```
import java.util.ArrayList;
import java.util.Collections;
import java.util.InputMismatchException;
import java.util.Scanner;

public class Main {

   private static ArrayList<Integer> expenses = new ArrayList<Integer>();

   public static void main(String[] args) {
      System.out.println("\n**************************************\n");
      System.out.println("\tWelcome to TheDesk \n");
      System.out.println("**************************************");
      optionsSelection();
   }

   private static void optionsSelection() {
      String[] arr = {
            "1. I wish to review my expenditure",
            "2. I wish to add my expenditure",
            "3. I wish to delete my expenditure",
            "4. I wish to sort the expenditures",
            "5. I wish to search for a particular expenditure",
            "6. Close the application"
      };

      for (int i = 0; i < arr.length; i++) {
         System.out.println(arr[i]);
      }
```

```
        System.out.println("\nEnter your choice:\t");

        try (Scanner sc = new Scanner(System.in)) {
            int options = sc.nextInt();

            switch (options) {
                case 1:
                    System.out.println("Your saved expenses are listed below: \n");
                    System.out.println(expenses + "\n");
                    optionsSelection();
                    break;
                case 2:
                    System.out.println("Enter the value to add your Expense: \n");
                    int value = sc.nextInt();
                    expenses.add(value);
                    System.out.println("Your value is updated\n");
                    System.out.println(expenses + "\n");
                    optionsSelection();
                    break;
                case 3:
                    System.out.println("You are about to delete all your expenses! \nConfirm again by selecting
the same option...\n");
                    int con_choice = sc.nextInt();
                    if (con_choice == options) {
                        expenses.clear();
                        System.out.println(expenses + "\n");
                        System.out.println("All your expenses are erased!\n");
                    } else {
```

```
                System.out.println("Oops... try again!");
            }
            optionsSelection();
            break;
        case 4:
            sortExpenses(expenses);
            optionsSelection();
            break;
        case 5:
                System.out.println("Enter the expense you need to search:\t");
                int expenseToSearch = sc.nextInt();
                searchExpenses(expenses, expenseToSearch);
            optionsSelection();
            break;
        case 6:
            closeApp();
            break;
        default:
            System.out.println("You have made an invalid choice!");
            break;
      }
   } catch (InputMismatchException e) {
      System.out.println("Invalid input. Please enter a valid choice.");
      optionsSelection();
   }
}

private static void closeApp() {
   System.out.println("Closing your application... \nThank you!");
```

```
    }

    private static void searchExpenses(ArrayList<Integer> arrayList, int expenseToSearch) {
        int leng = arrayList.size();

        boolean found = false;
        for (int i = 0; i < leng; i++) {
            if (arrayList.get(i) == expenseToSearch) {
                found = true;
                break;
            }
        }

        if (found) {
            System.out.println("Expense found in the list.");
        } else {
            System.out.println("Expense not found in the list.");
        }
    }

    private static void sortExpenses(ArrayList<Integer> arrayList) {
        Collections.sort(arrayList);
        System.out.println("Expenses sorted in ascending order: " + arrayList);
    }
}
```