CODING SHUTTLE

Spring Testing

Understanding Mockito

Mockito

Mockito is useful for testing components in isolation without relying on their actual implementations or external dependencies.

Mockito can be used for:

- **Mocking**: Creating mock objects to simulate the behavior of real objects.
- **Stubbing**: Defining the behavior of mock objects when specific methods are called.
- **Verification**: Checking if certain methods were called on mock objects

Mockito Methods

1. **Creating Mock**

    with @Mock Annotation

    with Mockito.mock(Classname)

2. **Stubbing Mock**

    when(T methodCall): Specifies a method call to be stubbed.

    thenReturn(T value): Sets a return value for a stubbed method call.

    thenThrow(Throwable... throwables): Throws an exception when a stubbed method call is made.

Mockito Methods

3. Verifying Methods

verify(T mock): Verifies that a method was called on a mock.

verify(T mock, VerificationMode mode): Verifies that a method was called with a specific verification mode (e.g., times, never).

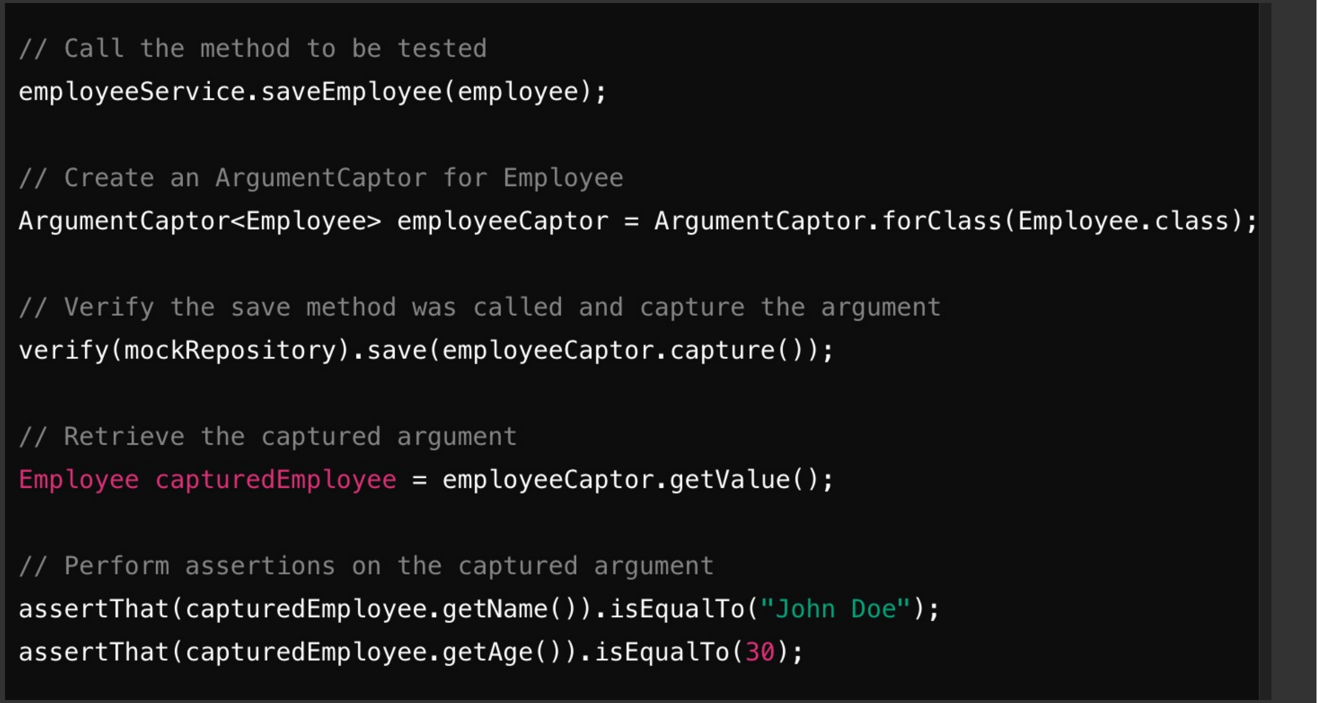
VerificationMode modes:

- times(int wantedNumberOfInvocations): Verifies the exact number of invocations.
- never(): Verifies that a method was never called.
- atLeastOnce(): Verifies that a method was called at least once.
- atLeast(int minNumberOfInvocations): at least called x times
- atMost(int maxNumberOfInvocations): at most called x times
- only(): Verifies that no other method was called on the mock.

Mockito Methods

4. Argument Captor:

It allows you to capture arguments for further assertions or validation, making it useful when you want to inspect the arguments that were passed to a method call.



```
// Call the method to be tested
employeeService.saveEmployee(employee);

// Create an ArgumentCaptor for Employee
ArgumentCaptor<Employee> employeeCaptor = ArgumentCaptor.forClass(Employee.class);

// Verify the save method was called and capture the argument
verify(mockRepository).save(employeeCaptor.capture());

// Retrieve the captured argument
Employee capturedEmployee = employeeCaptor.getValue();

// Perform assertions on the captured argument
assertThat(capturedEmployee.getName()).isEqualTo("John Doe");
assertThat(capturedEmployee.getAge()).isEqualTo(30);
```