slip28

```
class s28q1 {

   public static void main(String[] args) {
      int count = 0;



      for (String arg : args) {
         try {


            Integer.parseInt(arg);


            count++;
         } catch (NumberFormatException e) {


         }
      }

       System.out.println("Number of integers in the given list: " + count);
   }
}
```

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

```
public class s28q2 {

    public static void main(String[] args) {


        String[][] employeeData = new String[5][3];


        String[] columnNames = {"Employee Number", "Employee Name", "Salary"};


        for (int i = 0; i < 5; i++) {

            String eno = JOptionPane.showInputDialog("Enter Employee Number for
Employee " + (i + 1) + ":");

            String ename = JOptionPane.showInputDialog("Enter Employee Name for
Employee " + (i + 1) + ":");

            String salary = JOptionPane.showInputDialog("Enter Salary for Employee " +
(i + 1) + ":");



            employeeData[i][0] = eno;

            employeeData[i][1] = ename;

            employeeData[i][2] = salary;

        }


        JTable table = new JTable(employeeData, columnNames);

        JScrollPane scrollPane = new JScrollPane(table);


        JFrame frame = new JFrame("Employee Details");

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        frame.setSize(500, 300);
```

```
        frame.setLayout(new BorderLayout());

        frame.add(scrollPane, BorderLayout.CENTER);

        frame.setVisible(true);
    }
}
```

```
import tkinter as tk
from tkinter import messagebox
class CourseListApp:
    def __init__(self, master):
        self.master = master
        self.master.title("Computer Science Courses")

        self.label = tk.Label(master, text="Select a Computer Science Course:")
        self.label.pack(pady=10)

        self.course_listbox = tk.Listbox(master, width=50, height=10)
        self.course_listbox.pack(pady=10)

        self.courses = [
            "Data Structures and Algorithms",
            "Database Management Systems",
            "Operating Systems",
```

```
            "Computer Networks",
        ]
        for course in self.courses:
            self.course_listbox.insert(tk.END, course)

        self.details_button = tk.Button(master, text="Show Course Details",
command=self.show_course_details)
        self.details_button.pack(pady=20)

    def show_course_details(self):
        try:
            selected_index = self.course_listbox.curselection()[0]
            selected_course = self.courses[selected_index]

            messagebox.showinfo("Course Details", f"You selected: {selected_course}")
        except IndexError:
            messagebox.showwarning("No Selection", "Please select a course from the
list.")

root = tk.Tk()
app = CourseListApp(root)
root.mainloop()
```

```
def merge_lists_to_tuples(list1, list2):
    return list(zip(list1, list2))

def main():
    list1 = input("Enter the first list of elements (comma-separated): ").split(',')
    list2 = input("Enter the second list of elements (comma-separated): ").split(',')

    list1 = [item.strip() for item in list1]
    list2 = [item.strip() for item in list2]

    merged_list = merge_lists_to_tuples(list1, list2)

    print("Merged List of Tuples:")
    print(merged_list)

main()
```