A computer monitor with text on it

AI-generated content may be incorrect.

Assignment 1

CS251



Loai Hataba 20230553

abdullah mohammed 20230231

hossam abdelaziz 20230121

**Languages:**

Loai → Java

Abdullah → Java

Hossam → Java

**Learning:**

|  |  |  |
| --- | --- | --- |
| Name | Duration | Sources |
| Loai | 8 Hours | <https://youtube.com/playlist?list=PLJhTWoCm8I6DXaq7XECfyGKtsq4Z6fWZr&si=w_PmOmUhKEH6EyCl>  <https://youtu.be/drQK8ciCAjY?si=cmx5cv4of_BQn4k5> |
| Abdullah | 2 Days |  |
| Hossam |  |  |

**Food Alternative (App 1 Loai):**

Main Function:

Scanner scanner = new Scanner(System.in);

// Load food from JSON

String jsonPath = "food/foodDictionary.json";

List<FoodItem> foodList = GsonTool.loadFood(jsonPath);

while (true){

            printBanner();

            int menu = optionsMenu(scanner);

            switch(menu)

            {

                // Alternative Food

                case 1:

                    foodMenu(scanner, foodList);

                    int ans = continueApp(scanner);

                    if (ans == 0){

                        scanner.close();

                        System.exit(0);

                    }

                    break;

                //Add new Food

                case 2:

                    addFood(scanner, foodList, jsonPath);

                    int ans2 = continueApp(scanner);

                    if (ans2 == 0){

                        scanner.close();

                        System.exit(0);

                    }

                    break;

                // Delete Food

                case 3:

                    deleteFood(scanner, foodList, jsonPath);

                    int ans3 = continueApp(scanner);

                    if (ans3 == 0){

                        scanner.close();

                        System.exit(0);

                    }

                    break;

                case 4:

                    prinInfoBanner();

                    int ans4 = continueApp(scanner);

                    if (ans4 == 0){

                        scanner.close();

                        System.exit(0);

                    }

                    break;

                case 5:

                    System.out.println("\nGoodbye!!");

                    scanner.close();

                    System.exit(0);

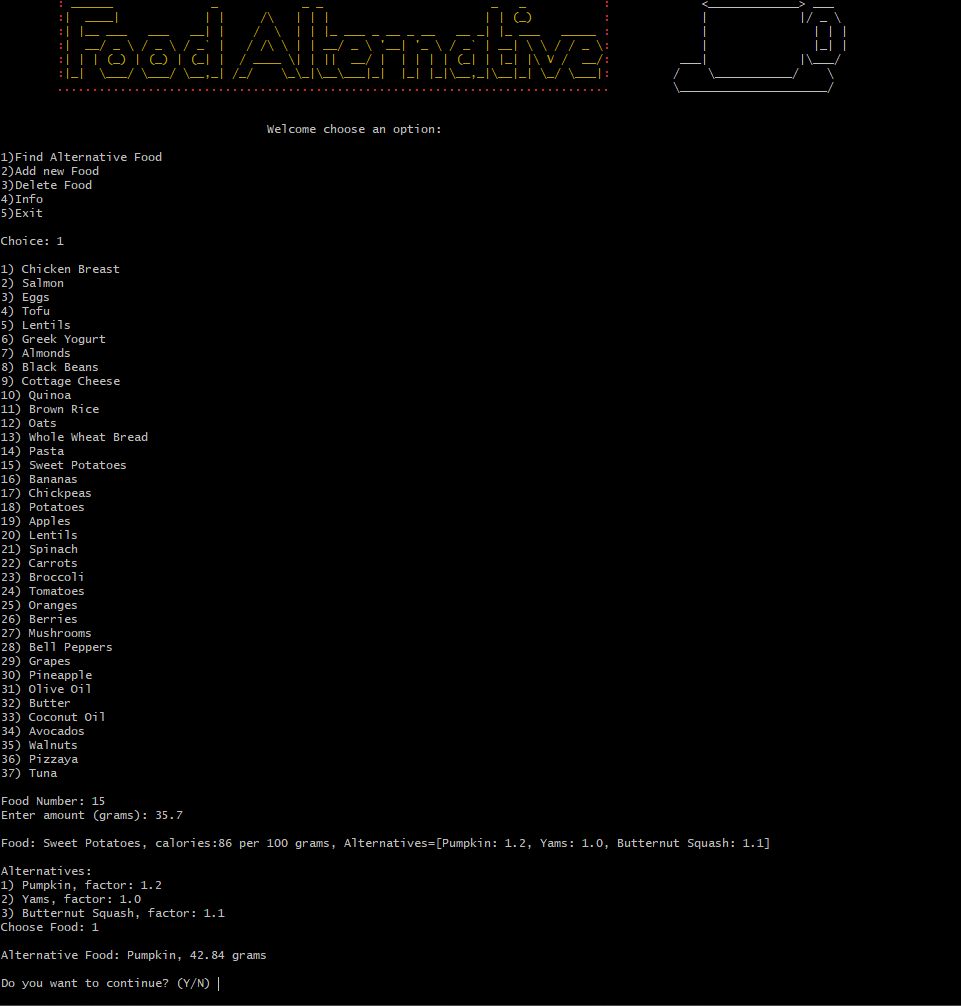
                    break;

            }

        }

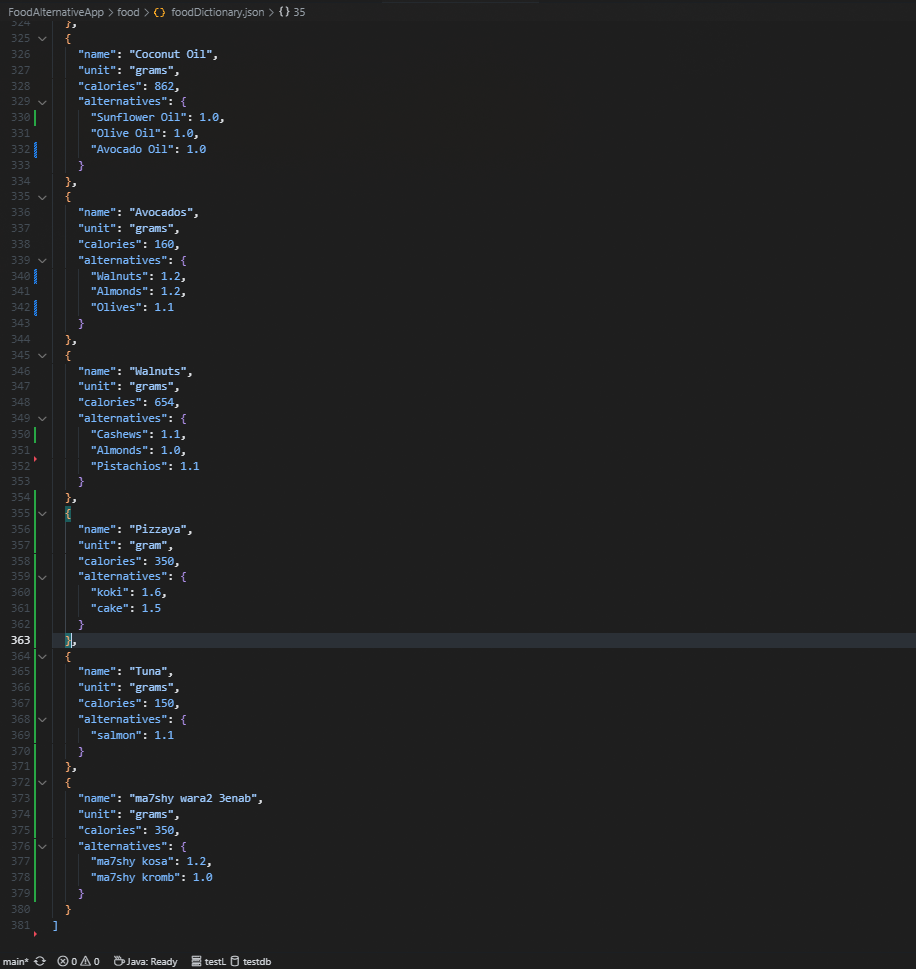
    }

Screenshots:

A screenshot of a computer

AI-generated content may be incorrect.

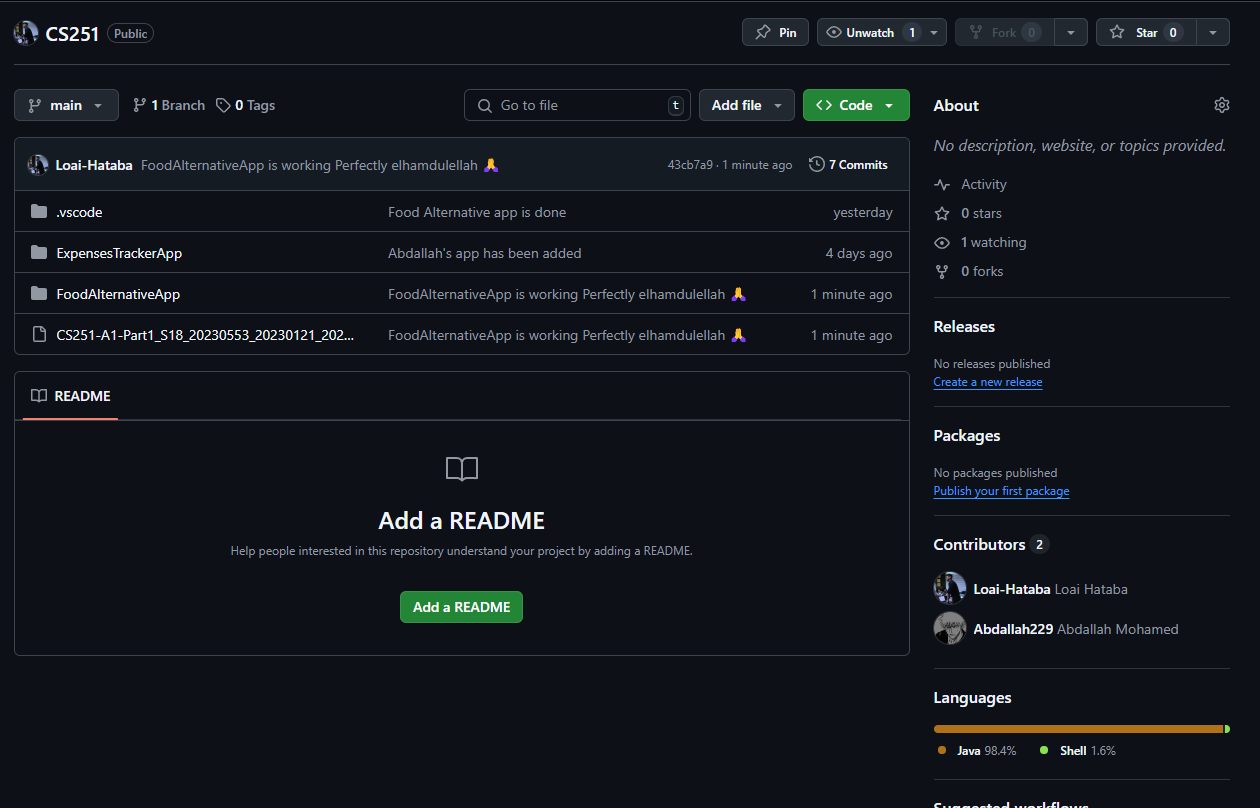






A screenshot of a computer

AI-generated content may be incorrect.



Video Link:

Link 😉

**Budget Tracker (App 2 Abdullah):**

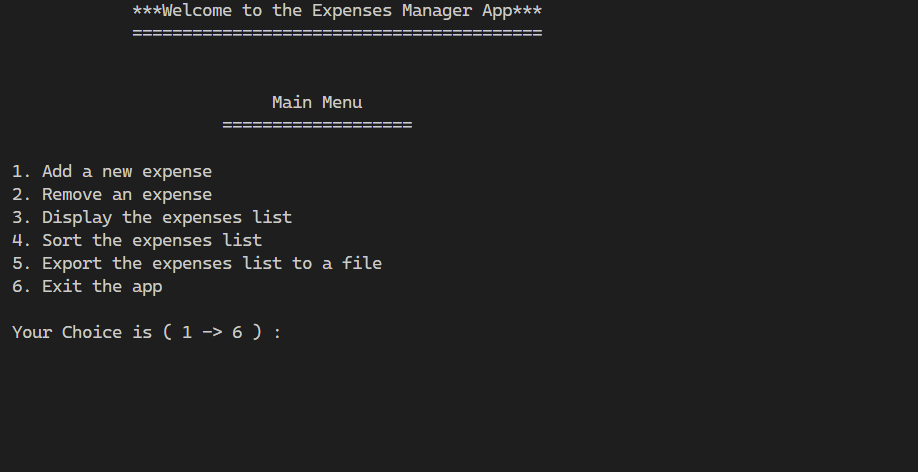
Main Function:

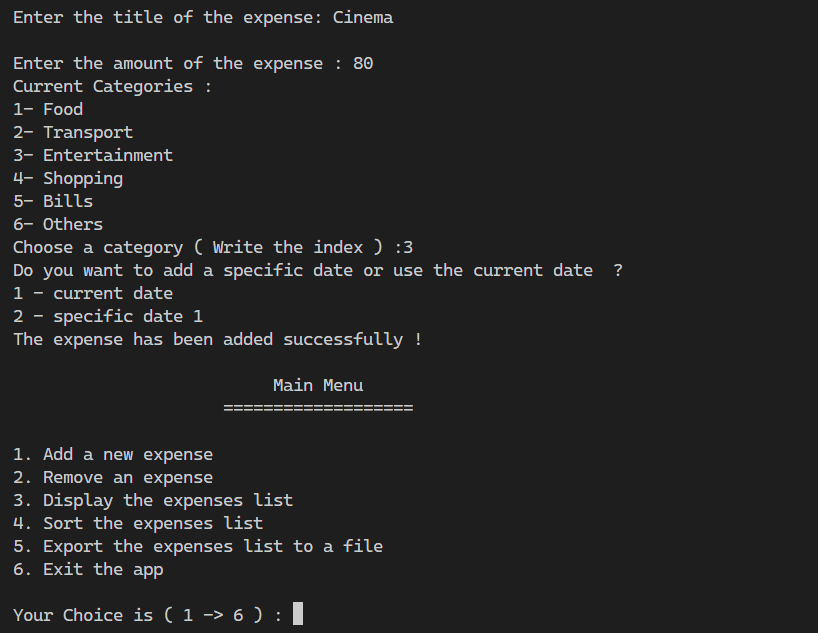
System.out.println(

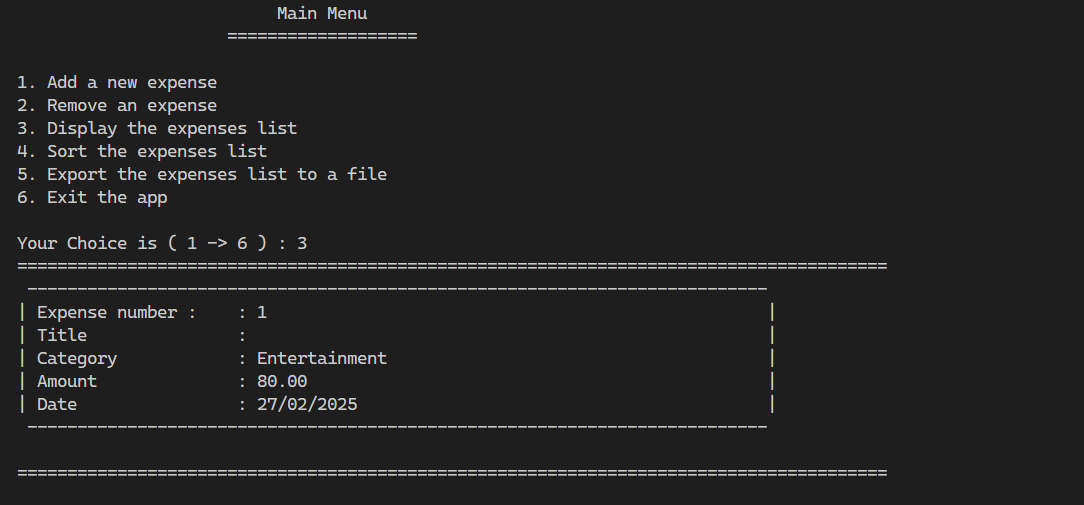
" \*\*\*Welcome to the Expenses Manager App\*\*\*");  
 System.out.println(

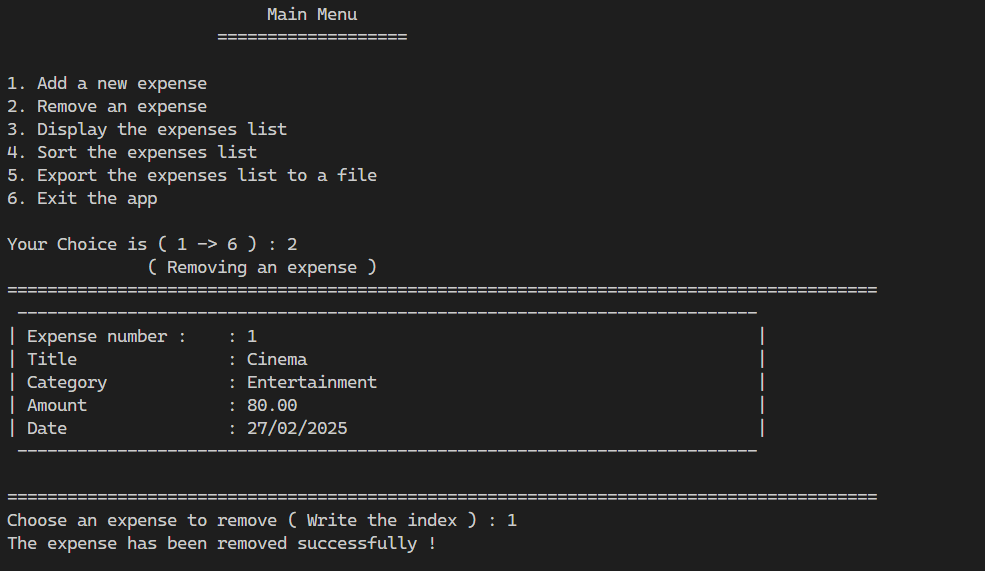
" =========================================\n");  
 // Create an instance of the expenses list   
 final ExpensesList myExpenses = new ExpensesList();  
 //The app menu :   
 while (true) {  
 System.out.println("\n Main Menu ");  
 System.out.println(" ===================\n");  
 // Main menu options :   
 // 1 Adding an expense :   
 System.out.println("1. Add a new expense ");  
 // 2 removing an expense :  
 System.out.println("2. Remove an expense ");  
 // 3 Display the expenses list :  
 System.out.println("3. Display the expenses list ");  
 // 4 Sort the expenses list :   
 System.out.println("4. Sort the expenses list ");  
 // 5 Export the expenses list to a file :  
 System.out.println("5. Export the expenses list to a file ");  
 // 6 Exit the app :  
 System.out.println("6. Exit the app ");  
 //read the user choice :  
  
 final int choice = validInput.getValidInt("\nYour Choice is ( 1 -> 6 ) : ", "Error : Invalid Choice !!", 1, 6);  
 switch (choice) {  
 case 1 ->  
 myExpenses.addExpense();  
 case 2 ->  
 myExpenses.removeExpense();  
 case 3 ->  
 myExpenses.displayExpenses();  
 case 4 ->  
 myExpenses.sortExpenses();  
 case 5 ->  
 myExpenses.exportExpenses();  
 case 6 -> {  
final int ch = validInput.getValidInt("Do you want to saving before closing ?\n1)Yes\n2)No ", "Error : Invalid Choice !!", 1, 2);  
 if (ch == 1) {  
 myExpenses.exportExpenses();  
 }  
 System.out.println("Terminating the program :(");  
 return;  
 }  
 default ->  
 throw new AssertionError();  
 }  
 }

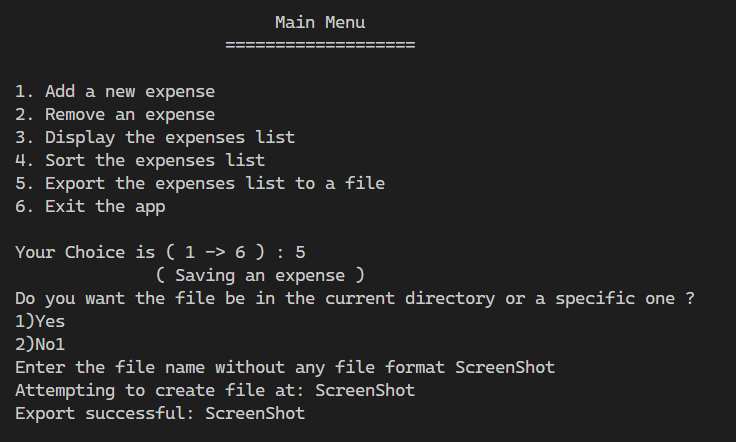
Screenshots:

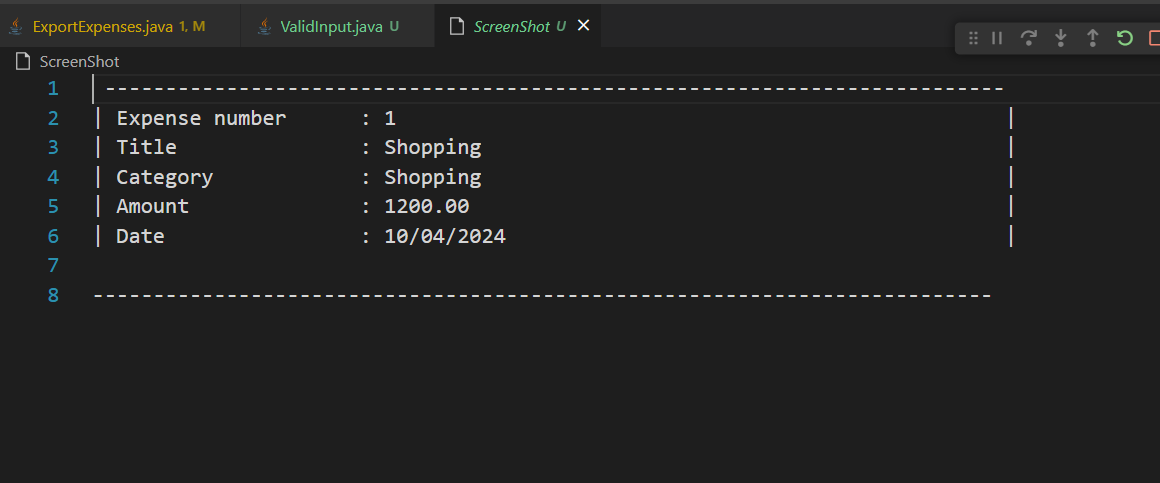












Video Link:

Link 😉

**Parking System (App 3 Hossam):**

Main Function:

Screenshots:

Video Link:

Link 😉