

**Thapelo Mookeng**  
**IT Software Design and**  
**Software**

**JD521**

**9202085375080**

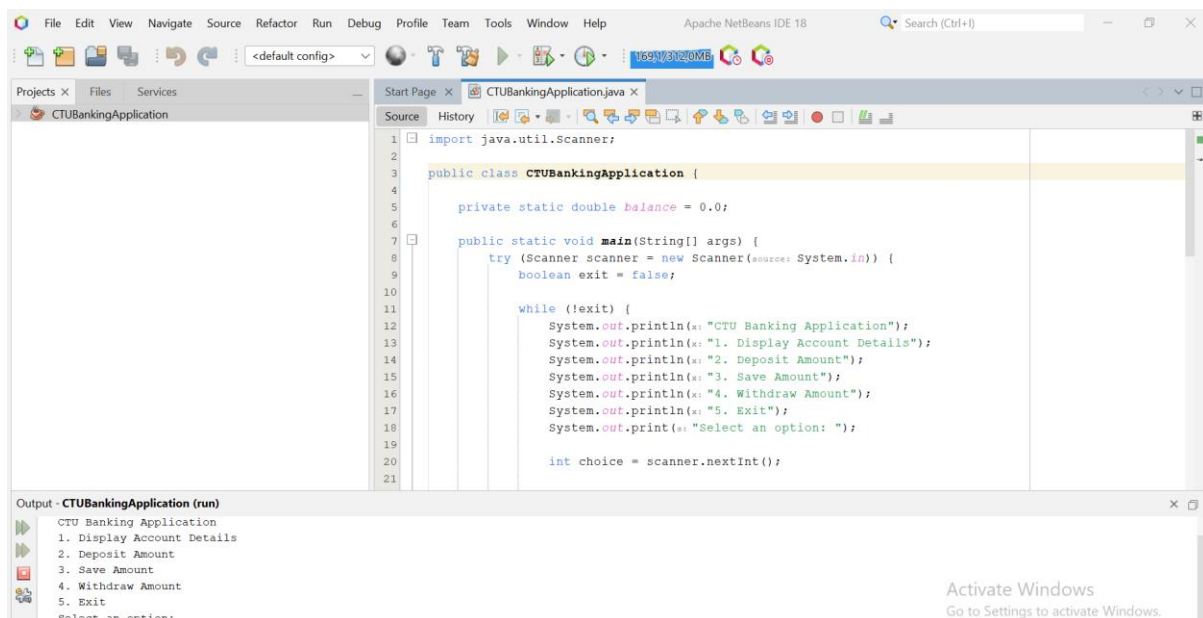
## Question 1: CTU Banking Application

1.1 Create a basic Java Application that will help CTU banking clients to perform basic transactions. The Application

should meet the following menu requirements:

- Display all account details
- Deposit the amount
- Save the amount
- Withdraw the amount
- Exit

Answer:



The screenshot shows the Apache NetBeans IDE 18 interface. The main editor window displays the source code for `CTUBankingApplication.java`. The code implements a simple banking application menu. Below the editor, the 'Output - CTUBankingApplication (run)' window shows the execution output, which matches the menu requirements listed in the question.

```
1 import java.util.Scanner;
2
3 public class CTUBankingApplication {
4
5     private static double balance = 0.0;
6
7     public static void main(String[] args) {
8         try (Scanner scanner = new Scanner(System.in)) {
9             boolean exit = false;
10
11             while (!exit) {
12                 System.out.println("CTU Banking Application");
13                 System.out.println("1. Display Account Details");
14                 System.out.println("2. Deposit Amount");
15                 System.out.println("3. Save Amount");
16                 System.out.println("4. Withdraw Amount");
17                 System.out.println("5. Exit");
18                 System.out.print("Select an option: ");
19
20                 int choice = scanner.nextInt();
21
22             }
23         }
24     }
25 }
```

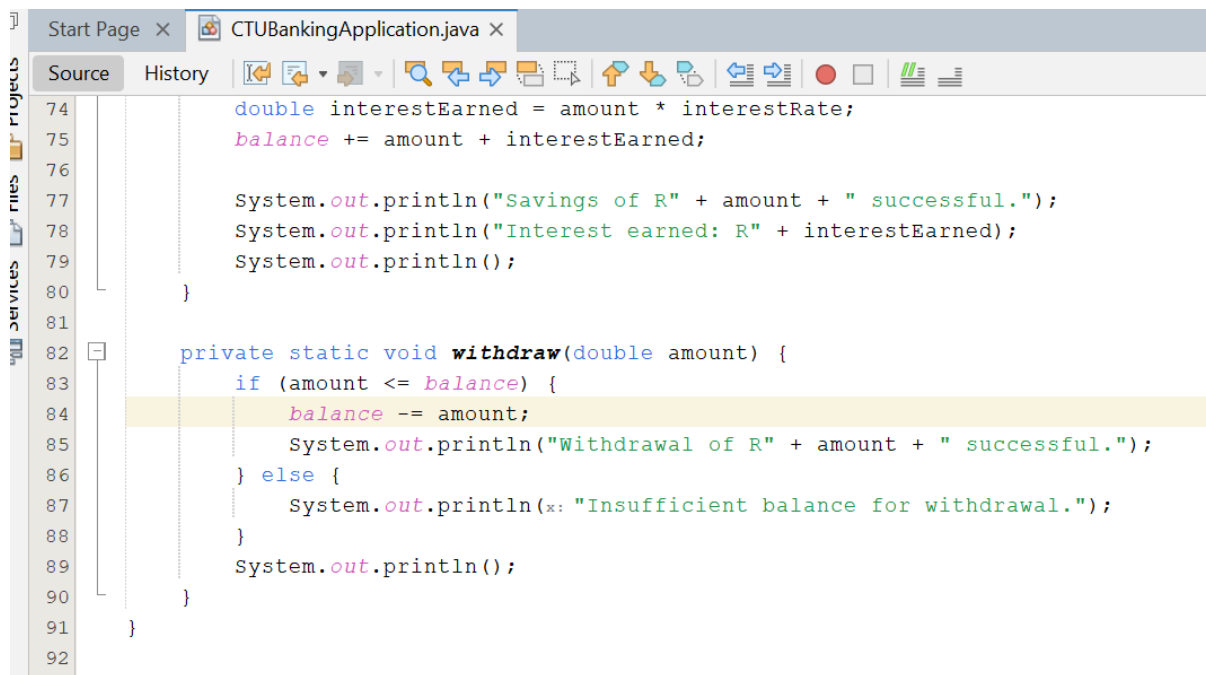
Output - CTUBankingApplication (run)

```
CTU Banking Application
1. Display Account Details
2. Deposit Amount
3. Save Amount
4. Withdraw Amount
5. Exit
Select an option:
```

```
Start Page x CTUBankingApplication.java x
Source History
20 int choice = scanner.nextInt();
21
22 switch (choice) {
23     case 1 -> displayAccountDetails();
24     case 2 -> {
25         System.out.print(s: "Enter the deposit amount: R");
26         double depositAmount = scanner.nextDouble();
27         deposit(amount: depositAmount);
28     }
29     case 3 -> {
30         System.out.print(s: "Enter the savings amount: R");
31         double savingsAmount = scanner.nextDouble();
32         save(amount: savingsAmount);
33     }
34     case 4 -> {
35         System.out.print(s: "Enter the withdrawal amount: R");
36         double withdrawalAmount = scanner.nextDouble();
37         withdraw(amount: withdrawalAmount);
38     }
39     case 5 -> {
40         exit = true;
41         System.out.println(x: "Exiting the application. Thank you!");
42     }
43     default -> System.out.println(x: "Invalid choice. Please select a valid option.");
44 }
45
46 System.out.println();
Activate Windows
```

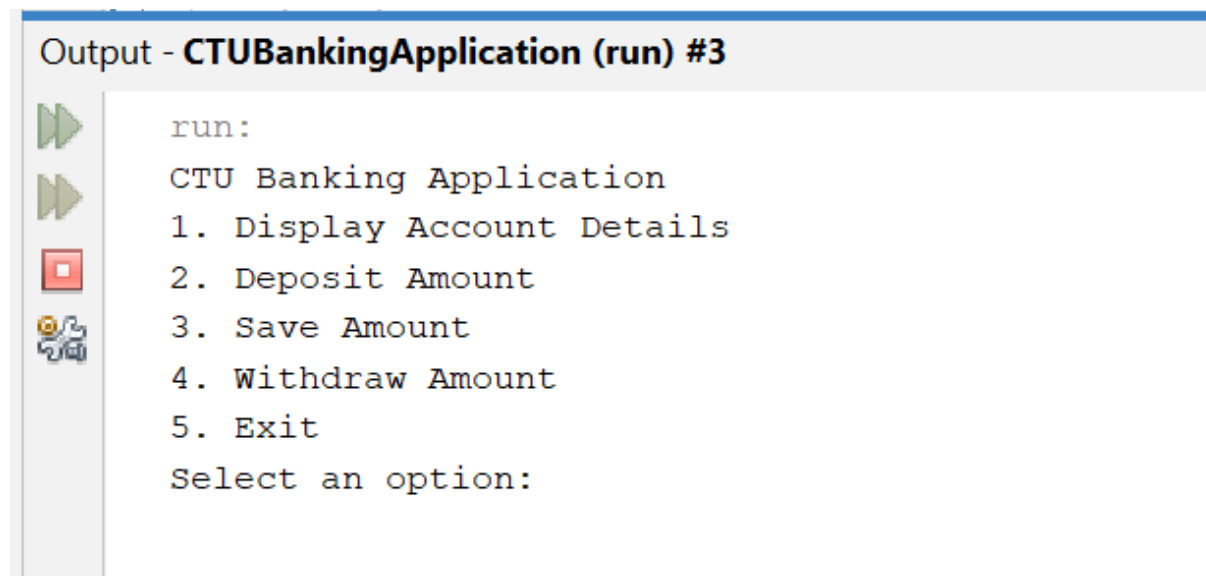
```
Start Page x CTUBankingApplication.java x
Source History
51 private static void displayAccountDetails() {
52     System.out.println(x: "Account Details:");
53     System.out.println("Current Balance: R" + balance);
54     System.out.println();
55 }
56
57 private static void deposit(double amount) {
58     balance += amount;
59     System.out.println("Deposit of R" + amount + " successful.");
60     System.out.println();
61 }
62
63 private static void save(double amount) {
64     // Implement interest rate calculation based on amount and duration
65     double interestRate = 0.0;
66     if (amount >= 100 && amount <= 500) {
67         interestRate = 0.005;
68     } else if (amount > 500 && amount <= 1000) {
69         interestRate = 0.02;
70     } else if (amount > 1000) {
71         interestRate = 0.05;
72     }
73
74     double interestEarned = amount * interestRate;
75     balance += amount + interestEarned;
76
77     System.out.println("Savings of R" + amount + " successful.");
78     System.out.println("Interest earned: R" + interestEarned);

```



```
74 double interestEarned = amount * interestRate;
75 balance += amount + interestEarned;
76
77 System.out.println("Savings of R" + amount + " successful.");
78 System.out.println("Interest earned: R" + interestEarned);
79 System.out.println();
80 }
81
82 private static void withdraw(double amount) {
83     if (amount <= balance) {
84         balance -= amount;
85         System.out.println("Withdrawal of R" + amount + " successful.");
86     } else {
87         System.out.println(x: "Insufficient balance for withdrawal.");
88     }
89     System.out.println();
90 }
91 }
92 }
```

OutPut:



```
Output - CTUBankingApplication (run) #3

run:
CTU Banking Application
1. Display Account Details
2. Deposit Amount
3. Save Amount
4. Withdraw Amount
5. Exit
Select an option:
```

### Output - CTUBankingApplication (run) #3



```
run:
CTU Banking Application
1. Display Account Details
2. Deposit Amount
3. Save Amount
4. Withdraw Amount
5. Exit
Select an option: 2
Enter the deposit amount: R10000
Deposit of R10000.0 successful.
```

### Output - CTUBankingApplication (run) #3



```
CTU Banking Application
1. Display Account Details
2. Deposit Amount
3. Save Amount
4. Withdraw Amount
5. Exit
Select an option: 3
Enter the savings amount: R3500
Savings of R3500.0 successful.
Interest earned: R175.0
```

### Output - CTUBankingApplication (run) #3



CTU Banking Application



1. Display Account Details



2. Deposit Amount



3. Save Amount

4. Withdraw Amount

5. Exit

Select an option: 4

Enter the withdrawal amount: R200

Withdrawal of R200.0 successful.

### Output - CTUBankingApplication (run) #3



CTU Banking Application



1. Display Account Details



2. Deposit Amount



3. Save Amount

4. Withdraw Amount

5. Exit

Select an option: 6

Invalid choice. Please select a valid option.

#### Output - CTUBankingApplication (run) #3



4. Withdraw Amount



5. Exit



Select an option: 6

Invalid choice. Please select a valid option.



CTU Banking Application

1. Display Account Details

2. Deposit Amount

3. Save Amount

4. Withdraw Amount

5. Exit

Select an option: 5

Exiting the application. Thank you!

**BUILD SUCCESSFUL** (total time: 8 minutes 25 seconds)