# Mocks, stubs, domain, and anemic objects interview Q&A

Posted on September 2, 2014 by Arulkumaran Kumaraswamipillai — No Comments ↓

0 Like Share 0 G+1

Q1. What are mock objects?
A1. Mock objects are used in unit testing to ensure that your tests don't fail due to volatility of the data changes. There are mocking frameworks like **EasyMock**, **Mockito**, and **PowerMock**.

The key point to remember regarding mock objects is the ability of the mock objects to **verify** if a particular method was invoked and if yes, **how many times was invoked**. This is demonstrated with the **last two lines with the verify**

9 tips to earn more | What can u do to go places? | **945+** members. LinkedIn Group. **Reviews**

## 600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

**statement**. This is one of the key differences between using a mock object versus a stub.

```
import static org.mockito.Matchers.any;
import static org.mockito.Mockito.mock;
import static org.mockito.Mockito.times;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.when;

import org.junit.Before;
import org.junit.Test;
import org.mockito.Mock;
import org.mockito.Mockito;
import org.mockito.MockitoAnnotations;

public class MyAppControllerTest
{
   private static final String PORTFOLIO_CODE =
   private static final java.util.Date VALUATION

   private MyAppService mockMyAppService;
   private MyAppController controller;

  @Mock
  HttpServletResponse response;

  @Before
  public void setup() {
    MockitoAnnotations.initMocks(this);
    controller = new MyAppController();
    mockMyAppService = mock(MyAppServiceImpl.cla
    controller.setMyAppService(mockMyAppService)
 }

 @Test
 public void testGetPositionFeedCSV() throws Exc
    String str = "dummyCSV";
    //Set up behavior
    when(mockMyAppService.getPositionFeedCSV(any
    when(response.getWriter()).thenReturn(writer

    //Invoke controller
    controller.getPositionFeedCSV(PORTFOLIO_CODE

    //Verify behavior
    verify(mockMyAppService, times(1)).getPositi
    verify(writer, times(1)).write(any(String.cl
 }
}

```

Q2. What is the difference between fake objects, mock objects, and stubs?

A2. **Fake objects** build a very lightweight implementation of the same functionality as provided by a component that you are faking. Since they take some shortcut, they are not suitable for production.

## As a Java Architect

Java architecture & design concepts interview Q&As with diagrams | What should

**Mocks are objects** pre-programmed with expectations which form a specification of the calls they are expected to receive. You can use mocking frameworks like EasyMock, Mockito, PowerMock, etc to achieve this. When an actual service is invoked, a mock object is executed with a known outcome instead of the actual service. With mock objects, you can verify if expected method calls were made and how many times.

**Stubs** are like a mock class, except that they don't provide the ability to verify that methods have been called or not called. Generally services that are not ready or currently not stable are stubbed to make the test code more stable or to proceed with your development work to swap to the actual implementation when it is ready.

**When to use what?** You use a Mock when it's an object that returns values that you set to the tested class. You use a Stub to mimic an Interface or Abstract class to be tested. In fact, the difference is very subtle and it doesn't really matter what you call it, fake, mock, or stub, they are all objects that aren't used in production, and used for managing complexity to write quality tests.

Q3. What is a domain object?

A3. A domain object means a **business object**. Domain logic or business logic reside in "domain objects" and "business objects" that are **protocol independent**. You can access them via any protocol. Domain objects "store data" and "stored data specific business logic" and "domain services" will have business logic and manipulate the "domain objects". Domain services often make use of a DAO layer to retrieve and store persistent data.

**Domain Object with business logic**

```
@Entity
@Table(name = "account_rebalance")
public class Rebalance extends GenericDomainObje

    @Id
    @GeneratedValue(strategy = GenerationType.AUT
```

be a typical Java EE architecture?

## Senior Java developers must have a good handle on

open all | close all
- Best Practice (6)
- Coding (26)
- Concurrency (6)
- Design Concepts (7)
- Design Patterns (11)
- Exception Handling (3
- Java Debugging (21)
- Judging Experience I
- Low Latency (7)
- Memory Managemen
- Performance (13)
- QoS (8)
- Scalability (4)
- SDLC (6)
- Security (13)
- Transaction Managem

## 80+ step by step Java Tutorials

open all | close all
- Setting up Tutorial (6)
- Tutorial - Diagnosis (2
- Akka Tutorial (9)
- Core Java Tutorials (2
- Hadoop & Spark Tuto
- JEE Tutorials (19)
- Scala Tutorials (1)

```
    @Column(name = "acc_rebal_id")
    private Long id;

    @Column(name = "available_cash")
    private Decimal cashAvailable = Decimal.ZERO;

    @Column(name = "funding_method")
    @Enumerated(EnumType.STRING)
    private FundingMethod fundingMethod = Funding

    @Column(name = "instrument_type")
    @Enumerated(EnumType.STRING)
    private InstrumentType instrumentType;

    //..... other state variables

    //getters and setters omitted

    //domain or business logic
    public boolean isCash(){
       //logic to determine if cash product based
    }

    public boolean isInvestment(){
        //logic to determine if cash product base
    }

    public String getMaxHedgeFundAmount() {
        //business logic to determine  max hedge
    }

    public getCashBalanceError() {
       //business logic
    }
}

```

Spring & HIbernate Tu
Tools Tutorials (19)
Other Tutorials (45)

## Preparing for Java written & coding tests

open all | close all
♦ Complete the given
Can you write code?
Converting from A to I
Designing your classe
Java Data Structures
Passing the unit tests
What is wrong with th
Writing Code Home A
Written Test Core Jav
Written Test JEE (1)

### Domain Service interface

```
public interface RebalanceService {
    public Long saveRebalance(Rebalance rebalance
    List<Rebalance> findRebalances(RebalanceSearc
    Boolean cancelRebalance(Long id);
    public BigDecimal calculateCashBalance(Rebala
}

```

## How good are your...to go places?

open all | close all
Career Making Know-
Job Hunting & Resum

### Domain Service implementation with business logic

```
public class RebalanceServiceImpl implements   Re

    @Resource
    private RebalanceRepository rebalRepository;

    public Long saveRebalance(Rebalance rebalanc
       return rebalRepository.saveRebalance(reba
    }

    List<Rebalance> findRebalances(RebalanceSear
        //businness logic and data access via re
    }
    Boolean cancelRebalance(Long id) {
       //businness logic and data access via reb
    }
    public BigDecimal calculateCashBalance(Rebal
       // calculation business logic
    }
}

```

Q4. What are data transfer objects?
A4. A **Data Transfer Object** (DTO) is an object that is used to encapsulate data, and send it from one layer of an application to another. DTOs are most commonly used by the Services layer to transfer data to and from the UI layer. The main benefit is to map domain centric data to view centric data. There are frameworks like **Dozer** to map data from a domain object to a DTO. This conversion can be an expensive process and may not be useful if there are no remote calls involved. The domain objects themselves can be used as DTOs.

DTOs can be used as the **models** in the **MVC pattern**.

Another use for DTOs is to **encapsulate parameters for remote calls** to minimize the network round trips. DTOs have state variables and getter/setter methods.

Q5. What is an anemic model?
A5. Anemic domain model is the use of a domain model where the "domain objects" contain little or no business logic. This contradicts the notion of object-oriented design where you have well encapsulated logic.

An **anemic domain model is an anti-pattern** because in an anemic model, your domain logic exists somewhere else, probably in a class full of class(static) method or in multiple places, all with conflicting logic.

Q6. What is the purpose of Dozer framework?
A6. Convert Domain Objects to DTOs and DTOs back to Domain Objects in multi-tiered and multi-layered arechitecture.

## Popular Posts

♦ 11 Spring boot interview questions & answers

**857 views**

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

**825 views**

18 Java scenarios based interview Questions and Answers

**447 views**

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

**401 views**

♦ 7 Java debugging interview questions & answers

**311 views**

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

**302 views**

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

**292 views**

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

**286 views**

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

**263 views**

8 Git Source control system interview questions & answers

**215 views**

Bio | **Latest Posts**

**Arulkumaran Kumaraswamipillai**

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in

2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

**About** Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

‹ ♦ Java Collection overview interview questions and answers

"In your Java experience" interview questions & answers ›

**Posted in** member-paid**,** Objects

**Tags:** Core Java FAQs**,** Java/JEE FAQs

## Leave a Reply

Logged in as geethika. Log out?

**Comment**

Post Comment

## Empowers you to open more doors, and fast-track

### Technical Know Hows

☀ Java generics in no time ☀ Top 6 tips to transforming your thinking from OOP to FP ☀ How does a HashMap internally work? What is a hashing function?
☀ 10+ Java String class interview Q&As ☀ Java auto un/boxing benefits & caveats ☀ Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect

### Non-Technical Know Hows

☀ 6 Aspects that can motivate you to fast-track your career & go places ☀ Are you reinventing yourself as a Java developer? ☀ 8 tips to safeguard your Java career against offshoring ☀ My top 5 career mistakes

## Prepare to succeed

☀ Turn readers of your Java CV go from “Blah blah” to “Wow”? ☀ How to prepare for Java job interviews? ☀ 16 Technical Key Areas ☀ How to choose from multiple Java job offers?

Select Category ▼

## © Disclaimer