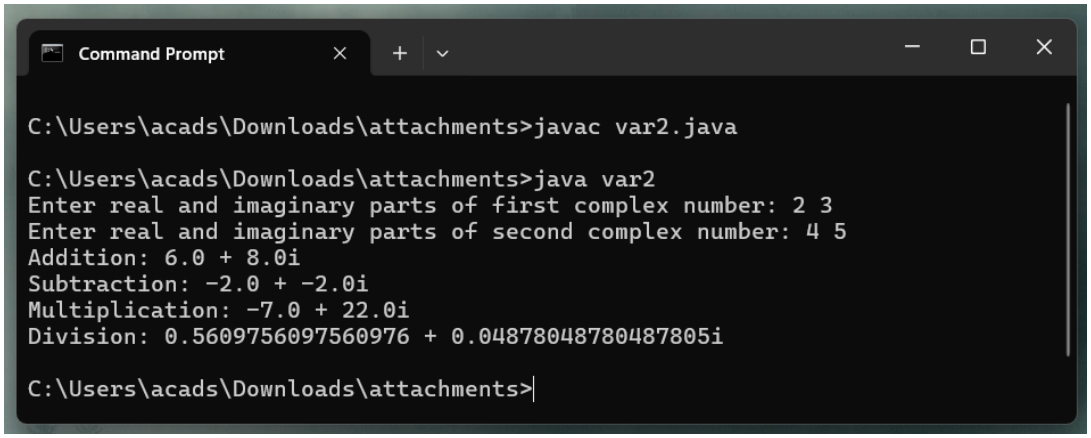
```
	}

	void conjugate(Complex c)
	{
		System.out.println("Conjugate: " + c.r + " - " + c.i + "i");
	}

	void polarForm(Complex c)
	{
		double magnitude = Math.sqrt(c.r * c.r + c.i * c.i);
		double angle = Math.atan2(c.i, c.r);
		System.out.printf("Polar Form: %.2f * (cos(%.2f) + i*sin(%.2f))\n",
magnitude, angle, angle);
	}
}

public class var5
{
	public static void main(String args[])
	{
		Scanner sc = new Scanner(System.in);

		System.out.print("Enter real and imaginary parts of first complex number:
");
		double r1 = sc.nextDouble();
		double i1 = sc.nextDouble();
		Complex c1 = new Complex(r1, i1);

		System.out.print("Enter real and imaginary parts of second complex number:
");
		double r2 = sc.nextDouble();
		double i2 = sc.nextDouble();
		Complex c2 = new Complex(r2, i2);

		c1.add(c2);
		c1.subtract(c2);
		c1.multiply(c2);
		c1.divide(c2);

		advComplex adv = new advComplex();
		adv.inverse(c1);
		adv.conjugate(c1);
		adv.polarForm(c1);

		sc.close();
	}
}
```

Output:

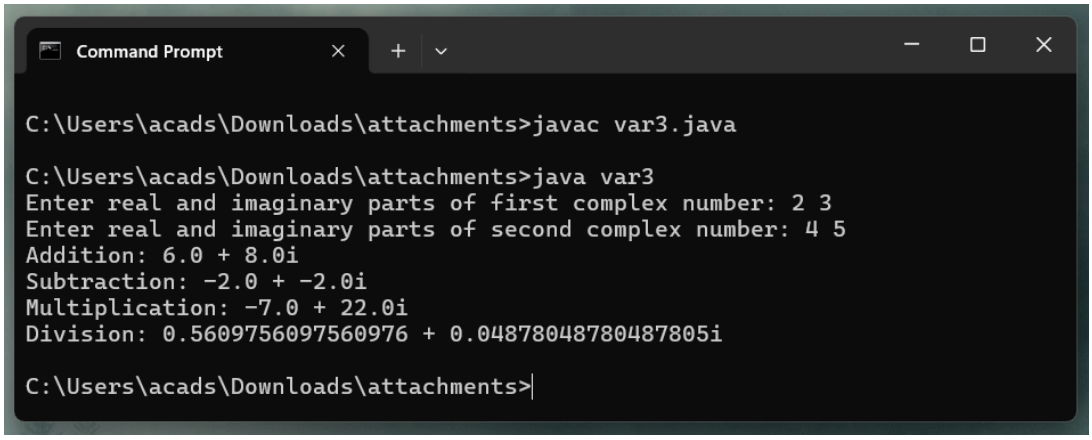
Variation 1 –



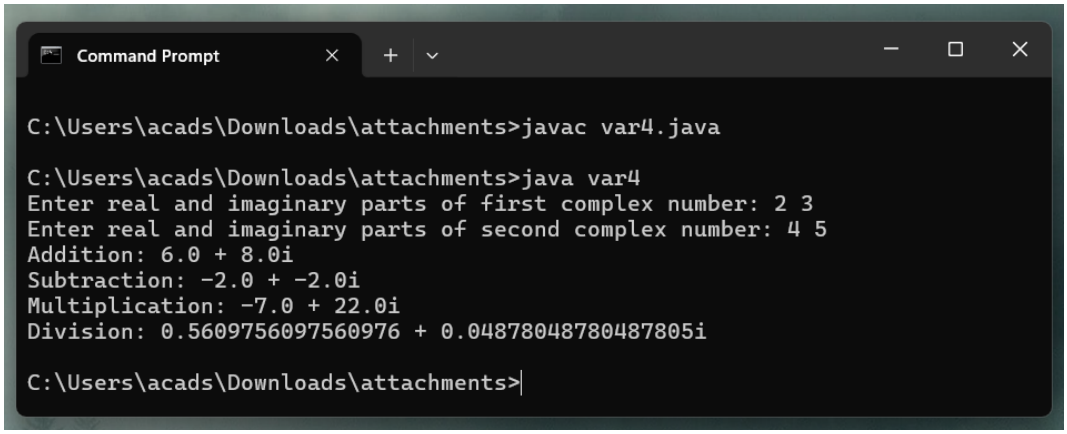
Variation 2 –



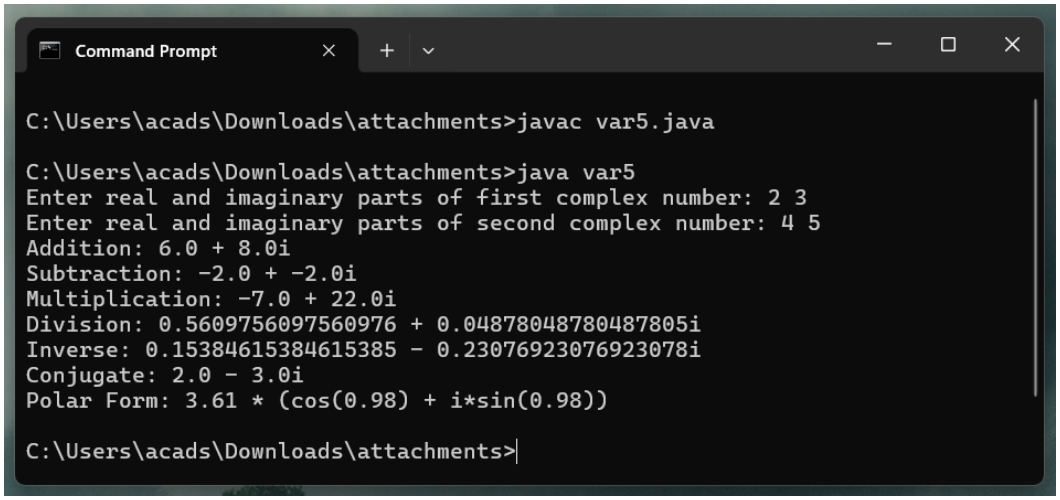
Variation 3 –



Variation 4 –



Variation 5 –



Post Lab Subjective/Objective type Questions:

1. Understand the following code and rewrite the code as per inform in following question.

```
class SalesTaxCalculator {
        float amount=100.0f; // instance variable
        float taxRate=10.2f; //instance variable
        void calculateTax() {
        float taxAmt = amount*taxRate/100;
        System.out.println(taxAmt);   }
        public static void main (String args[ ])        {
        SalesTaxCalculator obj1 = new SalesTaxCalculator();
        SalesTaxCalculator obj2 = new SalesTaxCalculator();
    System.out.println("Amount in Object 1: "+ obj1.amount);
    System.out.println("Tax Rate in Object 1: "+ obj1.taxRate);
    System.out.println("Amount in Object 2: "+ obj2.amount);
    System.out.println("Tax Rate in Object 2: "+ obj2.taxRate);
    }}
```

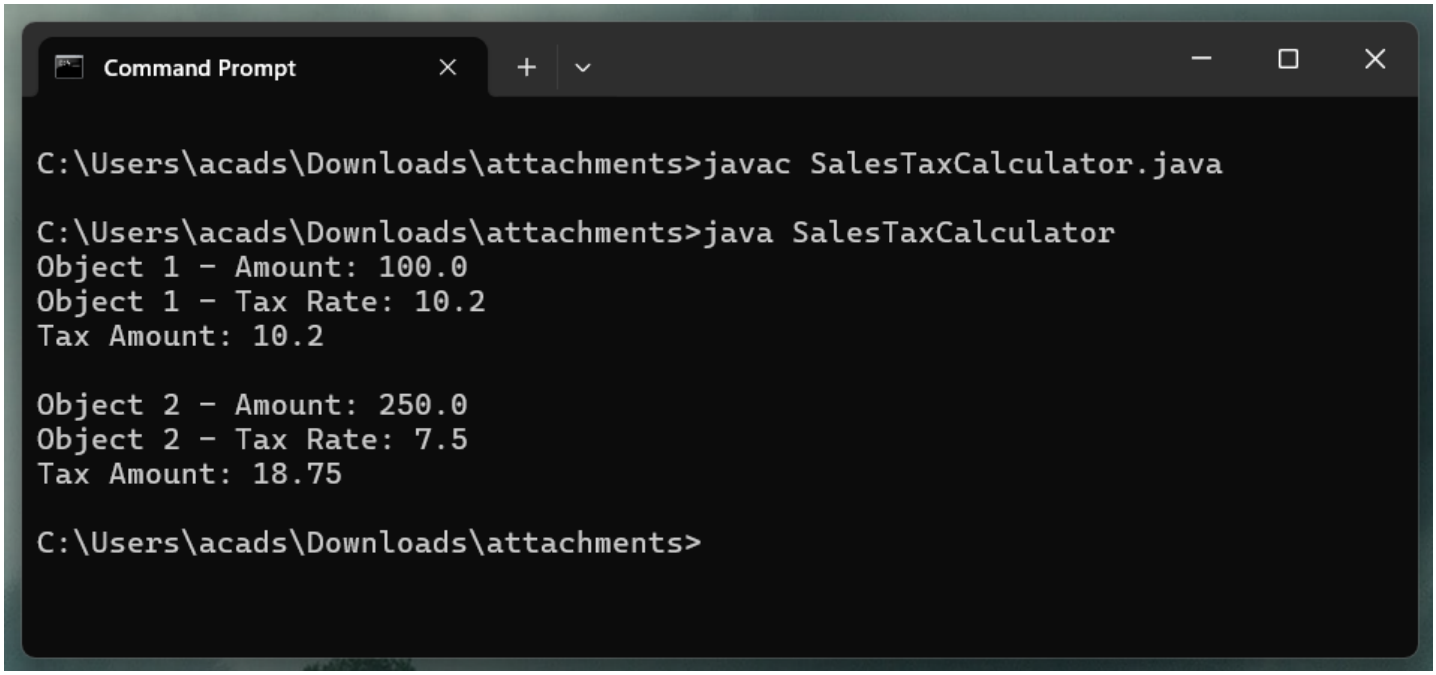
a) Modify to accept instance variable using a constructor with no argument and execute it.
b) Overload the constructor in part (a) and execute it.
c) Make the use of ***this*** keyword in part (b) and show its usages.

Ans:
Program-

```
class SalesTaxCalculator
{
	float amount;
	float taxRate;

	SalesTaxCalculator()
	{
		amount = 100.0f;
		taxRate = 10.2f;
	}

	SalesTaxCalculator(float amount, float taxRate)
	{
		this.amount = amount;
		this.taxRate = taxRate;
	}

	void calculateTax()
	{
		float taxAmt = amount * taxRate / 100;
		System.out.println("Tax Amount: " + taxAmt);
	}
```

```
      public static void main(String[] args)
      {
            SalesTaxCalculator obj1 = new SalesTaxCalculator();
            System.out.println("Object 1 - Amount: " + obj1.amount);
            System.out.println("Object 1 - Tax Rate: " + obj1.taxRate);
            obj1.calculateTax();

              System.out.println();

              SalesTaxCalculator obj2 = new SalesTaxCalculator(250.0f, 7.5f);
              System.out.println("Object 2 - Amount: " + obj2.amount);
              System.out.println("Object 2 - Tax Rate: " + obj2.taxRate);
              obj2.calculateTax();
      }
}
```

Output-



2. Explain the difference between instance variables and class variables.
Ans:

1. **Instance Variable:**
   Belongs to an object. Each object has its own copy. Defined without static.
2. **Class Variable:**
   Belongs to the class. Shared among all objects. Defined with static.

3. Explain the different types of constructors.
Ans:

2. **Default Constructor**:
   Provided by Java if no constructor is defined.
3. **No-Argument Constructor**:
   Defined by the programmer, takes no parameters.
4. **Parameterized Constructor**:
   Takes parameters to set custom values during object creation.
5. **Constructor Overloading**:
   Multiple constructors with different parameter lists in the same class.

Conclusion:

This experiment helped understand the use of classes, objects, and constructors in Java. It demonstrated operations on complex numbers and showed how constructor overloading and the this keyword support object initialization and encapsulation.

Signature of faculty in-charge with Date: