Evgen959 / Advanced_Backend Public

<> Code ⊙ Issues Pull requests Actions Projects Security Insights

Advanced_Backend / Lesen 044 / code / hw_na_les044_1 / test / MainTest.java

Evgen959 newLes44 76d693f · 10 hours ago

87 lines (72 loc) · 3.09 KB

Code Blame Raw

```
import org.junit.jupiter.api.Assertions;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.DisplayName;
import org.junit.jupiter.api.Test;

import java.util.ArrayList;
import java.util.List;

import static org.junit.jupiter.api.Assertions.*;

class MainTest {
private List<BankAccount> bankAccounts;
    @BeforeEach
    void setUp() {
        bankAccounts = List.of(
            new BankAccount(new Peroson("Jack", 18), "DE11111", 90),
            new BankAccount(new Peroson("Nick", 15), "DE11112", 1000),
            new BankAccount(new Peroson("Regina", 23), "DE11113", 8.23),
            new BankAccount(new Peroson("Ann", 17), "DE11114", 5700),
            new BankAccount(new Peroson("Charly", 32), "DE11115", 31000),
            new BankAccount(new Peroson("Duck", 41), "DE11116", 99),
            new BankAccount(new Peroson("Sam", 28), "DE11117", 108000),
            new BankAccount(new Peroson("John", 19), "DE11118", 77.67)
    );
    }

    @Test
    @DisplayName("get all clients")
    void listHandler1() {
        List<Peroson> expectedClients = new ArrayList<>();

        expectedClients.add(new Peroson("Jack", 18));
        expectedClients.add(new Peroson("Nick", 15));
        expectedClients.add(new Peroson("Regina", 23));
        expectedClients.add(new Peroson("Ann", 17));
        expectedClients.add(new Peroson("Charly", 32));
        expectedClients.add(new Peroson("Duck", 41));
        expectedClients.add(new Peroson("Sam", 28));
        expectedClients.add(new Peroson("John", 19));

        List<Peroson> actualList = Main.listHandler(bankAccounts, ba -> true, ba -> ba.getPerson());
        assertEquals(expectedClients, actualList);
    }

    @Test
    @DisplayName("get all clients with balance<100")
    void listHandler2() {
        List<Peroson> expectedClients = new ArrayList<>();

        expectedClients.add(new Peroson("Jack", 18));
        expectedClients.add(new Peroson("Regina", 23));
        expectedClients.add(new Peroson("Duck", 41));
        expectedClients.add(new Peroson("John", 19));
```

```java

        List<Peroson> actualClients = Main.listHandler(bankAccounts, ba -> ba.getBalance()<100, ba -> ba.getPerson());
        assertEquals(expectedClients, actualClients);
    }

    @Test
    @DisplayName("get all IBANs")
    void listHandler3() {
        List<String> expectedIban = new ArrayList<>();

        expectedIban.add("DE11111");
        expectedIban.add("DE11112");
        expectedIban.add("DE11113");
        expectedIban.add("DE11114");
        expectedIban.add("DE11115");
        expectedIban.add("DE11116");
        expectedIban.add("DE11117");
        expectedIban.add("DE11118");

        List<String> actualIban = Main.listHandler(bankAccounts, ba -> true, ba -> ba.getIBAN());
        assertEquals(expectedIban, actualIban);
    }

    @Test
    @DisplayName("get Name To")
    void listHandler4() {
        List<String> expectedClients = new ArrayList<>();

        expectedClients.add("DE11112");

        List<String> actualList = Main.listHandler(bankAccounts, ba -> ba.getPerson().getAge()<16, ba -> ba.getIBAN());
        assertEquals(expectedClients, actualList);
    }
}
```