1. What is java and list the features?

\*Java is Programming Language

\*Generally, Java is a simple, robust and secure programming language.

\*Here are the most important features of Java:

**1. Java is Simple:**

The Java programming language is easy to learn. Java code is easy to read and write.

**2. Java is Familiar:**

Java is similar to C/C++ but it removes the drawbacks and complexities of C/C++ like pointers and multiple inheritances. So if you have background in C/C++, you will find Java familiar and easy to learn.

**3. Java is an Object-Oriented programming language:**

Unlike C++ which is semi object-oriented, Java is a fully object-oriented programming language. It has all OOP features such as [abstraction](https://www.codejava.net/java-core/the-java-language/what-is-abstraction-in-java-the-why-and-the-truth), [encapsulation](https://www.codejava.net/java-core/the-java-language/what-is-encapsulation-in-java-the-what-why-and-how), [inheritance](https://www.codejava.net/java-core/the-java-language/what-is-inheritance-in-java-the-what-why-and-how) and [polymorphism](https://www.codejava.net/java-core/the-java-language/what-is-polymorphism-in-java-the-what-how-and-why).

**4. Java supports Functional programming:**

Since Java SE version 8 (JDK 8), Java is updated with functional programming feature like functional interfaces and Lambda Expressions. This increases the flexibility of Java.

**5. Java is Robust:**

With automatic garbage collection and simple memory management model (no pointers like C/C++), plus language features like [generics](https://www.codejava.net/java-core/collections/what-are-generics-in-java), [try-with-resources](https://www.codejava.net/java-core/the-java-language/using-try-with-resources-examples-java-7),… Java guides programmer toward reliable programming habits for creating highly reliable applications.

**6. Java is Secure:**

The Java platform is designed with security features built into the language and runtime system such as static type-checking at compile time and runtime checking (security manager), which let you creating applications that can’t be invaded from outside. You never hear about viruses attacking Java applications.

**7. Java is High Performance:**

Java code is compiled into bytecode which is highly optimized by the Java compiler, so that the Java virtual machine (JVM) can execute Java applications at full speed. In addition, compute-intensive code can be re-written in native code and interfaced with Java platform via *Java Native Interface* (JNI) thus improve the performance.

**8. Java is Multithreaded:**

The Java platform is designed with multithreading capabilities built into the language. That means you can build applications with many concurrent threads of activity, resulting in highly interactive and responsive applications.

**9. Java is Platform Independence:**

Java code is compiled into intermediate format (bytecode), which can be executed on any systems for which Java virtual machine is ported. That means you can write a Java program once and run it on Windows, Mac, Linux or Solaris without re-compiling. Thus the slogan “*Write once, run anywhere*” of Java.

Besides the above features, programmers can benefit from a strong and vibrant Java ecosystem:

* Java is powered by Oracle - one of the leaders in the industry. Java also gets enormous support from big technology companies like IBM, Google, Redhat,… so it has been always evolving over the years.
* There are a lot of open source libraries which you can choose for building your applications.
* There are many superior tools and IDEs that makes your Java development easier.
* There are many frameworks that help you build highly reliable applications quickly.
* The community around Java technology is very big and mature, so that you can get support easily.

That’s why Java has been ranking as #1 programming language in the world by popularity, according to [statistics by Tiobe Index](http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html).

So congratulate you on choosing Java as the programming language for developing awesome applications.

2. Why Java?

\* Because java is Platform independent language ,the .class file can run in any Operating system, in that operating system JRE (Java Runtime Environment)must be installed,

\* .class file dependent on JRE, independent on Operating System.

\* The JRE is developed separately for each operating system

3. What is the difference between JDK, JRE, JVM and JIT?

* **Java Virtual Machine (JVM)** is an abstract computing machine.
* **Java Runtime Environment (JRE)** is an implementation of the JVM.
* **Java Development Kit (JDK)**contains JRE along with various development tools like Java libraries, Java source compilers, Java debuggers, bundling and deployment tools.
* **Just In Time compiler (JIT)** is runs after the program has started executing, on the fly. It has access to runtime information and makes optimizations of the code for better performance.

4. What is a variable?

\*Variable is name of reserved memory location

\*Variables as container

5. Why variable?

\*To change the data

\*When you allowed to change anything is called variable

Example int i=5;

Int 🡪 int is datatype

i 🡪 I is the name of the variable

5 🡪is the value

6. What is a data type and list them?

7. List the size and default value of each data type?

Data type Keyword Size(bit) Size(byte) Default value (for fields)

Byte byte 8 1 0

Short short 16 2 0

Integer int 32 4 0

Long long 64 8 0L

Float float 64 8 0.0f

Double double 64 8 0.0d

Character char 16 2 ‘\u0000’

Boolean boolean 0 0 false

8. What is a method?

\*It contains the lines of statements which performs some operations

9. What is difference between Arguments and parameters?

A parameter is a variable in a method definition. When a method is called, the arguments are the data you pass into the method's parameters.

public void MyMethod(string myParam) { }

...

string myArg1 = "this is my argument";

myClass.MyMethod(myArg1);

**Parameter** is variable in the declaration of function.

**Argument** is the actual value of this variable that gets passed to function.

10.Can you write a method inside a method?

11.What is method overloading?

Examples

Search in FB

* Search in Phone(contacts)
* Wending machine
* Gun

Method Overloading:-In class we have two method with same name but different number of parameter but type of parameter should be change.

Imp:Real time example for method Overloading in java is Hibernate configuration.configure();//it will look for default method "hibernate.cfg.xml"

If we are changing the overloaded method, we have mention overloaded method configuration.configure("mysql.cfg.xml");

12. What is a class and objects?

What is class?

* Class is like blueprint it defines the structure of the object and behaviour of the object

What is object?

* Object is know something and object does something

ex:logo of your phone,color of your phone ,text to someone or call to someone

ex:class Building

{

int floors;

public void provideAcc()

{

//code

}

}

* Class: A class is a template/blue print from which individual objects are created.
* A class is a template/blue print that describes the data and behaviour of an object of the class.
* Object: object is an instance of a class. Object have states and behaviours.
* Object is a software bundle of variables and related methods.
* Object class: object class is default super most class in java.

Objects basically has two things

1) What object knows-->i.e variable

2) What object does-->i.e method

* If you want to store number,if you want to store your name,if you want to store your personal information i.e you required variable
* If you want to save something.if you want to execute something,if you have to print something you need method
* Objects are given by JVM but it requires design i.e given by Class
* You required object to work with you create objects you need design and that design is given by Class.
* Objects are real instances and class are imaginary parts
* Objects has memory i.e given by JVM
* If you want to work with class it requires objects
* Creating Object in java.
* In java object created inside heap memory and it requires new keyword and how much memory is required for that object,i.e given by constructor
* Object inside the heap memory is called as Instant.

13.What is constructor?

* Constructor is a member method
* Constructor name same as class name
* Constructor is used to provide memory for an object
* constructor never return anything

What is default constructor?

1. What is parameterised constructor and what is a use of them?
2. What is constructor overloading?
3. What is the use of “this” and “super”
4. What is Inheritance? And list the type of inheritance.

Examples

* Messenger and FB
* Mobiles
* TV’s

1. Why java does not support multiple inheritance
2. What is method overriding? And what are the rules for method overriding

Examples

* Mouse operation
* Car game

1. What is Abstraction?

Examples

* Camera
* ATM
* Bar code scanner

1. Difference between interface and abstract class
2. What is a package?
3. What is a use of import? When to we need to import? Which is the default package gets imported.
4. What are accesses specifies? And list them.
5. What is Encapsulation?
6. What is meant by java bean
7. What is Polymorphism?
8. What is the difference between String, String Buffer and String Builder
9. Explain Object class
10. What is casting? Explain
11. What is Scanner? And list Scanner class methods
12. What is meant by wrapper class
13. What is exception? And how to handle the Exception
14. What is meant by Throw able
15. Difference between Error and Exception
16. What are checked and unchecked exception
17. What is the difference between finally, final and finalize.
18. What is the difference between throw and throws
19. What is collection frame work
20. Difference between Array List and Linked List
21. Difference between Array List and Vector
22. Difference between List and Set
23. List the methods of Collection

* Write a program to count the sum of all the degits in a number.

Example : **input 5199**........ **OUTPUT 24**

* Write a program to find the number is palindrome or not
* Write a program to find the greatest degit in the number
* Write a program to find the least degit in a number
* Write a program to print the number from 1 - 100 that are devesible by only 5 but not by 10 and 3
* What is the output

**int** i = 0;

**for**( ;i!=0 ;)

{

System.*out*.println("hi...!");

i++;

}

* What will be the output

**int** i = 1;

**while**(i>0)

{

System.*out*.println(i);

i++;

}

* **Write a program to multiply 2 numbers and divide that value by 2 (using methods)**
* **What is a operator?**
* **List Arithmetic , relational, unary and Logical Operators**
* **What is the output of**

**class Program1**

**{**

**public static void main(String[] args)**

**{**

**int a=5;**

**int b = --a + --a + --a + a++;**

**System.*out*.println(b);**

**System.*out*.println(a);**

**}**

**}**

* **What is the output of**

**class Program2**

**{**

**public static void main(String[] args)**

**{**

**double d = 10.5;**

**float f = d;**

**System.*out*.println(f);**

**}**

**}**

* **Which will compile and which will not**

1. **double &ab = 89;**
2. **float f =10.5f;**
3. **int $i =90;**
4. **int i = (int)10.5;**
5. **boolean b =True;**
6. **char c = a;**
7. **char num = 7878;**
8. **char ch = "A";**

* **class Program1**

**{**

**static void main(String[] args)**

**{**

**System.*out*.println("Hi first Program");**

**}**

**}**

* **class Program2**

**{**

**public static void main(String[] args)**

**{**

**int a = 11;**

**int b = 2;**

**double ans = a/(double)b;**

**System.*out*.println(ans);**

**}**

**}**

* **class Program3**

**{**

**public static void main(String[] args)**

**{**

**for(int i =0;i<10;)**

**{**

**i--;**

**System.*out*.println(i);**

**}**

**}**

**}**

* **class Program4**

**{**

**public static void main(String[] args)**

**{**

**while(True)**

**{**

**System.*out*.println("HI i am in while loop");**

**}**

**}**

**}**

* **class Program5**

**{**

**public static void main(String[] args)**

**{**

**int a = 3;**

**if(a==5)**

**{**

**a--;**

**}**

**else if(a==4)**

**{**

**System.*out*.println("a is 4");**

**}**

**else**

**{**

**System.*out*.println("a have some value");**

**}**

**System.*out*.println("a is 5");**

**}**

**}**

* **class Program6**

**{**

**public static void main(String[] args)**

**{**

**for(int i=0;i<5;i++)**

**{**

**while(i>1)**

**{**

**System.*out*.println(i);**

**}**

**}**

**}**

**}**

* **class Program7**

**{**

**public static void main(String[] args)**

**{**

**String str = "I wish u all good luck";**

**String s[]=str.split(" ");**

**System.*out*.println(s.length);**

**}**

**}**

* **class Program8**

**{**

**public static void main(String[] args)**

**{**

**int i =0;**

**while(i<5)**

**{**

**i++;**

**System.*out*.println(i);**

**}**

**}**

**}**

* **class Program9**

**{**

**public static void main(String[] args)**

**{**

**int i =0;**

**while(i<5)**

**{**

**if(i==2)**

**{**

**i = 5;**

**}**

**System.*out*.println(i);**

**i++;**

**}**

**}**

**}**

* **class Program10**

**{**

**public static void main(String[] args)**

**{**

***add*(10,40);**

**}**

**static int add(int a , int b)**

**{**

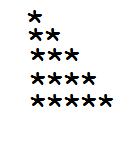
**int sum = a + b;**

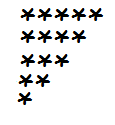
**System.*out*.println(sum);**

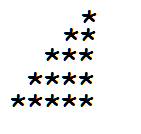
**}**

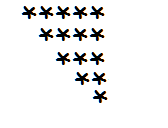
**}**

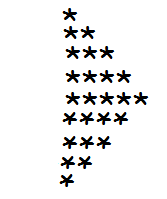
1. **Write a program to pass the name and age of a person into a method**
2. **Write a program to sort the number is ascending order**
3. **Write a program to find the largest and smallest elements of array**
4. **Write a program to sort the characters in a array**
5. **Write a program to find the even number in a array, and also find the largest even number of the array**
6. **Write a program to search the given number in a array**
7. **Write a program to swap the contents of two arrays**
8. **Write a program to add two numbers without using addetion opperator**
9. **Write a program to print pyramid and diamond pattern**
10. **Write a program to print all odd numbers that are divisible by 7 from 1-100**
11. **Write a program to find weather both the 2-dimention matrix are equal or not**
12. **Write a program to count the even digits in a number**
13. **Write a program to show method over loading and method over riding**





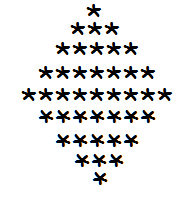


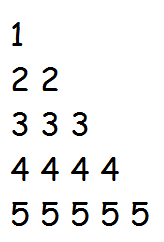


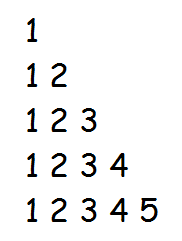


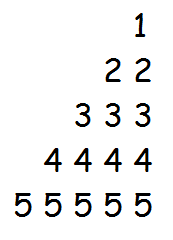


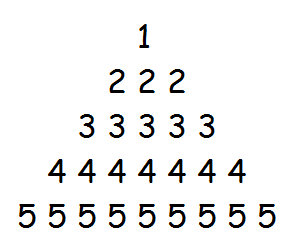


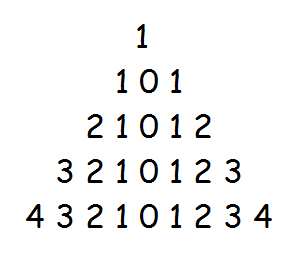


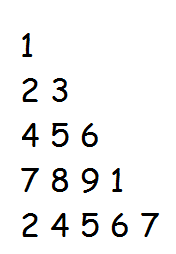












1. Write a program to print the tables from 1 to 10

(eg : 1 \* 1 =1          2 \* 1= 2      3 \* 1= 3……….so on)

1. Create 2 class, having 4 states and 4 behavior each and try to accesses them from other class by creating object
2. Write a program to show constructor over loading
3. Write a program to count the words in a string or in a line and reverse it

(eg: “Hi im vikas from Hassan”) outPut must be **5-words** and also reverse it has **“nassaH morf sakiv mi iH”**

1. Store the elements into array and retrieve them using while loop, while retriving  you must print the squares of the input numbers

(eg: 2 3 4 5 6 output should be     4 9 16 25 36)

8.Write a program to print out put has

(eg: 1 is odd, 2 is even, 3 is devisible by 3 and it is odd, 4 is even, 5 is odd, 6 is devisible by 3 and it is even…….)