slip14

```
class s14q1 {
    public static void main(String[] args) {
        int base = 2;
        int exponent = 3;
        int result = calPower(base, exponent);
        System.out.println(base + " raised to the power of " + exponent + " is " + result);
    }

    public static int calPower(int base, int exponent) {
        if (exponent == 0) {
            return 1; // base case: anything raised to the power of 0 is 1
        } else if (exponent < 0) {
            return 1 / calPower(base, -exponent);
        } else {
            return base * calPower(base, exponent - 1); // recursive case
        }
    }
}
```

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

```
public class EmployeeDetails {
    private JFrame frame1, frame2;
    private JTextField enoField, eNameField, salField;
    private JButton submitButton;

    public EmployeeDetails() {

        frame1 = new JFrame("Employee Details");
        frame1.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame1.setLayout(new FlowLayout());
         frame1.setSize(1000, 1000);
        enoField = new JTextField(5);
        eNameField = new JTextField(10);
        salField = new JTextField(5);

        frame1.add(new JLabel("Enter Employee No:"));
        frame1.add(enoField);
        frame1.add(new JLabel("Enter Employee Name:"));
        frame1.add(eNameField);
        frame1.add(new JLabel("Enter Salary:"));
        frame1.add(salField);
        frame1.add(new JLabel("Rs."));

        submitButton = new JButton("Submit");
        frame1.add(submitButton);
```

```
submitButton.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent e) {

        int eno = Integer.parseInt(enoField.getText());
        String eName = eNameField.getText();
        double sal = Double.parseDouble(salField.getText());



        frame2 = new JFrame("Employee Details");
        frame2.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame2.setLayout(new FlowLayout());

        frame2.add(new JLabel("Employee No: " + eno));
        frame2.add(new JLabel("Employee Name: " + eName));
        frame2.add(new JLabel("Salary: Rs." + sal));

        // Make the first frame invisible and the second frame visible
        frame1.setVisible(false);
        frame2.pack();
        frame2.setVisible(true);
    }
});

frame1.pack();
```

```
        frame1.setVisible(true);
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
            @Override
            public void run() {
                new EmployeeDetails();
            }
        });
    }
}
```

```
import tkinter as tk
import math

def calculate_cylinder():
    try:
        radius = float(radius_entry.get())
        height = float(height_entry.get())

        if radius < 0 or height < 0:
            result_label.config(text="Please enter positive values for radius and height.")
            return
        surface_area = 2 * math.pi * radius * (radius + height)
```

```
        volume = math.pi * radius ** 2 * height

        result_label.config(text=f"Surface Area: {surface_area:.2f}\nVolume: {volume:.2f}")
    except ValueError:
        result_label.config(text="Please enter valid numeric values for radius and height.")

root = tk.Tk()
root.title("Cylinder Calculator")

tk.Label(root, text="Enter Radius:").pack(pady=5)
radius_entry = tk.Entry(root)
radius_entry.pack(pady=5)

tk.Label(root, text="Enter Height:").pack(pady=5)
height_entry = tk.Entry(root)
height_entry.pack(pady=5)

calculate_button = tk.Button(root, text="Calculate", command=calculate_cylinder)
calculate_button.pack(pady=10)

result_label = tk.Label(root, text="")
result_label.pack(pady=10)

root.mainloop()
```

```
def encrypt_caesar(plain_text, shift):
```

```
    cipher_text = ""

    for char in plain_text:

        if char.isupper():

            cipher_text += chr((ord(char) + shift - 65) % 26 + 65)

        elif char.islower():

            cipher_text += chr((ord(char) + shift - 97) % 26 + 97)
        else:

            cipher_text += char
    return cipher_text

def main():

    plain_text = input("Enter the plain text: ")
    shift = int(input("Enter the shift value: "))

    cipher_text = encrypt_caesar(plain_text, shift)

    print(f"Plain Text: {plain_text}")
    print(f"Cipher Text: {cipher_text}")
```

main()