

**APIs for Java Developers**When Fortune 500 companies s
need Java File APIs, they use us

JAVA TUTORIAL

#INDEX POSTS

#INTERVIEW QUESTIONS

RESOURCES

HIRE ME

DOWNLOAD ANDROID APP

CONTRIBUTE

Subscribe to Download Java Design Patterns eBook

Full name

name@example.com

DOWNLOAD NOW

[HOME](#) » [INTERVIEW QUESTIONS](#) » OOPS INTERVIEW QUESTIONS AND ANSWERS

OOPS Interview Questions and Answers

APRIL 2, 2018 BY [PANKAJ](#) — [1 COMMENT](#)

Welcome to OOPS interview questions and answers. There are many Object Oriented Programming languages such as Java, C++ and Python. Having a clear idea about **OOPS concepts** is very important if you are going to face any interview on these programming languages. That's why I thought to share the top OOPS interview questions with you and provide detailed answers for them.

Table of Contents [\[hide\]](#)

1 OOPS Interview Questions

- 1.1 What is OOPS?
- 1.2 What are the advantages of OOPS concepts?
- 1.3 What is the difference between Procedural programming and OOPS?
- 1.4 What are the core concepts of OOPS?
- 1.5 What is Abstraction?
- 1.6 What is Encapsulation?
- 1.7 What is the difference between Abstraction and Encapsulation?
- 1.8 What is Polymorphism?
- 1.9 What is Inheritance?

- 1.10 What is multiple inheritance?
- 1.11 What is the diamond problem in inheritance?
- 1.12 Why Java does not support multiple inheritance?
- 1.13 What is Static Binding and Dynamic Binding?
- 1.14 What is the meaning of "IS-A" and "HAS-A" relationship?
- 1.15 What is Association?
- 1.16 What is Aggregation?
- 1.17 What is Composition?
- 1.18 What is Dependency?
- 1.19 What is the difference between Association and Dependency?
- 1.20 What is a Class?
- 1.21 What is an Object?

OOPS Interview Questions



1. What is OOPS?

Object Oriented Programming System is the programming technique to write programs based on the real world objects. The states and behaviors of an object are represented as the member variables and methods. In OOPS programming programs are organized around objects and data rather than actions and logic.

2. What are the advantages of OOPS concepts?

Major advantages of OOPS programming are;

1. **Simplicity:** OOPS programming objects model real world objects, so the complexity is reduced and the program structure is clear.
2. **Modularity:** Each object forms a separate entity whose internal workings are decoupled from other parts of the system.
3. **Modifiability:** It is easy to make minor changes in the data representation or the procedures in an OO program. Changes inside a class do not affect any other part of a program, since the only public interface that the external world has to a class is through the use of methods.

4. **Extensibility:** Adding new features or responding to changing operating environments can be solved by introducing a few new objects and modifying some existing ones.
5. **Maintainability:** Objects can be maintained separately, making locating and fixing problems easier.
6. **Reusability:** Objects can be reused in different programs.

3. **What is the difference between Procedural programming and OOPS?**

1. Procedural language is based on functions but object oriented language is based on real world objects.
2. Procedural language gives importance on the sequence of function execution but object oriented language gives importance on states and behaviors of the objects.
3. Procedural language exposes the data to the entire program but object oriented language encapsulates the data.
4. Procedural language follows top down programming paradigm but object oriented language follows bottom up programming paradigm.
5. Procedural language is complex in nature so it is difficult to modify, extend and maintain but object oriented language is less complex in nature so it is easier to modify, extend and maintain.
6. Procedural language provides less scope of code reuse but object oriented language provides more scope of code reuse.

4. **What are the core concepts of OOPS?**

OOPS core concepts are;

1. Abstraction
2. Encapsulation
3. Polymorphism
4. Inheritance
5. Composition
6. Association
7. Aggregation

5. **What is Abstraction?**

Abstraction is an OOPS concept to construct the structure of the real world objects. During this construction only the general states and behaviors are taken and more specific states and behaviors are left aside for the implementers.

6. **What is Encapsulation?**

Encapsulation is an OOPS concept to create and define the permissions and restrictions of an object and its member variables and methods. A very simple example to explain the concept is to make the member variables of a class private and providing public getter and setter methods. Java provides four types of access level modifiers: public, protected, no modifier and private.

7. **What is the difference between Abstraction and Encapsulation?**

1. "Program to interfaces, not implementations" is the principle for Abstraction and "Encapsulate what varies" is the OO principle for Encapsulation.
2. Abstraction provides a general structure of a class and leaves the details for the implementers. Encapsulation is to create and define the permissions and restrictions of an object and its member variables and methods.
3. Abstraction is implemented in Java using interface and abstract class while Encapsulation is implemented using four types of access level modifiers: public, protected, no modifier and private.

8. **What is Polymorphism?**

Polymorphism is the occurrence of something in various forms. Java supports various forms of polymorphism like polymorphic reference variables, polymorphic method, polymorphic return types and polymorphic argument types.

9. **What is Inheritance?**

A subclass can inherit the states and behaviors of its super class is known as inheritance.

10. **What is multiple inheritance?**

A child class inheriting states and behaviors from multiple parent classes is known as multiple inheritance.

11. **What is the diamond problem in inheritance?**

In case of multiple inheritance, suppose class A has two subclasses B and C, and a class D has two super classes B and C. If a method present in A is overridden by both B and C but not by D then from which class D will inherit that method B or C? This problem is known as diamond problem.

12. **Why Java does not support multiple inheritance?**

Java was designed to be a simple language and multiple inheritance introduces complexities like diamond problem. Inheriting states or behaviors from two different type of classes is a case which in reality very rare and it can be achieved easily through an object association.

13. **What is Static Binding and Dynamic Binding?**

Static or early binding is resolved at compile time. Method overloading is an example of static binding.

Dynamic or late or virtual binding is resolved at run time. Method overriding is an example of dynamic binding.

14. **What is the meaning of "IS-A" and "HAS-A" relationship?**

"IS-A" relationship implies inheritance. A sub class object is said to have "IS-A" relationship with the super class or interface. If class A extends B then A "IS-A" B. It is transitive, that is, if class A extends B

and class B extends C then A "IS-A" C. The "instanceof" operator in java determines the "IS-A" relationship.

When a class A has a member reference variable of type B then A "HAS-A" B. It is also known as Aggregation.

15. **What is Association?**

Association is a relationship between two objects with multiplicity.

16. **What is Aggregation?**

Aggregation is also known as "HAS-A" relationship. When class Car has a member reference variable of type Wheel then the relationship between the classes Car and Wheel is known as Aggregation. Aggregation can be understood as "whole to its parts" relationship.

Car is the whole and Wheel is part. Wheel can exist without the Car. Aggregation is a weak association.

17. **What is Composition?**

Composition is a special form of Aggregation where the part cannot exist without the whole. Composition is a strong Association. Composition relationship is represented like aggregation with one difference that the diamond shape is filled.

18. **What is Dependency?**

When one class depends on another because it uses that at some point in time then this relationship is known as Dependency. One class depends on another if the independent class is a parameter variable or local variable of a method of the dependent class. A Dependency is drawn as a dotted line from the dependent class to the independent class with an open arrowhead pointing to the independent class.

19. **What is the difference between Association and Dependency?**

The main difference between Association and Dependency is in case of Association one class has an attribute or member variable of the other class type but in case of Dependency a method takes an argument of the other class type or a method has a local variable of the other class type.

20. **What is a Class?**

A class is the specification or template of an object.

21. **What is an Object?**

Object is instance of class.

« PREVIOUS

Scala Advanced Interview Questions and Answers

NEXT »

RESTful Web Services Interview Questions

About Pankaj

If you have come this far, it means that you liked what you are reading. Why not reach little more and connect with me directly on [Google Plus](#), [Facebook](#) or [Twitter](#). I would love to hear your thoughts and opinions on my articles directly.

Recently I started creating video tutorials too, so do check out my videos on [Youtube](#).

FILED UNDER: [INTERVIEW QUESTIONS](#)

Comments

Anitha says

JULY 13, 2018 AT 4:14 AM

Your post is clear about OOPS concepts. Polymorphism, inheritance are clear to understand.

[Reply](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

☐

Save my name, email, and website in this browser for the next time I comment.

POST COMMENT



APIs for Java Developers

When Fortune 500 companies need Java File APIs, they use us

aspose.com



DOWNLOAD ANDROID APP



IMPORTANT INTERVIEW QUESTIONS

Java Interview Questions

- > [Core Java Interview Questions](#)
- > [String Interview Questions](#)
- > [Multithreading Interview Questions](#)
- > [Collections Interview Questions](#)
- > [Exception Interview Questions](#)
- > [Java Programming Interview Questions](#)
- > [Java 8 Interview Questions Part 1](#)
- > [Java 8 Interview Questions Part 2](#)
- > [Servlet Interview Questions](#)
- > [JSP Interview Questions](#)
- > [Struts 2 Interview Questions](#)
- > [JDBC Interview Questions](#)
- > [Spring Interview Questions](#)
- > [Hibernate Interview Questions](#)
- > [JSF Interview Questions](#)
- > [Web Services Interview Questions](#)
- > [Scala Basic Interview Questions](#)
- > [Scala Intermediate Interview Questions](#)
- > [Scala Advanced Interview Questions](#)
- > [Scala Interview Questions Summary](#)
- > [Common Job Interview Questions](#)

Miscellaneous

- > [Java ClassLoader](#)
- > [String StringBuffer StringBuilder](#)
- > [Java is Pass By Value](#)
- > [Java Heap vs Stack Memory](#)

RECOMMENDED TUTORIALS

Java Tutorials

- > [Java IO](#)
- > [Java Regular Expressions](#)

- > [Multithreading in Java](#)
- > [Java Logging](#)
- > [Java Annotations](#)
- > [Java XML](#)
- > [Collections in Java](#)
- > [Java Generics](#)
- > [Exception Handling in Java](#)
- > [Java Reflection](#)
- > [Java Design Patterns](#)
- > [JDBC Tutorial](#)

Java EE Tutorials

- > [Servlet JSP Tutorial](#)
- > [Struts2 Tutorial](#)
- > [Spring Tutorial](#)
- > [Hibernate Tutorial](#)
- > [Primefaces Tutorial](#)
- > [Apache Axis 2](#)
- > [JAX-RS](#)
- > [Memcached Tutorial](#)

Basic Java Tutorials

Download Java Plugins

Java Game Development Tutorials

Java Books for Beginners

Free Interactive Java Tutorial

Download Java Development Kit

Java Tutorials for Beginners

Beginners Java Script Tutorials



