# Lab Task 4: Code Inception

## 4.1. Select 1 error checklist

## A screenshot of a phone Description automatically generated

## 4.2. Find an algorithm that does challenging computations and it’s the most suitable against the selected checklist.

* My code is a simple banking system that performs 4 basic tasks.
* Deposit, withdraw, balance checking, and exit option from the system.

## 4.3. Create 2 versions of algorithms, one is without applying the checklist and another one is which takes care of the checklist.

### 4.3.1. Code without applying the checklist.

import java.util.Scanner;

public class ErrorCheck{

public static void main(String args[]){

Scanner input=new Scanner(System.in);

double bal=0;

boolean ex=false;

System.out.println("Welcome to Banking System");

while(!ex){

System.out.println("\nChoose an option:");

System.out.println("1. Deposit");

System.out.println("2. Withdraw");

System.out.println("3. Check Balance");

System.out.println("4. Exit");

System.out.print("Enter your choice: ");

int ch = input.nextInt();

switch (ch) {

case 1:

System.out.print("Enter the amount to deposit: ");

double depositAmount = input.nextDouble();

bal += depositAmount;

System.out.println("Deposit successful!");

break;

case 2:

System.out.print("Enter the amount to withdraw: ");

double withdrawAmount = input.nextDouble();

if (withdrawAmount > bal) {

System.out.println("Insufficient funds!");

} else {

bal -= withdrawAmount;

System.out.println("Withdrawal successful!");

}

break;

case 3:

System.out.println("Your balance: Rs" + bal);

break;

case 4:

ex = true;

System.out.println("Thank you for using the Banking System!");

break;

default:

System.out.println("Invalid choice! Please choose a valid option.");

}

}

}

}

### 4.3.2. Code with applying the checklist.

import java.util.Scanner;

public class ErrorCheck {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

double bal = 0;

boolean ex = false;

System.out.println("Welcome to Banking System");

while (!ex) {

System.out.println("\nChoose an option:");

System.out.println("1. Deposit");

System.out.println("2. Withdraw");

System.out.println("3. Check Balance");

System.out.println("4. Exit");

try {

System.out.print("Enter your choice: ");

int ch = input.nextInt();

switch (ch) {

case 1:

System.out.print("Enter the amount to deposit: ");

try{

double depositAmount = input.nextDouble();

if(depositAmount>0){

bal += depositAmount;

System.out.println("Deposit successful!");}

else

System.out.println("Invalid amount for deposit. ");

}catch(Exception e){System.out.println(e);}

break;

case 2:

try{

System.out.print("Enter the amount to withdraw: ");

double withdrawAmount = input.nextDouble();

if(withdrawAmount>0){

if (withdrawAmount > bal) {

System.out.println("Insufficient funds!");

} else {

bal -= withdrawAmount;

System.out.println("Withdrawal successful!");

}}

else{

System.out.println("Invalid Amount");

}

}catch(Exception e){

System.out.println("Invalid input!");

input.next();

}

break;

case 3:

System.out.println("Your balance: Rs" + bal);

break;

case 4:

ex = true;

System.out.println("Thank you for using the Banking System!");

break;

default:

System.out.println("Invalid choice! Please choose a valid option.");

}

} catch (Exception e) {

System.out.println("Invalid menu! Please enter a valid menu.");

input.next();

}

}

}

}

## 4.4. Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test ID | Description | Input Data | Expected Outcome | Actual Outcome | Status |
| TC\_01 | To deposit amount from account | Menu choice:1  Amount=3000 | Balance should increase by 3000, "Deposit successful!" message should be displayed |  |  |
| TC\_02 | To withdraw amount from account | Menu choice:2  Amount=2000 | Balance should decrease by 2000, "Withdrawal successful!" message should be displayed |  |  |
| TC\_03 | To check balance from account | Menu choice:3 | Current balance should be displayed |  |  |
| TC\_04 | To exit from the system | Menu choice:4 | Program should terminate and displayed “Thank you for using the Banking system” |  |  |
| TC\_05 | To deposit nonnumeric value in account | Menu: 1  Amount: abs | "Invalid input! Please enter a valid integer." message should be displayed |  |  |
| TC\_06 | To input nonnumeric menu | Menu: e | “Invalid menu! Please enter a valid menu.” Message should be displayed |  |  |
| TC\_07 | To input menu value greater than 4 | Menu: 5 | “Invalid choice! Please enter a valid option.” Message should be displayed |  |  |
| TC\_08 | To deposit negative amount in account | Menu: 1  Amount=-900 | “Invalid amount for deposit” Message should be displayed |  |  |
| TC\_09 | To withdraw amount greater than current balance of user | Menu: 2  Amount: 80000 | "Insufficient funds!" message should be displayed |  |  |
| TC\_10 | To withdraw non-numeric value | Menu: 2  Amount: w | “Invalid amount” message should be displayed |  |  |
| TC\_11 | To withdraw negative amount | Menu: 2  Amount: -900 | “Invalid amount” message should be displayed |  |  |