

# Programming Interview Questions And Answers Guide.



**Global Guideline.**

<https://globalguideline.com/>



# Programming Job Interview Preparation Guide.

### Question # 1

What is Python and what is scope of Python?

#### Answer:-

Python is an interpreted, interactive, object-oriented programming language. It is often compared to Tcl, Perl, Scheme or Java. Python combines remarkable power with very clear syntax. It has modules, classes, exceptions, very high level dynamic data types, and dynamic typing. There are interfaces to many system calls and libraries, as well as to various windowing systems (X11, Motif, Tk, Mac, MFC, wxWidgets). New built-in modules are easily written in C or C++. Python is also usable as an extension language for applications that need a programmable interface. The Python implementation is portable: it runs on many brands of UNIX, on Windows, OS/2, Mac, Amiga, and many other platforms. The Python implementation is copyrighted but freely usable and distributable, even for commercial use.

[Read More Answers.](#)

### Question # 2

If a class is declared without any access modifiers, where may the class be accessed?

#### Answer:-

A class that is declared without any access modifiers is said to have package access. This means that the class can only be accessed by other classes and interfaces that are defined within the same package.

[Read More Answers.](#)

### Question # 3

Does a class inherit the constructors of its superclass in programming?

#### Answer:-

A class does not inherit constructors from any of its super classes.

[Read More Answers.](#)

### Question # 4

What is the difference between shadow and override in programming?

#### Answer:-

Overriding is used to redefines only the methods, but shadowing redefines the entire element.

[Read More Answers.](#)

### Question # 5

What is multithreading in a programming language?

#### Answer:-

Multithreading is the mechanism in which more than one thread run independent of each other within the process in any programming language such as C, C++, Java etc...

[Read More Answers.](#)

### Question # 6

What are inner class and anonymous class concept in Programming?

#### Answer:-

Inner class in Programming:

Classes defined in other classes, including those defined in methods are called inner classes. An inner class can have any accessibility including private.

Anonymous class in Programming:

Anonymous class is a class defined inside a method without a name and is instantiated and declared in the same place and cannot have explicit constructors.

[Read More Answers.](#)



### Question # 7

What is meant by "method-wars" in Programming?

#### Answer:-

Before 1994 there were different methodologies like Rumbaugh, Booch, Jacobson, Meyer etc who followed their own notations to model the systems. The developers were in a dilemma to choose the method which best accomplishes their needs. This particular span was called as "method-wars".

[Read More Answers.](#)

### Question # 8

Who were the three famous amigos and what was their contribution to the object community?

#### Answer:-

The Three amigos namely,  
James Rumbaugh (OMT): A veteran in analysis who came up with an idea about the objects and their Relationships (in particular Associations).  
Grady Booch: A veteran in design who came up with an idea about partitioning of systems into subsystems.  
Ivar Jacobson (Objectory): The father of USECASES, who described about the user and system interaction.

[Read More Answers.](#)

### Question # 9

What is Downcasting in Programming?

#### Answer:-

Downcasting concept is the casting from a general to a more specific type, i.e. casting down the hierarchy in programming.

[Read More Answers.](#)

### Question # 10

Can a method be overloaded based on different return type but same argument type in programming?

#### Answer:-

No, method can not be overloaded because the methods can be called without using their return type in which case there is ambiguity for the compiler in programming languages.

[Read More Answers.](#)

### Question # 11

What are the advantages of OOP in a programming language?

#### Answer:-

Object oriented programming languages directly represent the real life objects. The features of OOP as inheritance, polymorphism, encapsulation makes it powerful.

[Read More Answers.](#)

### Question # 12

What is the difference between procedural and object-oriented programs?

#### Answer:-

1. In procedural program, programming logic follows certain procedures and the instructions are executed one after another. In OOP program, unit of program is object, which is nothing but combination of data and code.
2. In procedural program, data is exposed to the whole program whereas in OOPs program, it is accessible within the object and which in turn assures the security of the code.

[Read More Answers.](#)

### Question # 13

How many ways can an argument be passed to a subroutine in programming?

#### Answer:-

An argument can be passed in two ways in a programming language. They are Pass by Value and Passing by Reference.  
Passing by value: This method copies the value of an argument into the formal parameter of the subroutine.  
Passing by reference: In this method, a reference to an argument (not the value of the argument) is passed to the parameter.

[Read More Answers.](#)

### Question # 14

What is difference between overloading and overriding in programming language?

#### Answer:-

Difference between overloading and overriding in programming language is:

- a) In overloading, there is a relationship between methods available in the same class whereas in overriding, there is relationship between a superclass method and subclass method.
- b) Overloading does not block inheritance from the superclass whereas overriding blocks inheritance from the superclass.
- c) In overloading, separate methods share the same name whereas in overriding, subclass method replaces the superclass.
- d) Overloading must have different method signatures whereas overriding must have same signature.

[Read More Answers.](#)

### Question # 15



What is the difference between superclass and subclass in programming?

**Answer:-**

A super class is a class that is inherited in programming whereas the sub class is a class that does the inheriting in programming.

[Read More Answers.](#)

**Question # 16**

Do you know about Object Oriented Programming Essentials and History?

**Answer:-**

An object-oriented programming language (also called an OO language) is one that allows or encourages, to some degree, object-oriented programming methods. Simula (1967) is generally accepted as the first language to have the primary features of an object-oriented language. It was created for making simulation programs, in which what came to be called objects were the most important information representation. Smalltalk (1972 to 1980) is arguably the canonical example, and the one with which much of the theory of object-oriented programming was developed.

OO languages can be grouped into several broad classes, determined by the extent to which they support all features and functionality of object-orientation and objects: classes, methods, polymorphism, inheritance, and reusability.

[Read More Answers.](#)

**Question # 17**

When does a name clash occur in programming?

**Answer:-**

A name clash occurs when a name is defined in more than one place. For example., two different class libraries could give two different classes the same name. If you try to use many class libraries at the same time, there is a fair chance that you will be unable to compile or link the program because of name clashes.

[Read More Answers.](#)

**Question # 18**

List out some of the object-oriented methodologies?

**Answer:-**

Object Oriented Development (OOD) (Booch 1991,1994).  
Object Oriented Analysis and Design (OOA/D) (Coad and Yourdon 1991).  
Object Modeling Techniques (OMT) (Rumbaugh 1991).  
Object Oriented Software Engineering (Objectory) (Jacobson 1992).  
Object Oriented Analysis (OOA) (Shlaer and Mellor 1992).  
The Fusion Method (Coleman 1991).

[Read More Answers.](#)

**Question # 19**

Differentiate persistent & non-persistent objects in programming?

**Answer:-**

Persistent refers to an object's ability to transcend time or space. A persistent object stores/saves its state in a permanent storage system with out losing the information represented by the object.

A non-persistent object is said to be transient or ephemeral. By default objects are considered as non-persistent.

[Read More Answers.](#)

**Question # 20**

Differentiate Aggregation and containment in Programming?

**Answer:-**

Aggregation is the relationship between the whole and a part. We can add/subtract some properties in the part (slave) side. It won't affect the whole part.

Best example is Car, which contains the wheels and some extra parts. Even though the parts are not there we can call it as car.

But, in the case of containment the whole part is affected when the part within that got affected. The human body is an apt example for this relationship. When the whole body dies the parts (heart etc) are died.

[Read More Answers.](#)

**Question # 21**

What is a modifier explain?

**Answer:-**

A modifier, also called a modifying function is a member function that changes the value of at least one data member. In other words, an operation that modifies the state of an object. Modifiers are also known as 'mutators'. Example: The function mod is a modifier in the following code snippet:

```
class test
{
  int x,y;
  public:
  test()
  {
    x=0; y=0;
  }
  void mod()
  {
    x=10;
    y=15;
  }
}
```



```
}  
};
```

[Read More Answers.](#)

### Question # 22

Differentiate between a template class and class template in programming?

**Answer:-**

Template class: A generic definition or a parametrized class not instantiated until the client provides the needed information. It's jargon for plain templates.

Class template: A class template specifies how individual classes can be constructed much like the way a class specifies how individual objects can be constructed. It's jargon for plain classes.

[Read More Answers.](#)

### Question # 23

What is a dangling pointer in programming?

**Answer:-**

A dangling pointer arises when you use the address of an object after its lifetime is over. This may occur in situations like returning addresses of the automatic variables from a function or using the address of the memory block after it is freed. The following code snippet shows this:

```
class Sample  
{  
public:  
int *ptr;  
Sample(int i)  
{  
ptr = new int(i);  
}  
~Sample()  
{  
delete ptr;  
}  
void PrintVal()  
{  
cout << "The value is " << *ptr;  
}  
};  
void SomeFunc(Sample x)  
{  
cout << "Say i am in someFunc " << endl;  
}  
int main()  
{  
Sample s1 = 10;  
SomeFunc(s1);  
s1.PrintVal();  
}
```

In the above example when PrintVal() function is called it is called by the pointer that has been freed by the destructor in SomeFunc.

[Read More Answers.](#)

### Question # 24

Differentiate between the message and method in programming?

**Answer:-**

Message in Programming:

- \* Objects communicate by sending messages to each other.
- \* A message is sent to invoke a method.

Method in Programming:

- \* Provides response to a message.
- \* It is an implementation of an operation.

[Read More Answers.](#)

### Question # 25

What do you mean by programming analysis and design?

**Answer:-**

Analysis: It is the process of determining what needs to be done before how it should be done. In order to accomplish this, the developer refers the existing systems and documents. So, simply it is an art of discovery.

Design: It is the process of adopting/choosing the one among the many, which best accomplishes the users needs. So, simply, it is compromising mechanism.

[Read More Answers.](#)



### Question # 26

What are the steps involved in designing in programming?

#### Answer:-

Before getting into the design the designer should go through the SRS prepared by the System Analyst. The main tasks of design are Architectural Design and Detailed Design. In Architectural Design we find what are the main modules in the problem domains Detailed Design we find what should be done within each module.

[Read More Answers.](#)

### Question # 27

What do you mean by active and passive objects?

#### Answer:-

Active objects are one which instigate an interaction which owns a thread and they are responsible for handling control to other objects. In simple words it can be referred as client.

Passive objects are one, which passively waits for the message to be processed. It waits for another object that requires its services. In simple words it can be referred as server.

[Read More Answers.](#)

### Question # 28

Differentiate persistent & non-persistent programming objects?

#### Answer:-

Persistent refers to an object's ability to transcend time or space. A persistent object stores/saves its state in a permanent storage system with out losing the information represented by the object.

A non-persistent object is said to be transient or ephemeral. By default objects are considered as non-persistent.

[Read More Answers.](#)

### Question # 29

What is meant by software development method?

#### Answer:-

Software development method describes how to model and build software systems in a reliable and reproducible way. To put it simple, methods that are used to represent ones' thinking using graphical notations.

[Read More Answers.](#)

### Question # 30

What do you mean by static and dynamic modeling in programming?

#### Answer:-

Static modeling is used to specify structure of the objects that exist in the problem domain. These are expressed using class, object and USECASE diagrams. But Dynamic modeling refers representing the object interactions during runtime. It is represented by sequence, activity, collaboration and statechart diagrams

[Read More Answers.](#)

### Question # 31

Why generalization is very strong in programming?

#### Answer:-

Even though Generalization satisfies Structural, Interface, Behaviour properties. It is mathematically very strong, as it is Antisymmetric and Transitive. Antisymmetric: employee is a person, but not all persons are employees. Mathematically all As' are B, but all Bs' not A.

[Read More Answers.](#)

### Question # 32

Whether unified method and unified modeling language are same or different?

#### Answer:-

Unified method is convergence of the Rumbaugh and Booch. Unified modeling lang. is the fusion of Rumbaugh, Booch and Jacobson as well as Bertrand Meyer (whose contribution is sequence diagram). Its' the superset of all the methodologies.

[Read More Answers.](#)

### Question # 33

Differentiate the class representation of Booch, Rumbaugh and UML in programming?

#### Answer:-

If you look at the class representation of Rumbaugh and UML, It is some what similar and both are very easy to draw.

[Read More Answers.](#)

### Question # 34

What do you mean by "SBI" of an object in programming?

#### Answer:-

SBI stands for State, Behavior and Identity. Since every object has the above three.



State:

It is just a value to the attribute of an object at a particular time.

Behaviour:

It describes the actions and their reactions of that object.

Identity:

An object has an identity that characterizes its own existence. The identity makes it possible to distinguish any object in an unambiguous way, and independently from its state.

[Read More Answers.](#)

Global Guideline - COM

## Computer Programming Most Popular Interview Topics.

- 1 : [PHP Frequently Asked Interview Questions and Answers Guide.](#)
- 2 : [C++ Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 3 : [C Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 4 : [Software engineering Frequently Asked Interview Questions and Answers Guide.](#)
- 5 : [Cobol Frequently Asked Interview Questions and Answers Guide.](#)
- 6 : [Visual Basic \(VB\) Frequently Asked Interview Questions and Answers Guide.](#)
- 7 : [Socket Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 8 : [Perl Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 9 : [VBA Frequently Asked Interview Questions and Answers Guide.](#)
- 10 : [OOP Frequently Asked Interview Questions and Answers Guide.](#)



## About Global Guideline.

**Global Guideline** is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of [Interview Questions with Answers](#) and much more. Learn the most common technologies at Global Guideline. We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

\* This PDF was generated from <https://GlobalGuideline.com> at **November 29th, 2023**

\* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and [e-mail us](#) we will fix it.

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.  
[www.facebook.com/InterviewQuestionsAnswers](http://www.facebook.com/InterviewQuestionsAnswers)

Follow us on Twitter for latest Jobs and interview preparation guides  
<https://twitter.com/InterviewGuide>

Best Of Luck.

Global Guideline Team  
<https://GlobalGuideline.com>  
[Info@globalguideline.com](mailto:Info@globalguideline.com)