

Banking App Program using ArrayList Screenshots

1. Add Accounts & View All Accounts

The screenshot shows the BankingApp.java file and the console output. The code implements a menu-driven application with options to add, view, update, and delete accounts. The console shows the user selecting '1. Add Accounts', entering account details for 'Stella' (Savings, 1000, active), and then selecting '2. View All Accounts' to see the list of accounts.

```
1 import java.util.Scanner;
2
3 public class BankingApp {
4     private static Scanner in;
5     private static AccountService accountService;
6
7     public static void main(String[] args) {
8         in = new Scanner(System.in);
9         accountService = new AccountServiceImpl();
10
11         System.out.print("Welcome to Banking Application!");
12         while (true) {
13             System.out.println("\n");
14             System.out.println("1. Add Accounts");
15             System.out.println("2. View All Accounts");
16             System.out.println("3. View Account");
17             System.out.println("4. Update Account");
18             System.out.println("5. Delete Account");
19             System.out.println("6. Exit");
20             System.out.print("Enter the option: ");
21
22             int option = 0;
23             option = Integer.parseInt(in.next());
24             switch (option) {
25
26                 case 1:
27                     addAccount();
28                     System.out.println("Account has been added successfully!");
29                     break;
30                 case 2:
31                     viewAllAccounts();
32                     break;
33                 case 3:
34                     System.out.print("Enter the Account Id: ");
35                     int id = in.nextInt();
36                     Account acc = viewAccount(id);
37                     printHeader();
38                     printDetail(acc);
39                     break;
40                 case 4:
41                     updateAccount();
42                     System.out.println("Account has been updated successfully!");
43                     break;
```

Console Output:

```
BankingApp (1) [Java Application] E:\eclipse-jee-2022-09-R-win32-x86_64\plugins\org.eclipse.justj.openjdk.hotspot.jre.full
Welcome to Banking Application!

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]

1. Add Accounts
```

2. View Account & Update Account

The screenshot shows the BankingApp.java file and the console output. The code continues from the previous part, showing the user selecting '3. View Account' to view details of account ID 2 (Steffy, Savings, 2000.0, active). Then, the user selects '4. Update Account' to update account ID 2 with the name 'Hadi' and balance 3000. The console shows the updated account details.

```
1 import java.util.Scanner;
2
3 public class BankingApp {
4     private static Scanner in;
5     private static AccountService accountService;
6
7     public static void main(String[] args) {
8         in = new Scanner(System.in);
9         accountService = new AccountServiceImpl();
10
11         System.out.print("Welcome to Banking Application!");
12         while (true) {
13             System.out.println("\n");
14             System.out.println("1. Add Accounts");
15             System.out.println("2. View All Accounts");
16             System.out.println("3. View Account");
17             System.out.println("4. Update Account");
18             System.out.println("5. Delete Account");
19             System.out.println("6. Exit");
20             System.out.print("Enter the option: ");
21
22             int option = 0;
23             option = Integer.parseInt(in.next());
24             switch (option) {
25
26                 case 1:
27                     addAccount();
28                     System.out.println("Account has been added successfully!");
29                     break;
30                 case 2:
31                     viewAllAccounts();
32                     break;
33                 case 3:
34                     System.out.print("Enter the Account Id: ");
35                     int id = in.nextInt();
36                     Account acc = viewAccount(id);
37                     printHeader();
38                     printDetail(acc);
39                     break;
40                 case 4:
41                     updateAccount();
42                     System.out.println("Account has been updated successfully!");
43                     break;
```

Console Output:

```
BankingApp (1) [Java Application] E:\eclipse-jee-2022-09-R-win32-x86_64\plugins\org.eclipse.justj.openjdk.hotspot.jre.full

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: 2

Id      Name  Type    Balance  Is Active
2       Steffy Savings 2000.0   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: 3000
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=3000.0, active=true]

1. Add Accounts
2. View All Accounts
3. View Account
```

3. Delete Account & Exit Option

```
1 import java.util.Scanner;
2
3 public class BankingApp {
4     private static Scanner in;
5     private static AccountService accountService;
6
7     public static void main(String[] args) {
8         in = new Scanner(System.in);
9         accountService = new AccountServiceImpl();
10
11         System.out.print("Welcome to Banking Application!");
12         while (true) {
13             System.out.println("\n");
14             System.out.println("1. Add Accounts");
15             System.out.println("2. View All Accounts");
16             System.out.println("3. View Account");
17             System.out.println("4. Update Account");
18             System.out.println("5. Delete Account");
19             System.out.println("6. Exit");
20             System.out.print("Enter the option: ");
21
22             int option = 0;
23             option = Integer.parseInt(in.next());
24             switch (option) {
25
26                 case 1:
27                     addAccount();
28                     System.out.println("Account has been added successfully!");
29                     break;
30                 case 2:
31                     viewAllAccounts();
32                     break;
33                 case 3:
34                     System.out.print("Enter the Account Id: ");
35                     int id = in.nextInt();
36                     Account acc = viewAccount(id);
37                     printHeader();
38                     printDetail(acc);
39                     break;
40                 case 4:
41                     updateAccount();
42                     System.out.println("Account has been updated successfully!");
43                     break;
```

Console

BankingApp (1) [Java Application] E:\eclipse-jee-2022-09-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot

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=3000.0, 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=3000.0, 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!!!