

# Java Programming

## Questions & Answers

### Part-I

**Q Explain in brief concepts of Object Oriented Programming.**

Ans. Followings are concepts of OOP

- i. **Object** : Real world entity, concept, idea or abstraction is known as object. Every object has some features associated with it.
- ii. **Class**: Collection of similar objects is known as class or class is a user defined data type. Class is a template.
- iii. **Encapsulation** : Wrapping of data member and methods in class is known as encapsulation.
- iv. **Abstraction** : Abstraction is hiding complex information and showing only essential info.
- v. **Polymorphism**: Polymorphism means single interface multiple behavior. Polymorphism allows programmer to attach multiple behavior to single interface. i.e. method overloading.
- vi. **Inheritance**: Inheritance is process of creating new class using existing class. Newly created class is known as child class and existing class known as parent class.

**Q Write down features of Java or java's buzzwords.**

Ans. Followings are feature of java

- i. **Pure Object Oriented**: Java is a pure object oriented programming language. In java everything has to be written in class/interface. Nothing can be written outside of class/interface block.
- ii. **Platform Independent**: Program developed in java can be executed on any platform. Java uses both compiler and interpreter. Java program compiled into byte code which can be executed on any platform.
- iii. **Strongly Typed**: In java variable has to be declared first before using it.
- iv. **Robust**: Exception handling helps programmer to write robust application.
- v. **Garbage Collection**: Java reclaimed object's memory which are no longer in use. It's not programmer's responsibility to clean memory.
- vi. **Secure** : When a byte code come from network, byte code verifier first verify that byte code for malicious behavior. If byte code is a malicious program intend to harm host then program is not allowed to execute.
- vii. **Multithreaded**: In java application with multithreading capacity can be developed. A Multithreaded application can perform multiple task at a time.
- viii. **Interpreted**: Java's program is converted into a byte code instead of native code. This byte code is interpreted.

**Q Which primitive data types are available in Java?**

Ans. In java there are eight primitive data types are available

Data Type	Data size	Data size
byte	integer type of data	1Byte
short	integer type of data	2Byte
int	integer type of data:	4Byte
long	integer type of data	8Byte
float	Real number	4Byte
double	Real number	8Byte
boolean	true or false	1bit
char	Character	2Byte

**Q What is constructor?**

Ans. Constructor is used to initialize object. Constructor is special method which has same name as class name and has no return data type not even void. Constructor is called automatically when object of that class is created.

**Example:**

```
// Constructor Example
class example
{
    example()
    {
        System.out.println("Constructor Called");
    }
}
class demo
{
    public static void main(String arg[])
    {
        example ex=new example();
    }
}
```

**Q Explain constructor Overloading?**

Ans. When a class has multiple constructor with different arguments( number or type of arguments), this is known as constructor overloading. Constructor overloading is example of compile time polymorphism.

**Example:**

```
// Constructor Overloading Example
class example
{
    example()
    {
        System.out.println("Constructor without argument Called");
    }
}
```

```

example(int a)
{
    System.out.println("Constructor with argument called");
}
}
class demo
{
    public static void main(String arg[])
    {
        // non-parameterized constructor get called.
        example ex=new example();
        // parameterized constructor get called.
        example ex1=new example(10);
    }
}

```

**Q Explain automatic type promotion.**

Ans. Java perform arithmetic operations on equal data type. For example in a expression if there are two operands of integer and float type respectively then java automatically promote int to float. This is also known as automatic type promotion.

**NOTE:** Java won't allow arithmetic operations on byte, short, char data type. If there is any variable from these data types then those variable automatically promoted to integer data type variable. Then operation will be performed.

**Q Explain Implicit and Explicit Type casting.**

Ans. Type casting is process of assigning value of different data type to variable of different data type. If value of small data size is assigned to variable of large data size then it is known as **implicit type casting**.

```

// Implicit Type Casting
int a;
byte b=120;
// byte value (1 byte) to int variable (4 byte)
a=b;

```

When large data size value is assigned to small data size variable it is known as **explicit type casting**.

Explicit type casting is done by writing following syntax  
variable=(Type) value;

```

// Explicit Type Casting
int a=200;
byte b;
// int variable (4 byte) to byte variable
b=(byte)a;

```

**Q What is abstract class?**

Ans. Abstract class are classes whose object can't be created or class which can't be instantiated. Abstract class has to be declared with *abstract* keyword. Abstract class has one or more than one abstract method.

Abstract method is a method with abstract keyword and without definition.

**Q Why public, static keywords are used with main method in java?**

Ans. Java compiler(javac) is stored in "C:\Program Files\Java\Jdk1.8\bin" directory. Our program might be stored anywhere so public keyword is used so that javac[java compiler] can call main() method. A method with public keyword can be called from anywhere.

Static keyword allows calling of main method without creating object of class. As we know in order to call method of a class we need to create a object and using that object methods of class can be called. Another way is to declare method as static. Static method can be called by using class name, That's why static keyword is used so that main method can be executed without creating object.

**Q What is command line arguments in Java?**

Ans. Main method in java has String array as an argument, which allows user to provide input at execution time. All information provided by user stored in argument of main method. These argument helps in creating a command.

**Q What is method overloading?**

Ans. Method overloading is defining multiple methods with same name but with different arguments(either types of arguments or number of arguments). All these methods has to be in same class.

*NOTE: Method overloading can't be achieved by changing only return data type.*

Example:

```
// Method Overloading
class mth
{
    void show()
    {
        System.out.println("First method is called");
    }
    void show (int a)
    {
        System.out.println("Second method is called");
    }
}
```

**Q What is method overriding?**

Ans. When child class re-define a method which is available in parent class. Method in parent and child has same return-type and same signature. In method overriding different classes which are in parent-child relationship, have same methods(same name , same arguments and same return type).

```
// Method Overriding
class A
{
    void show()
    {
        System.out.println("Parent class show method");
    }
}
```

}

```
// Child class re-defining parent class' method show()
class B extends A
{
    void show ()
    {
        System.out.println("Child class show method");
    }
}
```

[http://www.jgyan.com/inheritance\\_method\\_overriding.php](http://www.jgyan.com/inheritance_method_overriding.php)

### Q What are the usage of final keyword?

Ans. final keyword can be used with class, method or variable.

- i. When final keyword is used with class then that class become non-inheritable. It means that class can't be inherited.
- ii. if final keyword is used with method then that method can't be redefined in subclass of class which has final method.
- iii. If final keyword is used with variable then that variable become constant.

### Q What are differences between checked and unchecked exceptions?

Ans. Checked exceptions are already identified in java. These exception are child of Exception class. Checked exception has to be handled. If program is generating these exception then it's programmer responsibility to handle it or explicitly specify that program can generate exception.

i.e. IOException

Unchecked exception are not checked during compilation of program. If there is any unchecked exception in the program then it won't create any problem for compilation. i.e.

ArrayIndexOutOfBoundsException

[link: http://www.jgyan.com/exception\\_type.php](http://www.jgyan.com/exception_type.php)

### Q How throw and throws are different to each other?

Ans. throw keyword is used to generate exception manually.

i.e.

```
throw new ArithmeticException();
```

throws keyword is used in situation where code can generate checked exception and programmer don't want to handle it. In this case programmer simply specify using throws keyword that it can generate specified exception.

i.e.

```
void show() throws IOException
{
}
```

[http://www.jgyan.com/exception\\_throws.php](http://www.jgyan.com/exception_throws.php)

**Q What is exception?**

Ans. Exception is run time error which cause abnormal termination of program.

**Q How to handle an exception?**

Ans. try-catch is used to handle exceptions. Piece of code which can possibly generate an exception is enclosed in try block. On occurrence of exception try block send control to catch block. So exception handling code has to be written in catch block. A try block can have multiple catch block for different-2 exceptions.

Example:

```
// Exception handling
try
{
    code which can generate exception
}

catch( <exception-type>)
{
    code which will handle exception
}
```

**Q What is difference between final, finally and finalize in Java?**

Ans. final keyword is used for following

- For making a constant.
- For making a class non-inheritable
- For making a method non-overridden

finally block is used in exception handling. Code written inside finally will be executed always whether exception occur or not.

In java object which are no longer in use, is cleaned from memory by garbage collector. If before cleaning object we want to process something then that code will be written inside finalize method. Finalize method is executed before garbage collection of object.

**Q What is inheritance?**

Ans. Inheritance is a feature which allows a programmer to create new class by using existing. Inheritance avoid rewriting of existing code for creating new class. Newly create class is known as child class and existing class is known as parent class. Child class inherit all properties of parent class except constructor. For making a child class extends keyword is used.

```
// Parent class
class A
{
}

// Child Class
class B extends A
{
}
```

**Q What is use of super keyword?**

Ans. super keyword is used in child class to refer parent class. Following are the use of super keyword

- i. When parent and child class both have same name variable then super keyword is used in child class to access parent class' member.

```
class A
{
  int x;
}

class B extends A
{
  int x;
  void display()
  {
    System.out.println("Class B member:"+x);
    System.out.println("Class A member:"+super.x);
  }
}
```

- ii. When object of child class is created, it will make a call to parent class' default constructor before calling it's own constructor. super keyword can be used to call parent class' parameterized constructor when object of child class is created.

```
class A
{
  A() { System.out.println("default constructor");}
  A(int a) { System.out.println("parameterized constructor");}
}

class B extends A
{
  B()
  {
    // following statement will call parent class parameterized constructor instead of default
    constructor
    super(12);
  }
}
```

**Q What is use of this keyword?**

Ans. this keyword is used to refer current object. It is used for two purpose

- i. To differentiate between instance's data member and formal arguments.
- ii. To call another constructor explicitly in class.

*For More Details--><http://www.jgyan.com/this%20keyword.php>*

**Q What is interface?**

Ans. Interface is like class which contain only abstract methods or constants. Method declared inside interface treated as abstract method internally. Only constant can be declared in interface using final keyword. Every declared method in interface is public and abstract.

```
// Interface example
interface one
{
    final int a=12;
    void show();
    void perform();
}
```

www.jgyan.com