

Banking App Program using Arrays Screenshots

1. Add Accounts & View All Accounts

The screenshot shows the Eclipse IDE with the `BankingApp.java` file open. The code defines a `BankingApp` class with a `main` method that uses a `Scanner` to take user input and a `switch` statement to handle menu options. The `main` method calls `addAccount()` and `viewAllAccounts()` based on the user's choice.

The Console output shows the following sequence of events:

```
1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 1
Enter Account Name: Stella
Enter Account Type: Savings
Enter Account Balance: 1000
Is Account Active(Enter true or false): true
Account has been added successfully!

1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 1
Enter Account Name: Steffy
Enter Account Type: Savings
Enter Account Balance: 2000
Is Account Active(Enter true or false): true
Account has been added successfully!

1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 2
Account [id=1, name=Stella, type=Savings, balance=1000.0, active=true]
Account [id=2, name=Steffy, type=Savings, balance=2000.0, active=true]
```

2. View Account and Update Account

The screenshot shows the Eclipse IDE with the `BankingApp.java` file open. The code defines a `BankingApp` class with a `main` method that uses a `Scanner` to take user input and a `switch` statement to handle menu options. The `main` method calls `viewAccount()` and `updateAccount()` based on the user's choice.

The Console output shows the following sequence of events:

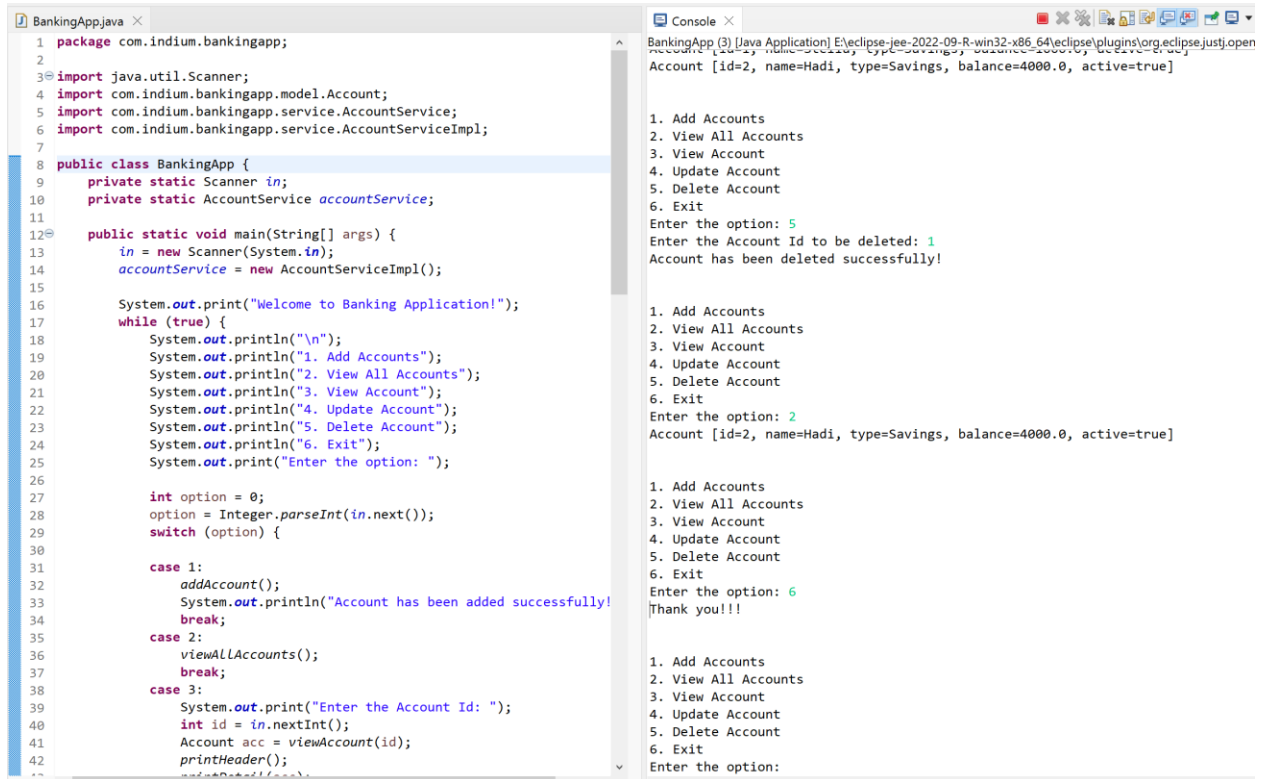
```
1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 3
Enter the Account Id: 1

   Id   Name  Type   Balance   Is Active
   --   -
    1   Stella Savings  1000.0    true

1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 4
Enter the Account Id to be updated: 2
Enter Account Name: Hadi
Enter Account Type: Savings
Enter Account Balance: 4000
Is Account Active(Enter true or false): true
Account has been updated successfully!

1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 2
Account [id=1, name=Stella, type=Savings, balance=1000.0, active=true]
Account [id=2, name=Hadi, type=Savings, balance=4000.0, active=true]
```

3. Delete Account and Exit Option



The screenshot displays the Eclipse IDE with the `BankingApp.java` file open on the left and the `Console` window on the right. The code in `BankingApp.java` implements a menu-driven application for managing accounts. It includes imports for `Scanner`, `Account`, `AccountService`, and `AccountServiceImpl`. The `main` method initializes a `Scanner` and an `AccountService` instance. A `while` loop displays a menu with six options: 1. Add Accounts, 2. View All Accounts, 3. View Account, 4. Update Account, 5. Delete Account, and 6. Exit. The user enters option 5, then the account ID 1. The console output shows the account details: `Account [id=2, name=Hadi, type=Savings, balance=4000.0, active=true]` and a confirmation message: `Account has been deleted successfully!`. The user then enters option 2, and the console shows the same account details. Finally, the user enters option 6, and the console shows `Thank you!!!`.

```
1 package com.indium.bankingapp;
2
3 import java.util.Scanner;
4 import com.indium.bankingapp.model.Account;
5 import com.indium.bankingapp.service.AccountService;
6 import com.indium.bankingapp.service.AccountServiceImpl;
7
8 public class BankingApp {
9     private static Scanner in;
10    private static AccountService accountService;
11
12    public static void main(String[] args) {
13        in = new Scanner(System.in);
14        accountService = new AccountServiceImpl();
15
16        System.out.println("Welcome to Banking Application!");
17        while (true) {
18            System.out.println("\n");
19            System.out.println("1. Add Accounts");
20            System.out.println("2. View All Accounts");
21            System.out.println("3. View Account");
22            System.out.println("4. Update Account");
23            System.out.println("5. Delete Account");
24            System.out.println("6. Exit");
25            System.out.print("Enter the option: ");
26
27            int option = 0;
28            option = Integer.parseInt(in.next());
29            switch (option) {
30
31                case 1:
32                    addAccount();
33                    System.out.println("Account has been added successfully!");
34                    break;
35                case 2:
36                    viewAllAccounts();
37                    break;
38                case 3:
39                    System.out.print("Enter the Account Id: ");
40                    int id = in.nextInt();
41                    Account acc = viewAccount(id);
42                    printHeader();
43            }
44        }
45    }
46
47    private void addAccount() {
48        // TODO: Add your code here
49    }
50
51    private void viewAllAccounts() {
52        // TODO: Add your code here
53    }
54
55    private Account viewAccount(int id) {
56        // TODO: Add your code here
57        return null;
58    }
59
60    private void printHeader() {
61        // TODO: Add your code here
62    }
63}
```

Console Output:

```
BankingApp (3) [Java Application] E:\eclipse-jee-2022-09-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.open
Account [id=2, name=Hadi, type=Savings, balance=4000.0, active=true]
1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 5
Enter the Account Id to be deleted: 1
Account has been deleted successfully!
1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 2
Account [id=2, name=Hadi, type=Savings, balance=4000.0, active=true]
1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option: 6
Thank you!!!
1. Add Accounts
2. View All Accounts
3. View Account
4. Update Account
5. Delete Account
6. Exit
Enter the option:
```