1.What is java?

Java is a popular programming language.

2.used for?

* Mobile applications (specially Android apps)
* Desktop applications
* Web applications
* Web servers and application servers
* Games
* Database connection
* And much, much more!

3.why use java?

* It is easy to learn and simple to use
* It is open-source and free
* It is secure, fast and powerful
* Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.)

4.java compiler?

Java has a compiler, it will scan the program and if there are any error then it will throw an error, if no errors then convertible in to high level program to machine readable language.

5.byte code?

Compiler compile our program to machine readable format and generates a byte code/class file. -🡪 .class

6.jvm?

Java virtual machine(windows)

It would look byte code/class file then it will execute the program. JVM generates an output.

7.java code compile ?

Javac hello.java

8.java file execution?

Java hello

9.System.out.println()?

It will print the code

10.diff b/w print() and println() ?

Print()🡪it does not insert a new line at the end of the output:\

Println()🡪it will insert a new line at the end of the output

11.java comments?(it will not execute)

1.single line comments(//)

Single-line comments start with two forward slashes (//).

2.multi line comments( /\* \*/)

12.java variables?

To store the data values into the variables.

🡪variable value will stored in the memory address.. it will automatically stored.

13Data types?

Data types can specify the different size and values that can be stores in the variable.

**2 types:**

**Primitive data type**—byte, short, int, long, float, double, char, Boolean

**Non primitive data type**—class, interfaces, string, array

14.Default value and default size for data types? 1byte=8bit

**Datatype default value size**

|  |  |  |  |
| --- | --- | --- | --- |
| Boolean | false | 1 bit | True/ false |
| Char | ‘\u000  0’ | 2 bytes | Single char/ascii 0 to 65535 |
| Byte | 0 | 1 byte | -123 to 123 |
| Short | 0 | 2bytes | -32k to 32k |
| Int | 0 | 4 bytes | -2^31 to (2^31)-1 |
| Long | 0L | 8 bytes | -2^62 plus |
| Float | 0.0f | 4 bytes | 6 to 7 decimal |
| double | 0.0d | 8 bytes | 15 decimal dig |

15.primitive number types?

**1.integer types**: byte, short, int , long

**2.floating point types** : float , double

16.Boolean?

Boolean is a data type. It can have one of two values.

Like🡪 yes/no, true/false

17.character?

To store a single character/ASCII values.

It will represent with in the single quote

18.non-primitive data type?

Non-primitive data types are called **Reference type.** Because they refer to object.

19.Difference b/w primitive and non primitive data type?

|  |  |
| --- | --- |
| **primitive** | **Non primitive** |
| Always has a value | null |
| Starts with lowercase | Starts with uppercase |
| Size is It depends on data type | All same size |

20.java type casting?

When you assign a value of one primitive data type to another type.

1.**widening**—byte-short-char-int-long-float-double

2.**narrowing**—double-float-long-int-char-short-byte

21.java operators?

It used to perform certain operations on variables and values.

1.arithmetic operator--+, - ,\* , / , %

2.assignment operator

3.logical operator--&&, || , !

4.relational operators-🡪<, > , <= , >=, ==, !=

5.ternory operator.-->(n1>n2) ? true:false;

22.what is if..else?

If you want to validate a condition.

23.what is if elseif?

If you want to validate a multiple condition , execute the different blocks of code depends on the conditions.

24.for loop?

Same task repeat again and again, until reach the condition.

25.for-each loop?

It is used to exclusively to loop through elements in an array.

26.while loop?

First you have to check the condition ,if the condition is true ,if block will be executed. If the condition is false got else block.

For ex..i=11 condiiton(i<=10)..it will displayed nothing

27.do while?

First statement will be printed first and the you have to check the condition.

Ex: i=11; condition(i<=10)—it will printed the I value is 11 because the statement will printed first

28.break statement?

🡪break statement is used to terminate the loop if the specify condition is met.

🡪also used to exit the control of a program.

29.continue statement?

🡪continue statement will skip the iteration and go to the beginning of the next iteration.

30.switch?

Fast and compare to if else conditions.

31.what is class?

Class is a **blueprint** for creating an object.

Class & objects along with attributes and methods

Ex:

Object🡪car

Attributes🡪weight and color

Method🡪drive() and brake()

32.what is object? And why we create an object?

When we create an object to access the class.

🡪object is used to access the member elements.

**Ex:**

Fruits f=new fruits();

New is a keyword to create a new class name

33.attributes?

🡪modify the attributes

🡪override the existing value

34.what is Method?

Methods are derived within the class. Methods are used to perform certain actions.

**Syntax**:

Access modifier return type method name

Public void addTwonum()

35.what is code reusability?

Same code will be reusable for all different parameters.

36.what is constructor?

Constructor is called whenever object is created for a class.

🡪constructor name is same as class name

🡪it does not have any access modifiers. If we use public it will run

🡪without mention any parameter & empty constructor—it will run

37.what is an array?

Array is used to store a multiple values in a single variable, instead of creating separate variables for each value.

🡪array index starts from 0

🡪3 ways of creating an array

38.types of array?

🡪single dimensional array

🡪multi dimensional array

39.what is two dimensional array?

two dimensional array is array of arrays.

For example:

We consider a tabular form ..table with rows and columns

40.Inheritance(oops concepts)

What is inheritance?

Which one object acquires all behaviours and properties of parent object is called inheritance.

🡪there is a relationship parent &child

41.Why use inheritance in java?

🡪method overriding(run time error can be achieved)

🡪code reusability

42.Types of inheritance?

* Single inheritance
* Multilevel inheritance
* Hierarchical inheritance
* Multiple inheritance

43.Single inheritance?

Where there is class inherited another class is called single inheritance.

🡪class B inherit the behaviours and properties of class A

44.Multilevel inheritance?(class A,B,C)

Where there is chain of inheritance is called multilevel inheritance

🡪class C is inherited the behaviours and properties of of class B, class B is B inherit the behaviours and properties of class A

45.Hierarchical inheritance?(class A,B,C)

Where there is two or more classes inherited in single class is called hierarchical inheritance.

🡪class B inherit the behaviours and properties of class A

🡪 class C inherit the behaviours and properties of class A

46.Multiple inheritance?

Multiple inheritance is not supported in java.

🡪for example,

Class C extends A,B🡪compiler get confused which class will be called,

🡪it is a compile time error

47.what is access modifiers?

To define the scope.

**1.private**

🡪access within the same class only.

Compile time error

**2.default(I wont mention default infront of the variable &method)**

🡪access within the same class

🡪access within the same package

**3.Protected(inside the public class I will write the code)**

🡪access within the same class

🡪access within the same package

🡪access outside the package(derived class only)

**4.public(inside the public class I will write the code)**

🡪access within the same class

🡪access within the same package

🡪access outside the package(sub class)

🡪access outside the package(global)

48.Super keyword?

Super keyword is used to refer an immediate parent class object.

**Usage:**

**1.**super is used to invoke a immediate parent class instance variable

2.super is used to invoke a immediate parent class method

3.super() is used to invoke an immediate parent class constructor.

🡪if we don’t call the parent class constructor explicitly ,then child class would call the parent class constructor implicitly

49.Polymorphism?

Under 2..

1.method overloading(compile time error)

2.method overriding(run time error)

50.What is method overloading?

When class has same method name and different no.of arguments or datatype. It is called method overloading.

51.What is method overriding?(public class extends)

When child class has same method name as in the parent class method, without change in the method signature ,then it is called method overriding.

🡪if we run the program , child class will be executed, if there is no method in the child class then only parent class will be executed.

3 rules:

🡪method must have same name in the parent class

🡪method must have same parameter in the parent class

🡪there must be a relationship between parent and child class(inheritance)

52.Final keyword?

It is used to restrict the user.

Final keyword along with class, methods, variable

🡪if I declared a variable as a final ,I cannot change the value

🡪if I declared a method as final, the method can’t override in the sub class. But method will be inherited.

🡪if I declare a class as final ,I cannot inherite the class

53.This keyword?

This is a reference variable and it is used to refer the current object.

**3 usage:**

🡪this can be used to invoke current class insatance variable

1.if I declared a local variables and instance variable names are same, then the output will be “null”.

2.overcome this output I will use this keyword to distinguish local variable and instance variable.

3.if I declared local variables and instance variables are different, then we no need to use this keyword

🡪this can be used to invoke current class method(in current class we ‘ve two method)

🡪this() can be used to invoke current class constructor.( in current class we ‘ve two constructor)

54.what is string?

String is a sequence of a characters.

🡪surrounded with double quotes

**2 ways to create string**:

1.literal

2.creating new

55.string pool memory?

Most of the Important things are stored in string, so they thought of creating a separate memory is called string pool memory or string constant pool.

56.Heap memory?

Values would be stored in different memory location but for string, if the values are same it will allocate single memory location.

57.why strings are immutable in java?

🡪we can’t destroy the object.

🡪if the values are assigned to the string object, then the value cannot be changed.

For ex:

String s1=”hello”;

s1.concat(“world”); directly we cant concat the string. We will store the concatenation string in existing object/new object.

Like this,  
String s2=s1.cancat(“world”);

58.string methods?

**1.To uppercase**

It returns a string in uppercase letter

**2.to lowercase**

It returns a string in lowercase letter

**3.charAt()**

The method returns the char value of the given index number

**4.compareTo()**

The method compares the given string to the current string lexigropically. It returns positive number, negative number , and 0(if the string is equal)

**5.concat()🡪immutable**

The method combines a given string at the end of the string, it returns a combined string

**6.contains()**

The method searches the sequence of character in this string, it return true if the sequence of character is found, otherwise it returns false

**7.endswith()**

Method checks if the string is ends with the specify suffix.

**8.startswith()**

Method checks if the string is starts with the given prefix

**9.length()**

Method is used to find the length of the string

**10.tochararray()**

The method converts the string to character array. It returns a newly created character array.

**11.indexOf()**

The method returns a position of the first occurrence of a given character or string

**12.replace()**

The method returns a string replacing the all old char to new character.

**13.trim()**

**it** eliminates leading and trailing whitespaces, the Unicode value of space character is \u0020

**14.substring()**

It returns a part of a string

2 parameters:

Beginning index—inclusive it starts from 0

End index ----exclusive it starts from 1

**15.split()**

**The** method used to split or separate the given string into the multiple substring separated by delimiters

**16.isEmpty()**

The method check if the current input string is empty or not. It will return a Boolean value

**17.equals()**

**The method**  compares the two strings. If two strings are equal it returns true. Or it returns false

**18.equalsignorecase()**

**The method compares the**  two strings, avoid the case consideration

**19.join()**

The method is to joined the string with the given delimiters

**20.lastindexof()**

The method is returns a last index value of a given character or substring.

🡪if the character or substring is not found it returns a negative number

**21.getBytes()**

**It**  converts each character in a string into bytes using default encoding platform..it will stored is bytes array..it will return a ascii value

**22.valueof()**

**In converts** all different types of value into string. Like, into to string, char to string, Boolean to string etc.

59.String buffer?

String buffer is a class and it is used to create a mutable string

3ways:

1.StringBuffer()

2. StringBuffer(String str)

3. StringBuffer(int length)

60.String builder?

String builder is a class and it is used to create a mutable string

3ways:

1.StringBuilder()

2. StringBuilder(String str)

3. StringBuilder(int length)

61.Stringbuilder and stringbuffer?

It is a mutable version of string. We can change the value

62.diff b/w string buffer and string builder?

|  |  |
| --- | --- |
| String buffer is a **thread safe** .if two thread can’t call the methods of string buffer simultaneously. | String builder is **not thread safe** ,if two threads can call the methods of string builder simultaneously |
| Less efficient | More efficient than string buffer |

63.String buffer & builder methods?

* Append
* Insert
* Delete
* Reverse
* Substring
* charAt
* ensure capacity
* length
* replace

64.Static keyword?

Static is a keyword, static means cannot change the value.

65.static variable?

If I declared a variable as static , I cannot change the value. This is the class level variable.

If I access the static variable using this

Classname . variablename=” “;

🡪static is belongs to class level, not for objects

🡪static doesn’t have any memory, it will own a class memory.

65.static variable and non static variable?

Static variable🡪 static String name;

Non static variable: int rollno; String gender;

This is also called instance variable.

66.static method?

Inside the static method, we can only access the static member elements.

🡪we cannot access the instance variables, because the compiler would get confused which object would get call.

So we passing an arguments, and then we will access the elements.

67.static block?

Static block is used to initialize a value of static variables.

🡪it will executed first

68.instance block?

Instance block is created inside the class but outside of any method.

69.What is encapsulation?

Process of hiding data from the user.

🡪combining both variables and methods together as single unit

🡪it is used to improve security of a code

70.steps to achieve encapsulation?

🡪all the variables and methods as a private.

🡪provide get and set method to modify and view the variable values.

71.what is abstraction?

Process of hiding implementation details from the user.

Example” ATM

72. two ways to achieve abstraction?

🡪abstract class

🡪abstract interface

73.abstract class full details?

* If I’m including the abstract infront of the class is called abstract class.
* If I’m not defining the abstract method inside the abstract class , so that becomes an abstract method.
* Inside the abstract class we can have the normal methods also.
* Class which extends the abstract class , it will forcefully write the logic for abstract method.
* So the normal class implement the unimplemented methods.
* Here , I’m defining the unimplemented methods.
* We can’t create an object for abstract class, the reason is abstract class have unimplemented methods.
* We can create an object for implemented(extends) class.
* We ‘ve unimplemented method calls abstract method.
* We will write the abstract method logic only in normal class
* We can create as many classes.
* Abstract class can also have variables.

🡪final variables, normal variables, static variables, non standard variables.

* abstract class can also have a constructor

74.what is interface?

Interface is blueprint of a class. It has static constants and abstract methods.

🡪we can’t create an object for interface

🡪we can create an object only for implemented class

🡪 interface-implements

🡪all the variables inside the interface is called static & default variables.

🡪all the method inside the interface would get call abstract method.

🡪if I’m not include a abstract keyword infront of the method, the compiler will consider the normal method as abstract method

75.why use interface?

🡪 interface can achieve abstraction. Interface can have only the abstract methods, not in the method body.

🡪it can support multiple inheritance in java

76.multiple inheritance (we solve it using interface)

* class can implements 2 interfaces simultaneously.

🡪if I declared a different method name in 2 interfaces, then class can implement the unimplemented methods

🡪if I declared a same method name in 2 interfaces, it will pic specific method name. and write the logic

* Interface can extend the another interface

77.static & default interface?

🡪if I declared a method is default , we will call the method using class object

🡪if I declared method as static , we will access the method using classsname . methodname.

Interface & class 🡪implements

Class & class 🡪 extends

Interface & interface 🡪 extends

78.Packages?

Collection of similar classes.

🡪we are grouping the similar classes, interface collections and sub packages.

**Syntax:**

Package packagename;

79. Two type of packages?

🡪user defined packages(if im writing a code to specific)

🡪built in packages( comes handy when installed java)

80.import?

🡪to access the packages using import statement.

🡪To import the packages with no changes in the directory structure.

81. 3 ways to access the packages?

🡪to import the specific class from packages.

🡪to import all the classes from the packages

🡪using fully qualified name instead of importing the packages.

82.built in packages?

Import java.io.File;

Import java.io.\*;

83.File operations?

🡪create a file

**2 ways to creating a file**

1->buffer reader

2->scanner class

🡪write a file

🡪read a file

🡪delete a file

84. what is collection?

85.collection framework?

86.why we need this collection?

87.disadvantages of array?

88.overcome this using collection?

89.collection architecture?

90.difference b/w list and set?

91.map interface?

92. methods that are there in the collection and list?

|  |  |
| --- | --- |
| collection | list |
| 10 methods | 6 methods |

**Linkedlist and arraylist can have all the above methods.**

93.Linkedlist

🡪linkedlist can also have the methods of arraylist has

|  |
| --- |
| Methods specific into the Linkedlist 6 methods |
| Maintain the insertions order |
| Allow duplicates |
| Allow null values |

94.operations of list?

2 operations.

95.we should know when to use what?(linkedlist and arraylist)

96.arraylist structure and linkedlist ,queue,stack structure?

Linkedlist -6 methods

97.set(interface)?

3 types.

hashSet🡪8 points 🡪9 methods

linkedhashset->

treeset->

**default memory and fill factor?**

98.compare arraylist default memory and hashset memory?

99.queue?

2 classes;

🡪Linkedlist

🡪Priority queue

100.diff b/w linkedlist and priority queue?

101. common 6 methods for both linkedlist and priority queue.

102.map interface?

4 classes

103.hashmap

🡪6 points

🡪11 methods

🡪entry 🡪3 methods

104.Exception handling?

105.advantage of exception handling with example?

106.heirarchy of exception?

107.diff b/w checked, unchecked exception and error?

108.types of java exception?

109. java exception keywords?

* Try
* Catch
* Finally
* Throw
* Throws

110.user defined exception?(how to create exception on my own?)

3 rules

111.throw keyword 🡪2 uses?

112.Printstacktrace?

113.ex.getmessage?

114.diff b/w try..catch and throws?

115.Thread?

116.Sleep?

117.Multiple catch block?

118.How are the keywords **final, finally and finalize** different from each other?

finalize() method, garbage collector immediately destroys the Java object.

119. Difference between **throw, throws & Throwable** in java ?

120.difference b/w hashTable, Linkedhashmap and TreMap?

121.**which collections are thread safe?**

* HashTable
* Stack
* Vector

### 122. **Can you add a null element into a HashMap or Hashtable?**

🡪I can add a null key and many null values in hash map.

🡪hashtable doesn’t allow the null elements.

### 123. **Can you add a null element into a TreeSet or HashSet?**

🡪 I can add null elements is hashset

🡪in Treeset it doesn’t allow null elements