Created by : myckelraj

Java Applications:

- Is universal, class-based, object-oriented and platform independent.
- Java is extensively used for Java application development in desktop applications, data centres, gaming consoles, scientific supercomputers, mobile applications like Netflix, Spotify, and LinkedIn, IOT's things.

Note:

- Because it will support all the object-oriented concepts.
- Without creating class, we cannot write the single line of code
- If any language supports these six object-oriented features, those types of languages are called object-oriented languages.
- Static typed programming language

```
int x=100;
  x=200; // allowed
 x="Eagle"// not allowed because we specified datatype.
```

Ways of Executing a Java Program:

Java Source code (.java) = Java compiler (javac) => Java Bytecode(.class) => Java virtual machine: Machine code: System output

Features Of Java:

(object-oriented, case sensitive, platform dependent).

Java Enterprise Edition:

Spring, Struts, Hibernate, Apache Hadoop, and JSF are the most popular Java frameworks for building Web applications.

Java Standard Edition:

It includes Java programming apis such as java.lang, java.io, java.net, java.util, java.sql, java.math, etc. It includes core topics like oops, strings, Regex, exceptions, Inner classes, Multithreading, I/O Stream, Networking, AWT, Swing, Reflection, Collection, etc.

JDK:

- ❖ It represents full-fledged Java software. Whenever we want to develop Java based applications we have to download and install JDK. That's the first component.

- ❖ To compile the Java code, we need a toolkit and that tool is known as JDK (Java Development Kit). It contains the JRE + JAVAC compilers.

JRE:

- We don't need to install complete Jdk software and we just need to have only the JRE component is enough to provide the run time environment on which we install and run our applications.

- If we install the JDK by default we can get the JRE and We know Jre is a subset of JDK. Someone developed the java applications and I would rather run the application then proceed only with JRE.

- It contains class libraries + JVM. JVM validates the byte code and loads the class.

JVM:

- o We can call JVM the Heart of Java.
- o By default JVM comes along with the JRE itself. We write the programs in English format so this should be converted into byte code and byte code is understandable by any machine it can be any type of operating system.
- o It is usually responsible for converting the byte code into machine code.
- o It find the errors

Java Independent Language

Java is developed in such a way that it does not depend on any hardware or software due to the fact that it compiles and converts it into byte code, which can be run on multiple systems.

Note:

- ❖ It will support windows, mac, Linux
- ❖ Suppose we can write the java program on windows the same java program we can execute on other operating systems. We don't need to change the code from one operating system to another system. We can easily execute the programs on different platforms.

Bytecode

It is platform independent code, and the set of instructions for a Java virtual machine may differ from system to system.

Machine code A set of instructions in machine language or binary which can be directly executed by the CPU.

Created by : **myckelraj**

Variables:

- A variable is a container that holds the value while the Java program is executed. A variable is assigned a data type.
- A variable is a name for a memory location. There are three types of variables in Java: local, instance, and static.

Local Variable:

- A variable declared inside the body of the method is called a local variable.
- A local variable cannot be defined with the "static" keyword.

Instance Variable:

- A variable declared inside the class but outside the body of the method, is called an instance variable.
- It is not declared as static.

Static variable (class variable)

- The static variable can be used to refer to the common property of all objects. The static variable gets memory only once in the class area at the time of class loading.
- It has the advantage of catching errors at compile time rather than at execution time. Static variables in Java are stored in heap memory.

Data Types

Primitive data types:

Number without decimal: byte(1-byte), short(2), int(4), long(8)
Number with decimal: float(4), double(8)
Single character: char(2)
True or False: Boolean(1-bit)

Type	Size	Range	Default Value	Example

byte	1 byte	-128 to 127	0	byte b = 123;
short	2 bytes	-32768 to 32767	0	short s = 1234;
Int	4 bytes	-2^{31} to $2^{31}-1$	0	int i = 123456;
long	8 bytes	-2^{63} to $2^{63}-1$	0L	long l = 312456L; long ll = 312456l;
float	4 bytes	1.4E-45f to 3.4028235e38f 7 decimal digits	0.0f	float f = 123.45f; float ff = 123.45F;
double	8 bytes	4.9E-324 to 1.7976931348623157E308 15 decimal digits	0.0d	double d = 1234.67d; double dd = 1234.67D;
char	2 bytes	'\u0000' (or 0) to '\uffff' (or 65,535 inclusive)	'\u0000'	char c = 'C';

boolean	Represents 1 bit of information	Not Applicable	False	boolean b = true;

The non-primitive data types include

Strings, Arrays, Class, Object, Interface, ArrayList

How many types of operators are available in Java?

Operator Type	Category	Precedence
Unary	Postfix	*Expr*++ *expr*--
	Prefix	++*expr* --*expr* +*expr* -*expr* ~ !
Arithmetic	Multiplicative, division, modulo	* / %
	Additive, subtraction	+ -
Shift	Shift	<< >> >>>
Relational	Comparison	< > <= >= instanceof
	Equality	== !=
Bitwise	Bitwise AND	&
	Bitwise exclusive OR	^
	Bitwise inclusive OR	|
Logical	Logical AND	&&

	Logical OR	||
Ternary	Ternary	? :
Assignment	Assignment	= += -= *= /= %= &= ^= |= <<= >>= >>>=

Five Ways To Create the Objects:-

- Using a new Keyword:
- Using clone() method:
- Using newinstance() method
  - Deserialization: The reverse process (byte-stream to object) of serialisation is called deserialization.
- Serialization: (The process of converting an object into a sequence of byte-stream.)

OOPS:

- OOP is faster and easier to execute.
- OOP provides a clear structure for the programs and makes the code easier to maintain, modify and debug.

Class:

(State-Attributes-properties & behaviour-methods)

- A blueprint from which the object is created and it does not occupy memory space and [Logical entity]
- It contains a collection of variables and methods.

Object:

- It is an instance of a class and occupies memory space.
- Every object has state and behaviour in the form of instance fields and methods respectively. [Physical entity]

Created by : **myckelraj**

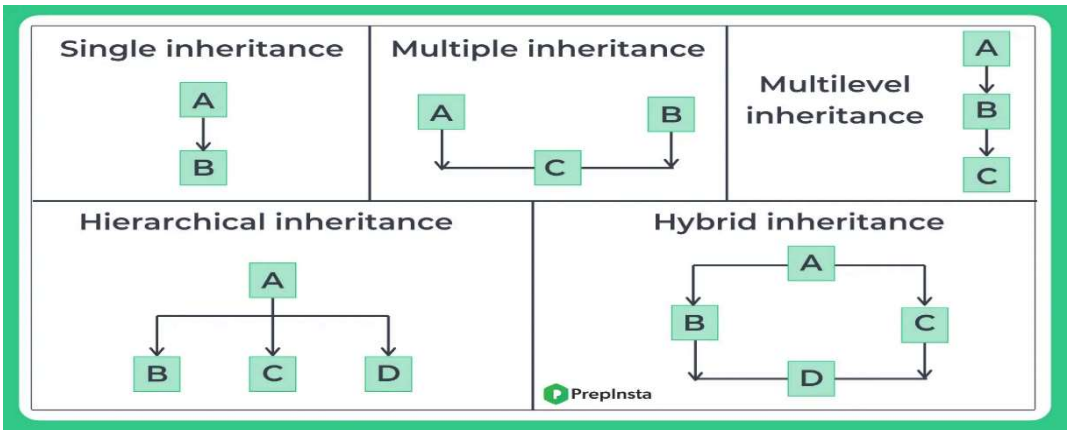
Inheritance:

- It is possible to inherit attributes and methods from one class to another. We group the "inheritance concept" into two categories:
  Subclass (child) - the class that inherits from another class
  Superclass (parent) - the class being inherited from
- It represents the IS-A relationship, which is also known as a parent-child relationship.

Types:

- When a class inherits another class, it is known as a single inheritance.
- When there is a chain of inheritance, it is known as multilevel inheritance
- When two or more classes inherit a single class, it is known as hierarchical inheritance.
- When a single class inherit two or more classes called as Multiple inheritance

[1. Re-usability 2. Avoid Duplication]



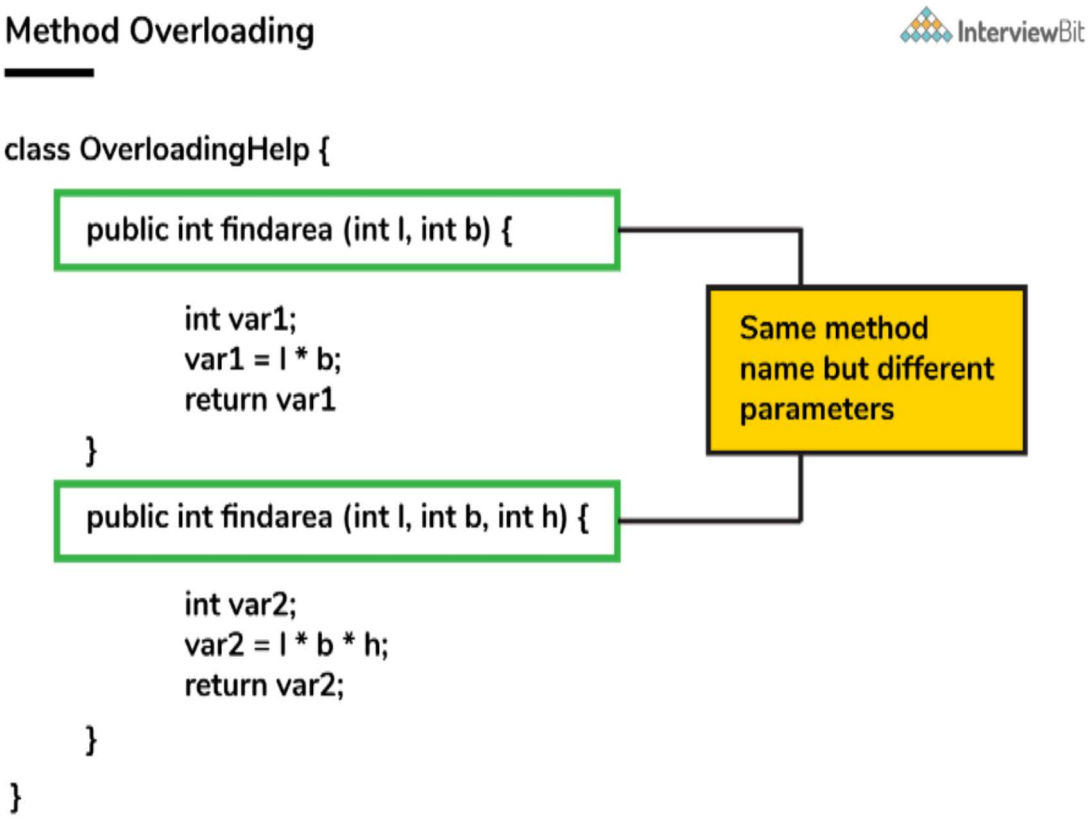
Polymorphism:

It is said that one thing has many forms. It can be categorised into two methods are

Method Overloading:

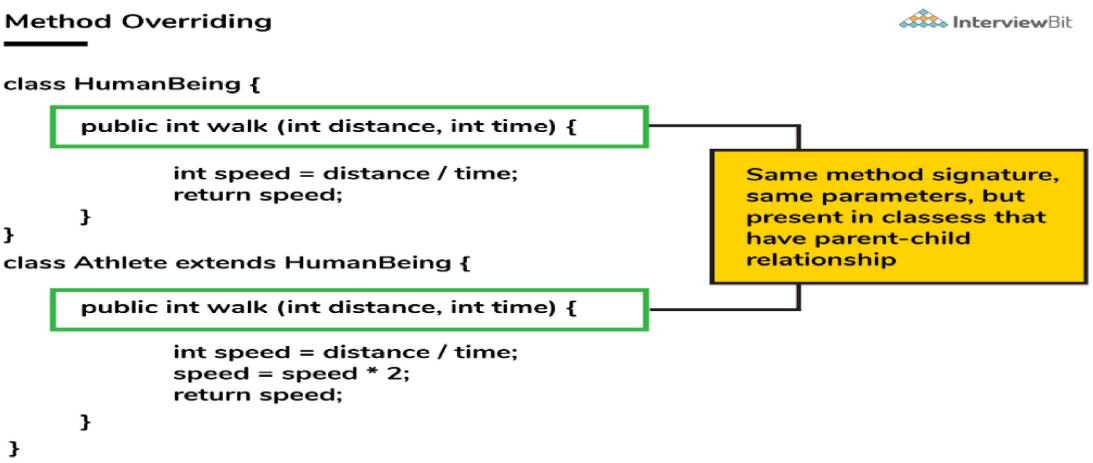
- If a class has multiple methods with the same name but different parameters. It is a compile-time polymorphism.
- Parameters and data type of parameters should be different.

We don't need to remember multiple method nameS and it simplified easily to remember one method name.



B) Method overriding:

- Method overriding is used to provide the specific implementation of a method that is already provided by its superclass.
- Method overriding is used for runtime polymorphism.
- The method must have the same name and the same parameter as in the parent class. There must be an IS-A relationship (inheritance).



Overriding: It is possible only in multiple classes(inheritance). We can't change the signature of the method.
Overloading: It is possible in single and multiple classes(inheritance). We can change the signature of the method.

Abstraction:

- Abstraction is the process of hiding certain details and showing only essential information to the user. E.g. Considering an ATM Machine, we all know how to perform operations on the ATM machine like cash withdrawal, money transfer, retrieval of mini-statements, etc. But we did not know the internal happenings of the ATM.

- Abstract class: this is a restricted class that cannot be used to create objects. An abstract class can have both abstract and regular methods or Non-Abstract methods.

- Abstract method: it can only be used in an abstract class, and it does not have a body and is provided by the subclass.

Interface:

- An interface is a completely "abstract class" that is used to group related methods with empty bodies.
- It supports the functionality of multiple interfaces and we can define it with interface keywords.
- Java Interface contains a method declaration but not its definition.
- We can create object references for the interface but we cannot instantiate the interface.
- Static method can access directly from interface and Static variable can access directly from interface.
- It contains public, default, static, abstract methods.
- It contains public, static, final variables.

[interface- implements-class
class-extend-class
interface-extend-interface]

Abstract Class	Interface Class

Both abstract and non-abstract methods may be found in an abstract class.	The interface contains only abstract methods.
Abstract Class supports final, non-final, static, non-static variables	Interface has supports only static and final variables
Multiple inheritance is not supported by the Abstract class.	Multiple inheritance is supported by the Interface.
Abstract Keyword is used to declare Abstract class.	Interface keyword is used to declare the interface
Extend keyword is used to extend an Abstract Class.	Interface Keyword is used to implement the interface.

Encapsulation:

- **A wrapping up of variables and methods into a single entity. It makes all the data members of the class private.**
- Private variables can be accessed by using setter and getter methods.

- Encapsulation is important in Java because it provides a level of security, promotes modularity and flexibility, and improves code reusability.

Difference Class & Interface

Class	Interface
1. It can be instantiated create object	It can't instantiate variable and and create object
2.It can create concrete method	It can't create concrete method
3.All access modifiers are used	Only one modifier as public

What is the difference between encapsulation and abstraction

- Encapsulation is concerned with hiding the internal details of an object. Abstraction is concerned with hiding the implementation details of a class.
- Encapsulation solves the problem at the implementation level. Abstraction solves the problem at the design level.

Methods

- A group of statements which will perform.
- It is used to achieve the reusability of code.

Constructor:

- A constructor is a special method that is used to initialise objects. There are two types of constructors in Java, as mentioned below:

Default constructor

- Is the type that does not accept any parameter value. It is used to set initial values for object attributes.

Parameterized Constructor

- Is the type of constructor that accepts parameters as arguments

Differences between constructor and method of a class in Java:

- Constructor is used for initialising the object state. Method is used for exposing the object's behaviour.
- Constructor has no return type. Method should have a return type.
- Constructor gets invoked implicitly. Method has to be invoked on the object explicitly.

- Constructor name should be equal to the class name. The name of the method can have any name or have a class name too.

What is the difference between static (class) method and instance method?

Static method	Instance method
Static method is associated with a class rather than an object.	The instance method is associated with an object rather than a class.
Static methods can be called using the class name only without creating an instance of a class.	The instance method can be called on a specific instance of a class using the object reference.
Static methods do not have access to this keyword.	Instance methods have access to this keyword.
This method can access only static members of the class	This method can access both static and non-static members of the class.

Static methods cannot be overridden.	Instance methods can be overridden.

11.Keywords

- Keywords can be defined as a set of predefined reserved words that have a special meaning for the compiler.
- It cannot be used as classes, methods, variables, or any other identifier

This keyword

In Java, this is a reference variable that refers to the current object. We can say that we cannot use this() and super() keywords in the same block.

Super Keyword

In Java, a reference variable is used to refer to an immediate parent class object. Super can be used to refer to an immediate parent class instance variable, method, class constructor.

Final keyword:

- It stops value change, method overriding, and inheritance.
- Final variable: When a variable is declared as final in Java, the value can't be modified once it has been assigned.
- Final method: A method declared as final cannot be overridden by its children's classes. A constructor cannot be marked as final.
- Final class: No classes can be inherited from the class declared as final. But that final class can extend other classes for its usage.

Modifiers in Java:-

Access Modifiers - controls the access level (Public, Private, Protected and Default)

(public =>The code is accessible for all classes
private => The code is only accessible within the declared class
default =>The code is only accessible in the same package.
Protected =>The code is accessible in the same package and subclasses)

Non-Access Modifiers - do not control access levels but provide other functionality (static, final, synchronized)

Created by : **myckelraj**

Package:

- It is a collection of related classes and interfaces. We use packages to avoid name conflicts, and to write better, more maintainable code.
- Built-in packages (packages from the Java API). User-defined packages (create your own packages).

Arrays:

An array is a collection of elements of the same datatype.

We can store multiple values into a single variable.
Arrays are fixed in size, and their elements are ordered.

```
int a[]= new int[6];
int a[][]= new int[3][3];
```

Char

In Java, char is used to store a single character. The character must be enclosed in single quotes.

String:

A string in Java is an object that holds multiple characters. It is not a primitive data type. A string can be created by placing characters between a pair of double quotes (").
(==) to compare the objects
(.equal) to compare the values of the objects

Why is String immutable in Java?

- **String is immutable, its value cannot be modified. After its declaration, it continues to stay in the string pool as long as it is not removed in the form of garbage.**
- Otherwise, any hacker could change the referenced value to cause security issues in the application

Thread:

- Threads allow a program to operate more efficiently by doing multiple things at the same time.

- Threads can be used to perform complicated tasks in the background without interrupting the main program.

Checked Exceptions:

- A checked exception is an exception that is checked (notified) by the compiler at compilation-time, these are also called compile time exceptions.
- These exceptions cannot simply be ignored, the programmer should take care of them.

1.filenotfoundexception

2.ioexception

3.SQLEXCEPTIONS

4.classnotfoundexceptions

Unchecked Exception:

- An unchecked exception is an exception that occurs at the time of execution.
- These are also called runtime exceptions. These include programming bugs, such as logic errors or improper use of an API.

1.arrayindexoutofboundsexception:

2. Null pointer exception

3. Arithmetic exception:

Try and catch block:

- The try statement allows you to define a block of code to be tested for errors while it is being executed.
- The catch statement allows you to define a block of code to be executed, if an error occurs in the try block.

Finally Block:

The finally block follows a try block or a catch block. If you want to execute, no matter what happens in the protected code.

Throw Keyword:

- The throw statement allows you to create a custom error.
- The throw statement is used together with an exception type.
- There are many exception types available in Java:
- {arithmeticexception,filenotfoundexception, arrayindexoutofboundsexception, securityexception).

Throws Keyword:

If a method does not handle a checked exception, the method must declare it using the throws keyword. The throws keyword appears at the end of a method's signature.

17.Enum:

- Is a special class that represents a group of constants. Enum is short for "enumerations," which means specifically listed.
- An enum cannot be used to create objects, and it cannot extend other classes.
- The valueof() method and the ordinal() method.

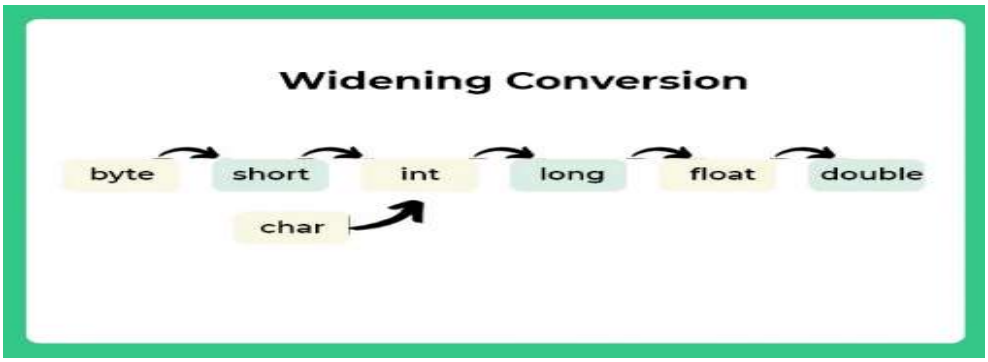
Why And When To Use Enums

-Use enum when you have values that you know aren't going to change, like month days, days, colours, or decks of cards.

Casting:

- Converting a data type of smaller size to a data type of larger size is known as Widening conversion.
- This type of casting/conversion is done automatically

Implicitly.



- Converting a data type of larger size to a data type of smaller size is known as a narrowing conversion.
- This type of conversion does not occur automatically; hence, we need to convert explicitly using the cast operator “( )”.
- Thus, it is also known as explicit type casting.

Created by : **myckel raj**

Autoboxing:
Is the process of automatically converting primitive data types into their corresponding wrapper classes, such as int to Integer, long to Long, float to Float, boolean to Boolean, double to Double, and short to Short.

Unboxing:

It simply involves autoboxing in reverse. Unboxing is the process of converting a wrapper class object to its corresponding primitive type.

Equal Vs (==)
.equal is used for checking the equality of contents between two objects.
== operator is used for comparing addresses (or references)

Why is the main method static in Java?
It is methods that belong to the classes, not to an individual object. Because there must be only 1 main method in the java program as the execution starts from the main method.

Singleton classes:

- It defines a class that has only one instance and provides a global point of access to it".
- It Saves memory because the object is not created at each request. Only a single instance is reused again and again.

Why does the java array index start with 0?

It is because the 0 index array avoids the extra arithmetic operation to calculate the memory address. [Base Address + (index * no_of_bytes)]

What do you understand about marker interfaces in Java?

Marker interfaces, also known as tagging interfaces, are those interfaces that have no methods or constants defined in them.

What will happen if we don't declare the main as static?

We can declare the main method without using static and without getting any errors. But, the main method will not be treated as the entry point to the application or the program.

Why do we need wrapper classes?

The wrapper class is an object class that encapsulates the primitive data types, and we need them for the following reasons:

1. Wrapper classes are final and immutable
2. Provides methods like valueof(), parseInt(), etc.
3. It provides the features of autoboxing and unboxing.

How many ways can you take input from the console?
There are two methods to take input from the console in Java mentioned below:

1. Using Command line argument
2. Using Bufferedreader Class
3. Using Console Class
4. Using Scanner Class

What do you mean by aggregation?
Aggregation is a term related to the relationship between two classes best described as a “has-a” relationship.

What is literal in java programming?

Any constant value that can be assigned to the variable or used in an expression is called a literal /constant. For example, 10, 2code,, 'a', true, "refresh java,"etc.c are literals.

What is the difference between a compiler and Interpreter

The interpreter scans the program line by line and translates it into machine code, whereas the compiler scans the entire program first and then translates it into machine code. The interpreter shows one error at a time, whereas the compiler shows all errors and warnings at the same time.

Bubble sort It compares adjacent elements and swaps them if they are in the wrong order until the entire array is sorted
Insertion sort The array is virtually split into a sorted and an unsorted part. Values from the unsorted part are picked and placed at the correct position in the sorted part.

Created by : **myckelraj**

Selection sort

It repeatedly finds the minimum element from the unsorted part of the array and swaps it with the first unsorted element until the entire array is sorted.

JSP:

- 1. It is a tag-based Approach,
- 2. Session Time management is enabled by default.
- 3.Jsp is easy to code. It allows reading/writing data easily in the database.
- 4.Jsp does have implicit objects viz request, response, Session, out
- 5.It is portable, powerful, flexible, and easy to maintain
- 6.web.xml mapping is not necessary for JSP.

Servlets

- 1.It is a pure java code.
- 2.Session Time management is not enabled by default.
- 3.it is not easy to code.
- 4.It does not have implicit objects.
- 5.It is not portable, powerful, flexible, and easy to maintain.
- 6.web.xml mapping is necessary for servlets.

JDBC:

Java Database Connectivity is an API(Application programming interface) used in java programming to interact with databases. It allows a Java Program to access database management Systems

Hibernate:-

- *Hibernate is an ORM (Object-Relational Mapping) tool that is used to save the Java objects in the database system.*
- Hibernate implements the specifications of JPA (Java Persistence API) for data persistence (the handling of data in Java applications).
- Java Application => The ORM tool internally uses the JDBC API to interact with the database)=> Database.
- we don't need to write database specific queries. there is no need to create tables in the database manually.

Created by : **myckel raj**

Apache Tomcat web server:

- This server is a JSP/Servlet container. It can handle both static pages and dynamic pages. The static pages are generated by using HTML.
- The dynamic pages are generated using Servlet and JSP. It can be used only for hosting JAVA based code.
- Apache Tomcat server is slower compared to Apache web server. It can be more robust compared to an Apache web server. Apache Web server is faster compared to Apache Tomcat server.

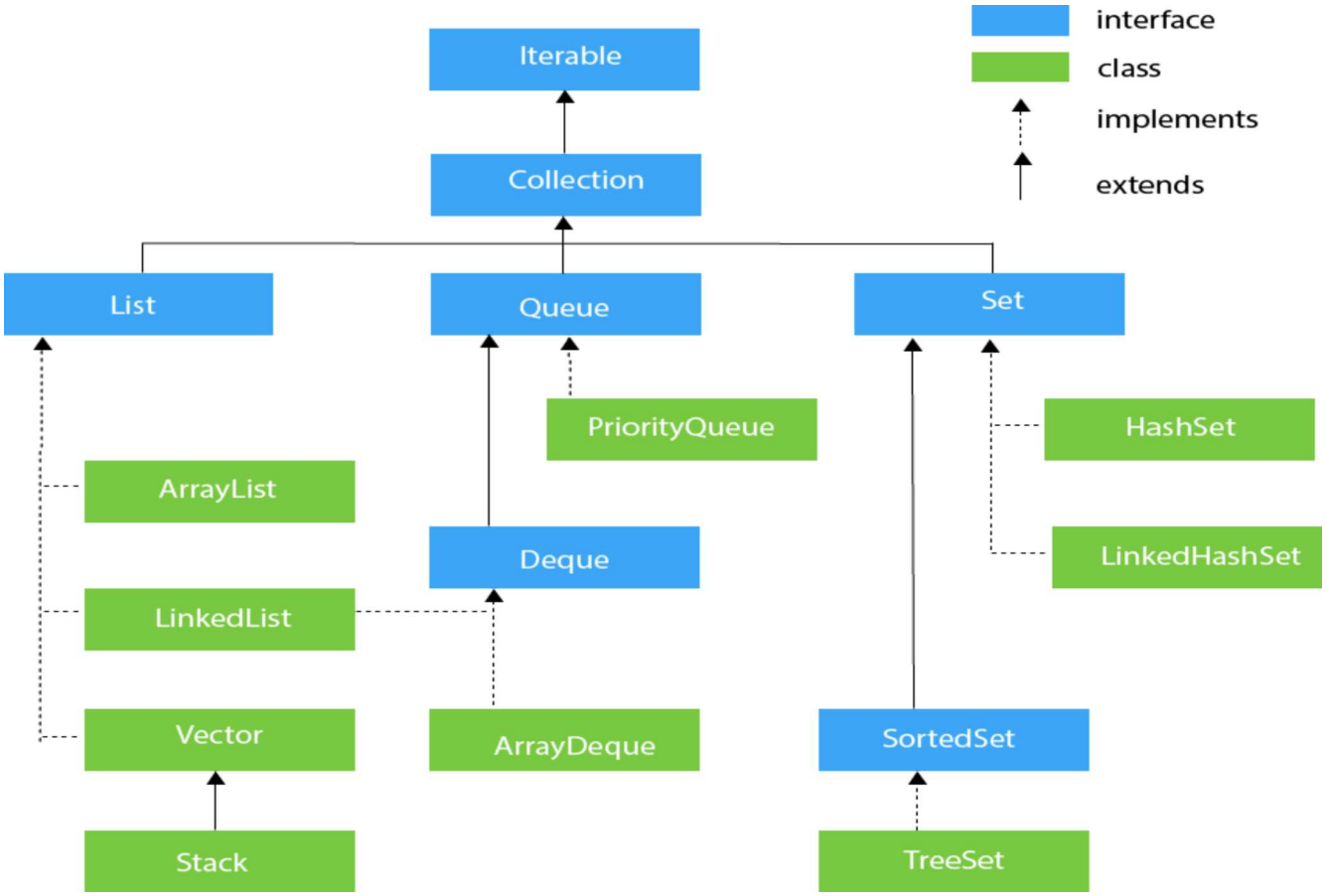
Java Servlets:

- A servlet is a Java Programming language class. Servlet is a technology which is used to create a web application.

- A client sends a request to the server and the server generates the response, analyses it and responds to the client. Better performance, Portability, Robust, Secure. It uses the Java language.
- For example: A Railways reservation system is a good real time example of understanding servlets. When a user searches for train schedules and availability, such information requests are received by servlets containing the origin ,destination, and travel dates. It can then search in train schedules databases to find matching trains and availability and then display them to the user.

Collections in Java

- The Collection in Java is a framework that provides an architecture to store and manipulate the group of objects
- Java Collection means a single unit of objects. The Java Collection framework provides many interfaces (Set, List, Queue, Deque) and classes (arraylist, Vector, linkedlist, priorityqueue, hashset, linkedhashset, treeset).



Stack:

- It stores items that have a very short life such as methods, variables, and reference variables of the objects.
- It follows the LIFO order.It has faster access, allocation, and deallocation. Allocation and deallocation is done automatically by the compiler.
- It is thread-safe

Heap:

- It stores objects and Java Runtime Environment (JRE) classes.
- It does not follow any order because it is a dynamic memory allocation.
- It has slower access, allocation, and deallocation.
- It was done manually by a programmer and it is not thread safe.

Class loader

A program that belongs to JRE (Java Runtime Environment).

The task of the class loader is to load the required classes and interfaces to the JVM when required. Example: To get input from the console, we

require the scanner class. And the Scanner class is loaded by the class loader

Shallow copy: The shallow copy only creates a new reference and points to the same object.

Deep copy: we create a new object and copy the old object value to the new object.

Differences between Java and C++?

Basis	C++	Java
Platform	C++ is Platform Dependent	Java is Platform Independent
Application	C++ is mainly used for System Programming	Java is Mainly used for Application Programming
Hardware	C++ is nearer to hardware	Java is not so interactive with hardware
Global Scope	C++ supports global and namespace scope.	Java doesn't support global scope.

Created by : **myckelraj**

What's the difference between the methods sleep() and wait()?

Sleep()	Wait()
The sleep() method belongs to the thread class.	Wait() method belongs to the object class.
Sleep does not release the lock that the current thread holds.	Wait() release the lock which allows other threads to acquire it.
This method is a static method.	This method is not a static method.

What is the difference between Set and Map?

Set	Map
The Set interface is implemented using java.util package.	The map is implemented using java.util package.

It can extend the collection interface.	It does not extend the collection interface.
It does not allow duplicate values.	It allows duplicate values.
The set can sort only one null value.	The map can sort multiple null values.

What is the difference between Comparable and Comparator?

Comparable	Comparator
The interface is present in java.lang package.	The Interface is present in java.util package.
Provides compareto() method to sort elements.	Provides compare() method to sort elements.

It provides single sorting sequences.	It provides multiple sorting sequences.

Exception Handling

An Exception is an Event that interrupts the normal flow of the program and requires special processing.

How many types of exceptions can occur in a Java program?

Built-in Exceptions

Arrayindexoutofboundsexceptions

- Classnotfoundexception
- Filenotfoundexception
- Ioexception
- Nullpointerexception
- Arithmeticexception
- Interruptedexception
- Runtimeexception

User-Defined Exceptions: User-defined exceptions are defined by the programmers themselves to handle some specific situations.

What is the difference between Checked Exception and Unchecked Exception?

Checked Exception: Checked Exceptions are the exceptions that are checked during compile time of a program.

Unchecked Exception: Unchecked are the exceptions that are not checked at compile time of a program.

Created by : **myckelraj**

What are the advantages of multithreading?

- Responsiveness
- Resource Sharing
- Economy
- Scalability
- Better Communication
- Utilisation

Process Vs Thread?
A process is a program in execution. Thread is a single sequence of instructions within a process.

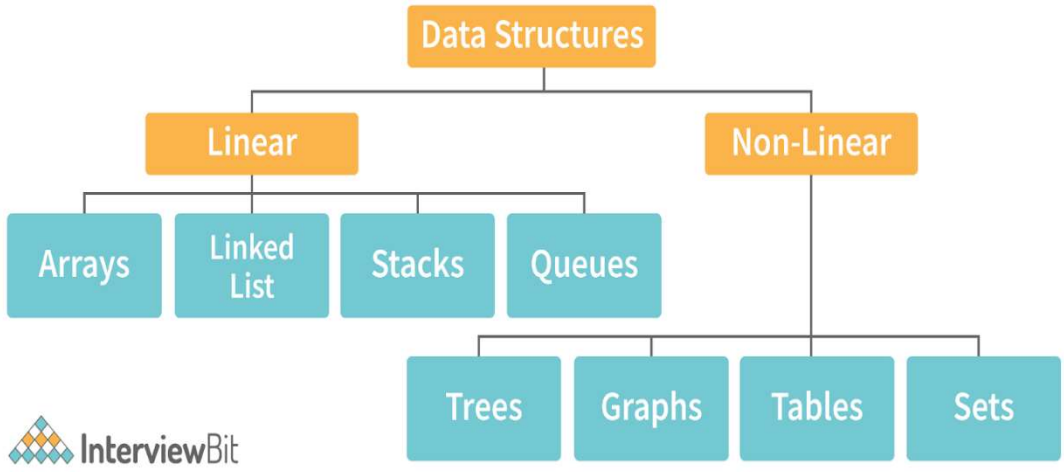
The process does not share data with each other. Threads share data with each other. The thread takes less time to terminate.

Differentiate between List and Set.

List	Set
Ordered	Unordered
List allows duplicates.	Set does not allow duplicate values.
List is accessed by index.	Set is accessed by hashcode.

Multiple null elements can be stored.	Null elements can store only once.
Examples are arraylist, linkedlist, etc.	Examples are hashset and treeset. Linkedhashset etc.

Data structures:
Data structure is a way of defining, storing & retrieving data in a structural & systematic way.



Different operations available in stack data structure?

Push: This adds an item to the top of the stack. The overflow condition occurs if the stack is full.

- Pop: This removes the top item of the stack. Underflow condition occurs if the stack is empty.
- Top: This returns the top item from the stack.
- Is empty: This returns true if the stack is empty else false.
- Size: This returns the size of the stack

Different operations available in queue data structure?

- Enqueue: This adds an element to the rear end of the queue. Overflow conditions occur if the queue is full.

- Dequeue: This removes an element from the front end of the queue. Underflow conditions occur if the queue is empty.
- Is empty: This returns true if the queue is empty or else false.
- Rear: This returns the rear end element without removing it.
- Front: This returns the front-end element without removing it.
- Size: This returns the size of the queue.

Array vs ArrayList

Array	ArrayList
Arrays are static in nature. Arrays are fixed length data structures. You can't change their size once they are created.	ArrayList is dynamic in nature. Its size is automatically increased if you add elements beyond its capacity.
Arrays can hold both primitives as well as objects.	ArrayList can hold only objects.
Arrays can be iterated only through *for* loop or *for-each* loop.	ArrayList provides iterators to iterate through their elements.
The size of an array is checked using *length* attribute.	The size of an ArrayList can be checked using *size()* method.
Array gives constant time performance for both add and get operations.	ArrayList also gives constant time performance for both add and get operations provided adding an element doesn't trigger resize.
Arrays don't support generics.	ArrayList supports generics.
Arrays are not type safe.	ArrayList are type safe.
Arrays can be multi-dimensional.	ArrayList can't be multi-dimensional.
Elements are added using assignment operator.	Elements are added using add() method.

Arraylist	Linkedlist
1) arraylist internally uses a dynamic array to store the elements.	Linkedlist internally uses a doubly linked list to store the elements.
2) Manipulation with arraylist is slow because it internally uses an array. If	Manipulation with linkedlist is faster than arraylist because it uses a

any element is removed from the array, all the other elements are shifted in memory.	doubly linked list, so no bit shifting is required in memory.
3) An arraylist class can act as a list only because it implements List only.	Linkedlist class can act as a list and queue both because it implements List and Deque interfaces.
4) arraylist is better for storing and accessing data.	Linkedlist is better for manipulating data.
5) The memory location for the elements of an arraylist is contiguous.	The location for the elements of a linked list is not contagious.
6) Generally, when an arraylist is initialized, a default capacity of 10 is assigned to the arraylist.	There is no case of default capacity in a linkedlist. In linkedlist, an empty list is created when a linkedlist is initialized.
7) To be precise, an arraylist is a resizable array.	Linkedlist implements the doubly linked list of the list interface.

Created by : **myckelraj**

Stack Vs Queue

1.Stack is a linear data structure where data is added and removed from the top.
Queue is a linear data structure where data is ended at the rear end and removed from the front.

2.Stack is based on the LIFO(Last In First Out) principle.
Queue is based on the FIFO(First In First Out) principle.

3.Insertion operation in Stack is known as push.
Insertion operation in Queue is known as enqueue.

4.Delete operation in Stack is known as pop.
Delete operation in Queue is known as dequeue

5.Used in solving recursion problems.
Used in solving sequential processing problem

Hashset:

A hashset is a collection of items where every item is unique, and it is found in the java.util package:

Hashmap:

However, store items in "key/value" pairs, and you can access them by an index of another type (e.g. A String).

HashMap vs HashSet:

Parameter	HashMap	HashSet
Interface	This is core difference among them.HashMap implements Map interface	HashSet implement Set interface
Method for storing data	It stores data in a form of key->value pair.So it uses put(key,value) method for storing data	It uses add(value) method for storing data
Duplicates	HashMap allows duplicate value but not duplicate keys	HashSet does not allow duplicate values.
Performance	It is faster than hashset as values are stored with unique keys	It is slower than HashMap
HashCode Calculation	In hash map hashcode value is calculated using key object	In this,hashcode is calculated on the basis of value object.Hashcode can be same for two value object so we have to implement equals() method.If equals() method return false then two objects are different.

	TreeSet	LinkedHashSet	HashSet
Ordering	Sorted in ascending order	Maintains insertion order	No defined ordering
Underlying Data Structure	Balanced Binary Search Tree (Red-Black Tree)	Doubly Linked List + Hash Table	Hash Table
Duplicates	Does not allow duplicate elements	Does not allow duplicate elements	Does not allow duplicate elements
Performance	Slower performance for insertion, deletion, and retrieval compared to HashSet	Slightly slower performance compared to HashSet	Fastest performance for insertion, deletion, and retrieval
Iteration Order	Sorted based on natural ordering or custom comparator	Maintains the insertion order	No defined iteration order
Null Values	Does not allow null values	Allows a single null value	Allows multiple null values
Usage	When elements need to be sorted in a specific order	When elements need to be maintained in insertion order	When elements need to be stored without any ordering

Transient

Is a variable modifier used in serialization. At the time of serialization, if we don't want to save the value of a particular variable in a file, then we use the transient keyword.

Transient keywords play an important role to meet security constraints.

There is no use/impact of declaring the final and static variable as transient.

Created by : myckel raj | Created by : GitHub