

bankingapp-with-junit - bankingapp-with-junit/src/com/indium/bankingapp/service/AccountServiceHashMapImpl.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

AccountServiceHashMapImpl.java

```
1 package com.indium.bankingapp.service;
2
3 import com.indium.bankingapp.model.Account;
4
5 public class AccountServiceHashMapImpl implements AccountService<Account> {
6     private Map<Integer, Account> accounts = new HashMap<>();
7
8     @Override
9     public boolean createAccount(Account account) {
10         if (accounts.containsKey(account.getId())) {
11             // Account with the same ID already exists
12             return false;
13         }
14         accounts.put(account.getId(), account);
15         return true;
16     }
17
18     @Override
19     public boolean updateAccount(int id, Account account) {
20         if (accounts.containsKey(id)) {
21             accounts.put(id, account);
22             return true;
23         }
24         return false; // Account not found
25     }
26
27     @Override
28     public boolean deleteAccount(int id) {
29         if (accounts.containsKey(id)) {
30             accounts.remove(id);
31             return true;
32         }
33         return false; // Account not found
34     }
35
36     @Override
37     public Account getAccount(int id) {
38         if (accounts.containsKey(id)) {
39             return accounts.get(id);
40         }
41         return null;
42     }
43 }
```

bankingapp-with-junit - bankingapp-with-junit/src/com/indium/bankingapp/service/AccountServiceHashMapImpl.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

AccountServiceHashMapImpl.java x

```
13 public boolean createAccount(Account account) {
14     if (accounts.containsKey(account.getId())) {
15         // Account with the same ID already exists
16         return false;
17     }
18     accounts.put(account.getId(), account);
19     return true;
20 }
21
22 @Override
23 public boolean updateAccount(int id, Account account) {
24     if (accounts.containsKey(id)) {
25         accounts.put(id, account);
26         return true;
27     }
28     return false; // Account not found
29 }
30
31 @Override
32 public boolean deleteAccount(int id) {
33     if (accounts.containsKey(id)) {
34         accounts.remove(id);
35         return true;
36     }
37     return false; // Account not found
38 }
39
40 @Override
41 public Account getAccount(int id) {
42     return accounts.get(id);
43 }
44
45 @Override
46 public Collection<Account> getAllAccounts() {
47     return accounts.values();
48 }
49 }
```