Write a program that demonstrates any 5 string operations using String class.

```
package assignment2;
import java.io.*;

public class Assignment2 {
   public static void main(String[] args) {
      try {
         DataInputStream d = new DataInputStream(System.in);

         System.out.println("Enter a String:");
         String s = d.readLine();

         // 1. length()
         System.out.println("Length of string: " + s.length());

         // 2. toUpperCase()
         System.out.println("Uppercase string: " + s.toUpperCase());
         // 3. toLowerCase()
         System.out.println("Lowercase string: " + s.toLowerCase());
         // 4. charAt()
         System.out.println("Character at index 2: " + s.charAt(2));
         // 5. substring()
         System.out.println("Substring from index 1 to 4: " + s.substring(1, 4));
      } catch (Exception e) {
         System.out.println(e);
      }
   }
}
```

Write a program that demonstrate simple inheritance

```
package assignment4;

// Parent class
class Animal {
   public void move() {
      System.out.println("Animals can move");
   }
}

// Child class
class Dog extends Animal {
   public void move() {
      System.out.println("Dogs can walk and run");
   }
}

// Main class
public class Assignment4 {
   public static void main(String[] args) {

      Animal a = new Animal();
      Animal b = new Dog();

      a.move();   // Parent class method
      b.move();   // Child class overridden method
   }
}
```

Write a program that demonstrate event handling for 3 types of events.

```
import java.awt.*;
import java.awt.event.*;

public class EventHandlingDemo extends Frame
    implements ActionListener, MouseListener, KeyListener {

  Button b;
  Label l;

  EventHandlingDemo() {
    b = new Button("Click Me");
    l = new Label("Perform an event");

    setLayout(new FlowLayout());
    add(b);
    add(l);

    // Register events
    b.addActionListener(this);
    addMouseListener(this);
    addKeyListener(this);

    setSize(300, 200);
    setVisible(true);
  }

  // 1. ActionEvent (Button Click)
```

```
    public void actionPerformed(ActionEvent e) {
        l.setText("Button Clicked");
    }

    // 2. MouseEvent (Mouse Click)
    public void mouseClicked(MouseEvent e) {
        l.setText("Mouse Clicked");
    }

    // Unused MouseListener methods
    public void mousePressed(MouseEvent e) {}
    public void mouseReleased(MouseEvent e) {}
    public void mouseEntered(MouseEvent e) {}
    public void mouseExited(MouseEvent e) {}

    // 3. KeyEvent (Key Press)
    public void keyPressed(KeyEvent e) {
        l.setText("Key Pressed");
    }

    // Unused KeyListener methods
    public void keyReleased(KeyEvent e) {}
    public void keyTyped(KeyEvent e) {}

    public static void main(String[] args) {
        new EventHandlingDemo();
    }
}
```

Write a program that demonstrate use of color dialog, input dialog box and menus

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

public class DialogMenuDemo extends JFrame implements ActionListener {

  JMenuItem colorItem, inputItem, exitItem;
  JLabel label;

  DialogMenuDemo() {
    // Label
    label = new JLabel("Dialog & Menu Demo", JLabel.CENTER);
    add(label);

    // Menu Bar
    JMenuBar mb = new JMenuBar();
    JMenu menu = new JMenu("Options");

    colorItem = new JMenuItem("Choose Color");
    inputItem = new JMenuItem("Input Dialog");
    exitItem = new JMenuItem("Exit");

    menu.add(colorItem);
    menu.add(inputItem);
    menu.add(exitItem);
    mb.add(menu);
```

```
    setJMenuBar(mb);

    // Register events
    colorItem.addActionListener(this);
    inputItem.addActionListener(this);
    exitItem.addActionListener(this);

    setSize(400, 300);
    setVisible(true);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public void actionPerformed(ActionEvent e) {

    // Color Dialog
    if (e.getSource() == colorItem) {
        Color c = JColorChooser.showDialog(this, "Select a Color", Color.white);
        if (c != null) {
            label.setForeground(c);
        }
    }

    // Input Dialog
    if (e.getSource() == inputItem) {
        String name = JOptionPane.showInputDialog(this, "Enter your name:");
        if (name != null) {
            label.setText("Hello " + name);
        }
```

```
    }

    // Exit
    if (e.getSource() == exitItem) {
      System.exit(0);
    }
  }

  public static void main(String[] args) {
    new DialogMenuDemo();
  }
}
```

Write a program to illustrate use of 5 swing components

```
package assignment7;

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

public class FiveSwingComponents extends JFrame implements ActionListener {

  JLabel label;
  JTextField textField;
  JCheckBox checkBox;
  JComboBox<String> comboBox;
  JButton button;

  FiveSwingComponents() {
```

```
setLayout(null);

label = new JLabel("Enter Name:");
label.setBounds(50, 30, 100, 25);
add(label);

textField = new JTextField();
textField.setBounds(150, 30, 150, 25);
add(textField);

checkBox = new JCheckBox("Student");
checkBox.setBounds(150, 70, 100, 25);
add(checkBox);

comboBox = new JComboBox<>(new String[]{"Java", "Python", "C++"});
comboBox.setBounds(150, 110, 100, 25);
add(comboBox);

button = new JButton("Submit");
button.setBounds(150, 150, 100, 30);
add(button);

button.addActionListener(this);

setSize(400, 250);
setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
    }

    public void actionPerformed(ActionEvent e) {
        System.out.println("Name: " + textField.getText());
        System.out.println("Student: " + checkBox.isSelected());
        System.out.println("Course: " + comboBox.getSelectedItem());
    }

    public static void main(String[] args) {
        new FiveSwingComponents();
    }
}
```

Write a program that demonstrate polymorphism

```
package assignment;

class Animal {
    void sound() {
        System.out.println("Animal makes a sound");
    }
}

class Dog extends Animal {
    void sound() {
        System.out.println("Dog barks");
    }
}
```

```
class Cat extends Animal {
    void sound() {
        System.out.println("Cat meows");
    }
}

public class PolymorphismDemo {
    public static void main(String[] args) {

        Animal a;   // reference of parent class

        a = new Dog();
        a.sound();   // calls Dog's sound()

        a = new Cat();
        a.sound();   // calls Cat's sound()
    }
}
```