Source Code Of Handling User Authentication

1.UserNamePassword.java

**package** com.example;

**public** **class** UsernamePassword {

**private** String username;

**private** String password;

// Constructor

**public** UsernamePassword(String username, String password) {

**this**.username = username;

**this**.password = password;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

}

2.Authentication.Java

**package** com.example;

**public** **class** Authentication {

**public** **boolean** authenticateUser(UsernamePassword usernamePassword) {

**return** "admin".equals(usernamePassword.getUsername()) &&

"admin".equals(usernamePassword.getPassword());

}

}

3.AuthenticationTestClass

1.AuthenticationTrueCase

package com.example;

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

import org.junit.jupiter.api.Test;

public class AuthenticationTestTrue AuthenticationTestTrue {

@Test

public void testAuthenticateUser\_ValidCredentials\_ReturnsTrue() {

// Arrange

Authentication authentication = new Authentication();

UsernamePassword validCredentials = new UsernamePassword("admin", "admin");

// Act

boolean result = authentication.authenticateUser(validCredentials);

// Assert

assertTrue(result);

}

}

2.AuthenticationTestFalse

package com.example;

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

import org.junit.jupiter.api.Test;

public class AuthenticationTestFalse {

@Test

public void testAuthenticateUser\_InvalidCredentials\_ReturnsFalse() {

// Arrange

Authentication authentication = new Authentication();

UsernamePassword invalidCredentials = new UsernamePassword("user123", "password123");

// Act

boolean result = authentication.authenticateUser(invalidCredentials);

// Assert

assertFalse(result);

}

}