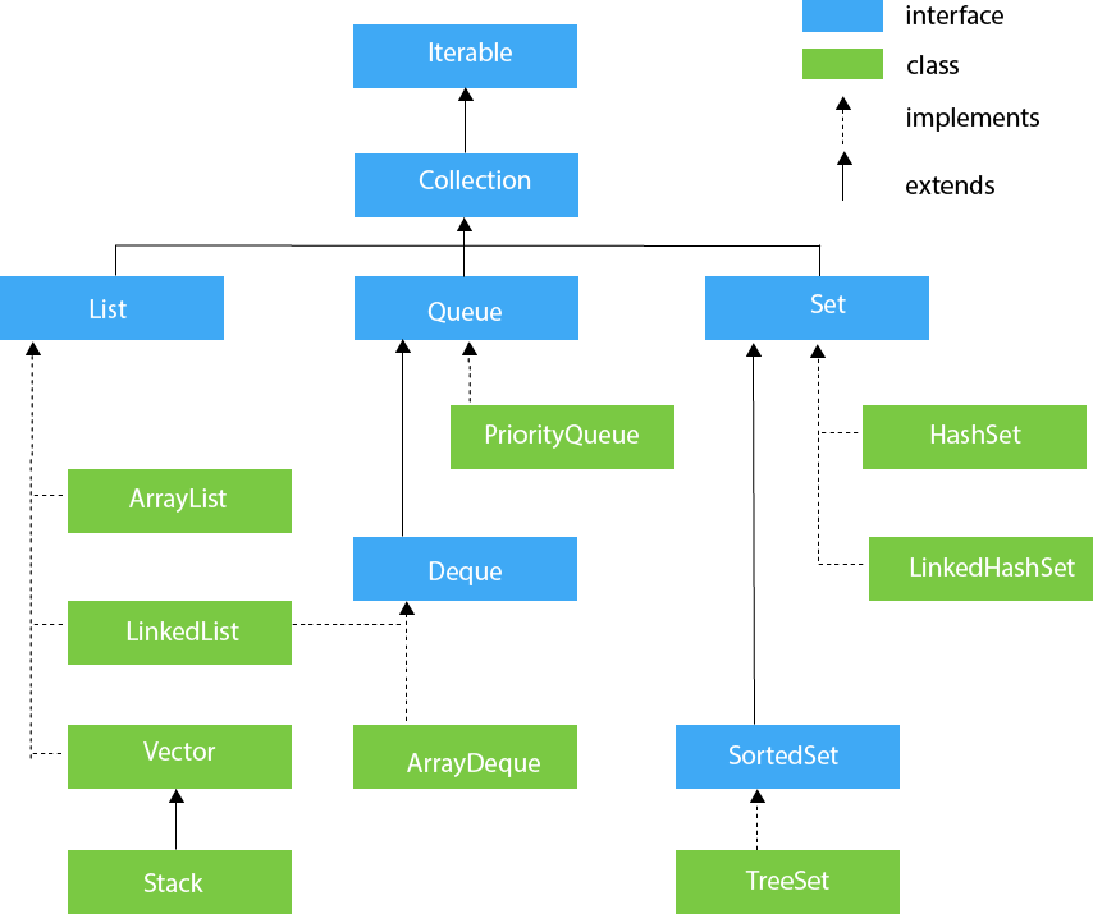
**Generics:**

Java Generics programming is introduced in J2SE 5 to deal with type-safe objects. It makes the code stable by detecting the bugs at compile time.

This is introduced to achieve type safety.

**Collection:**

The Collection in Java is a framework that provides an architecture to store and manipulate the group of objects.



* Hash-Set class implements Set Interface. It represents the collection that uses a hash table for storage. Hashing is used to store the elements in the Hash-Set. It contains unique items.
* Java Tree-Set class implements the Set interface that uses a tree for storage. Like Hash-Set, Tree-Set also contains unique elements. However, the access and retrieval time of Tree-Set is quite fast. The elements in Tree-Set stored in ascending order. TreeSet<String> set=new TreeSet<String>();
* List interface is the child interface of Collection interface. It inhibits a list type data structure in which we can store the ordered collection of objects. It can have duplicate values.

List <data-type> list1= new ArrayList();

List <data-type> list2 = new LinkedList();

List <data-type> list3 = new Vector();

List <data-type> list4 = new Stack();

A map contains values on the basis of key, i.e. key and value pair. Each key and value pair is known as an entry. A Map contains unique keys.

Java is not 100% Object-oriented because it makes use of eight primitive data types such as boolean, byte, char, int, float, double, long, short which are not objects.

**Abstract Datatype:**

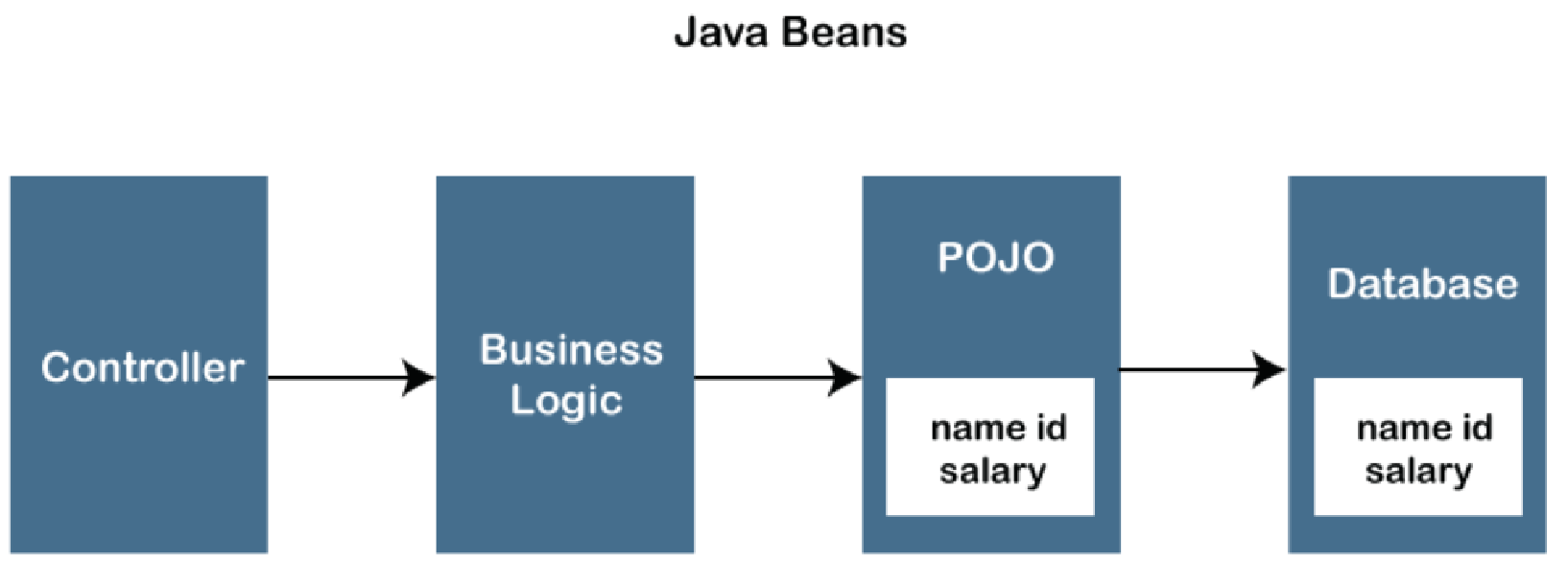
The definition of ADT only mentions what operations are to be performed but not how these operations will be implemented. It does not specify how data will be organized in memory and what algorithms will be used for implementing the operations. It is called “abstract” because it gives an implementation-independent view.

**POJOs:**

POJO in Java stands for Plain Old Java Object. It is an ordinary object, which is not bound by any special restriction. The POJO file does not require any special class-path. It increases the readability & re-usability of a Java program.

Below are some properties of the POJO class:

* The POJO class must be public.
* It must have a public default constructor and may have the arguments constructor.
* All objects must have some public Getters and Setters to access the object values by other Java Programs.
* The object in the POJO Class can have any access modifies such as private, public, protected. But, all instance variables should be private for improved security of the project.
* A POJO class should not extend predefined classes, should not implement prespecified interfaces and should not have any prespecified annotation.



**Interface in Java:**

An interface in Java is a blueprint of a class. It has static constants and abstract methods. It is used to achieve abstraction and multiple inheritance in Java.

we can have default and static methods in an interface.

we can have private methods in an interface.

note - Multiple inheritance by interface occurs if a class implements multiple interfaces or also if an interface itself extends multiple interfaces.



**C++ VS Java:**

~~C++ is an object-oriented programming language commonly used to create large-scale applications.~~ The main differences between C++ and Java are their platform dependence, use and thread support. C++ is platform-dependent, as opposed to Java, which is platform-independent. Java has in-built thread support, whereas C++ relies on third-party libraries for thread support. In terms of use, C++ is mainly for system programming, while Java is for application programming.

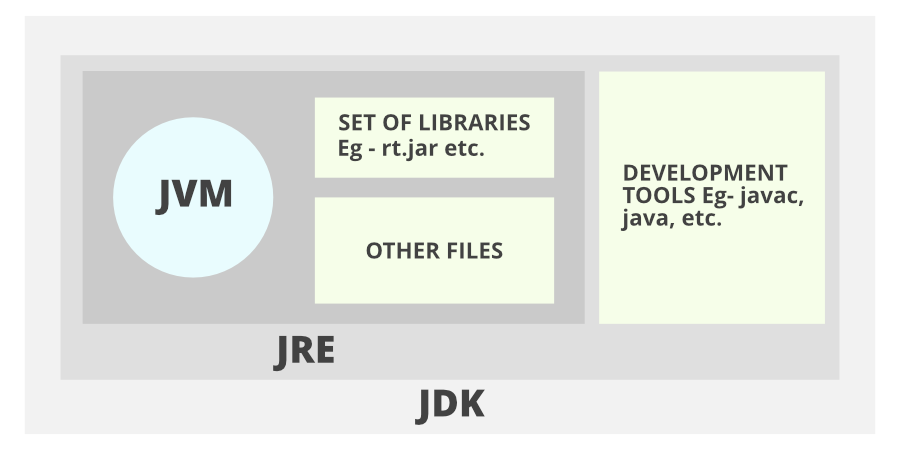
**Streams:**

A stream is a sequence of objects that supports various methods which can be pipelined to produce the desired result.

intermidate operations : map , filter , sorted

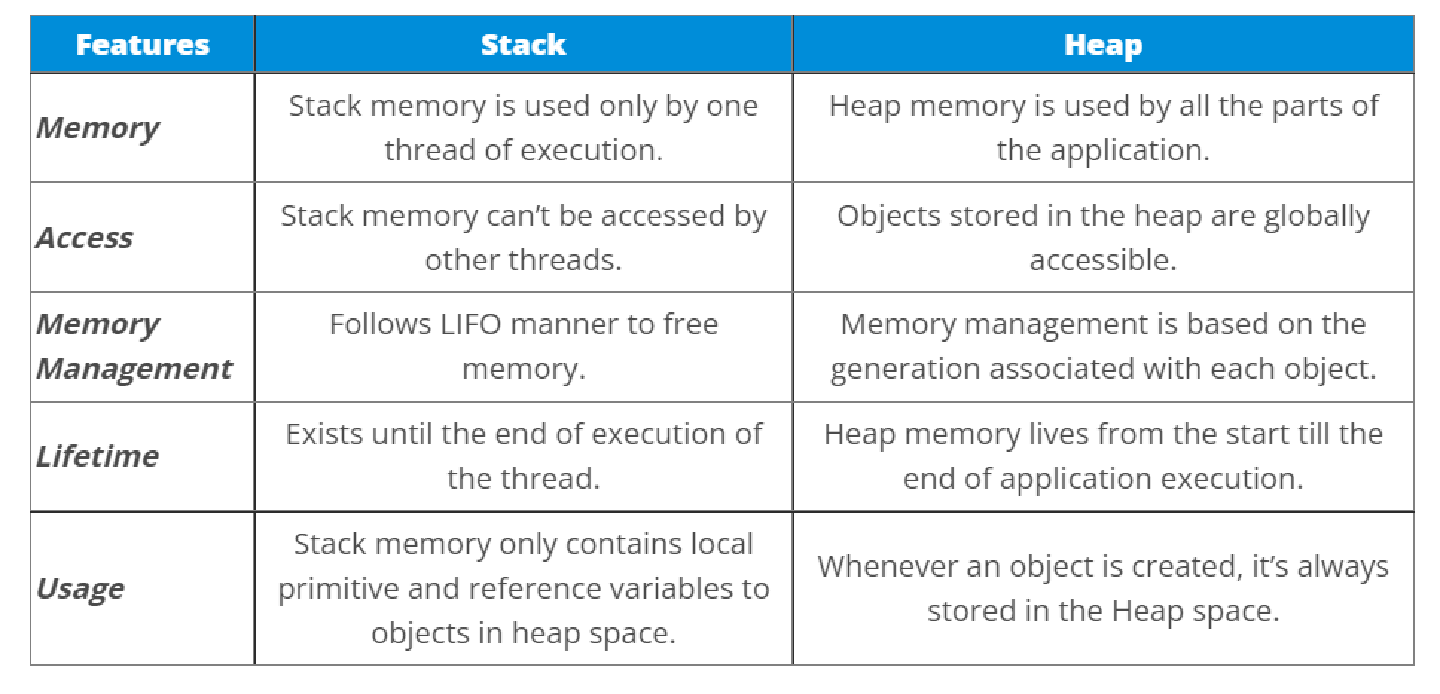
terminal operation : collect , foreach , reduce

**JRE , JVM and JDK:**



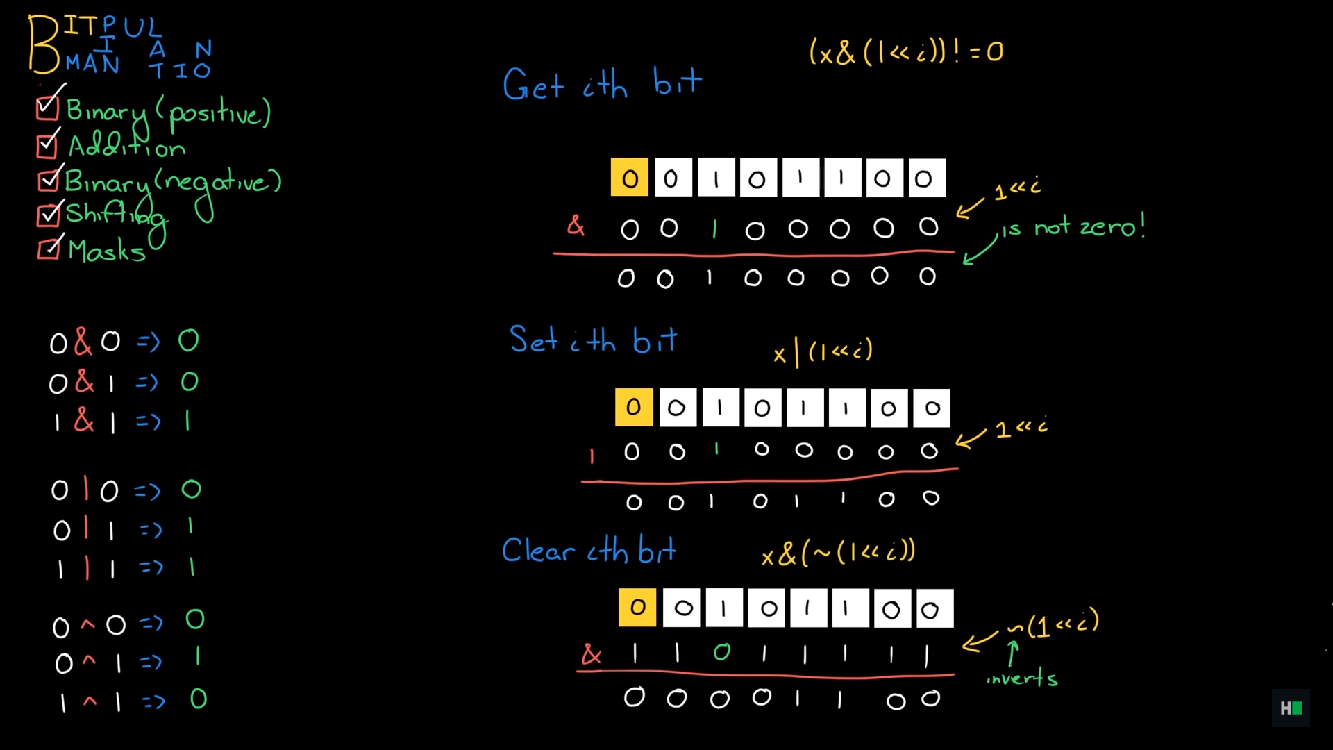
****

**stack vs heap memory**

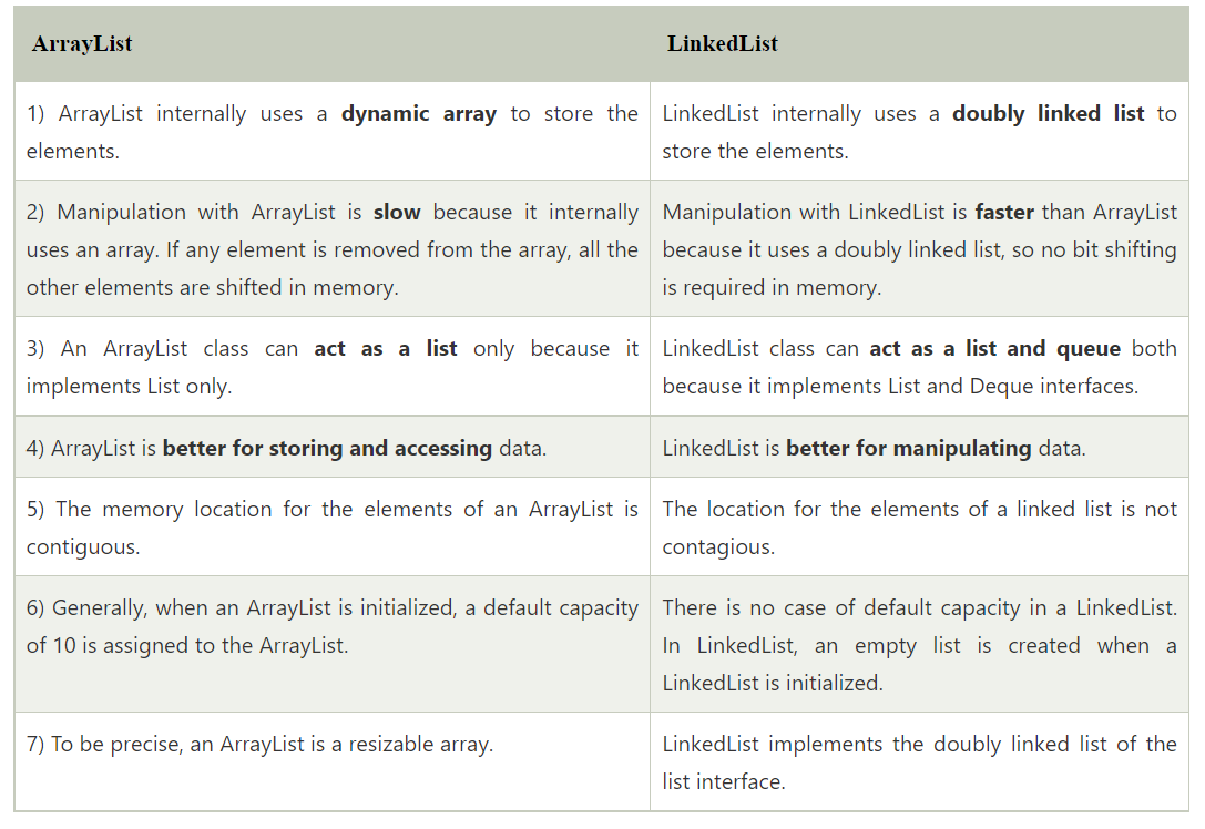


**Bit manuplation in c++:**

[**https://lex.infosysapps.com/web/en/viewer/youtube/lex\_auth\_012876384340353024138?collectionId=lex\_auth\_012862798083506176336&collectionType=Course&pathId=lex\_auth\_0135786108963962884009**](https://lex.infosysapps.com/web/en/viewer/youtube/lex_auth_012876384340353024138?collectionId=lex_auth_012862798083506176336&collectionType=Course&pathId=lex_auth_0135786108963962884009)

****

**ArrayList VS LinkList:**



**System.arraycopy**

public static void arraycopy(Object source\_arr, int sourcePos,

Object dest\_arr, int destPos, int len)

Parameters :

source\_arr : array to be copied from

sourcePos : starting position in source array from where to copy

dest\_arr : array to be copied in

destPos : starting position in destination array, where to copy in

len : total no. of components to be copied.

* 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.
* async makes a function return a Promise.await makes a function wait for a Promise

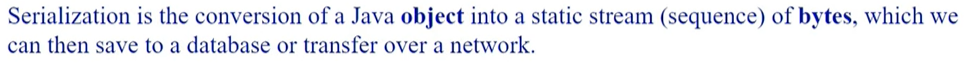
The main method is static because there is no object of the class existing when the Java runtime starts. And String args[] is the array of command line arguments passed to the code.

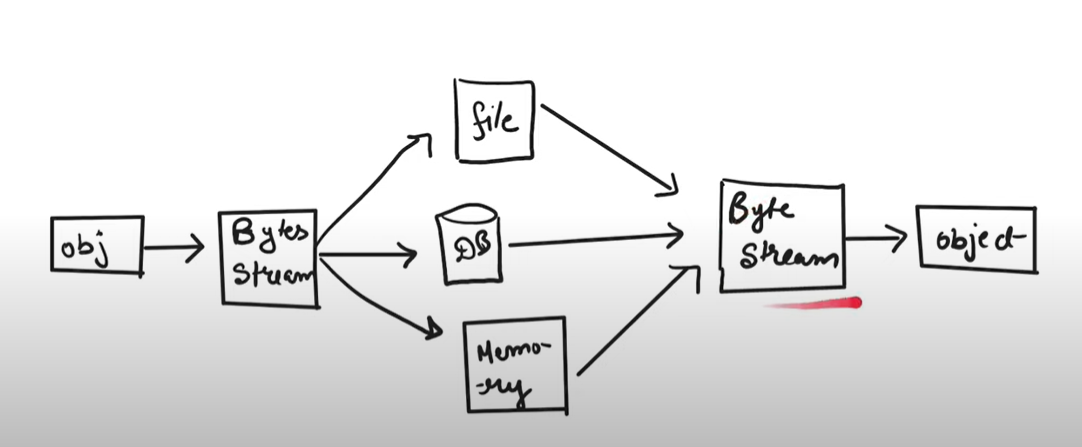
**Serializable interface:**

To serialize an object means to convert its state to a byte stream so that the byte stream can be reverted back into a copy of the object. A Java object is serializable if its class or any of its superclasses implements either the java.io.Serializable interface or its subinterface, java.io.Externalizable. Deserialization is the process of converting the serialized form of an object back into a copy of the object.

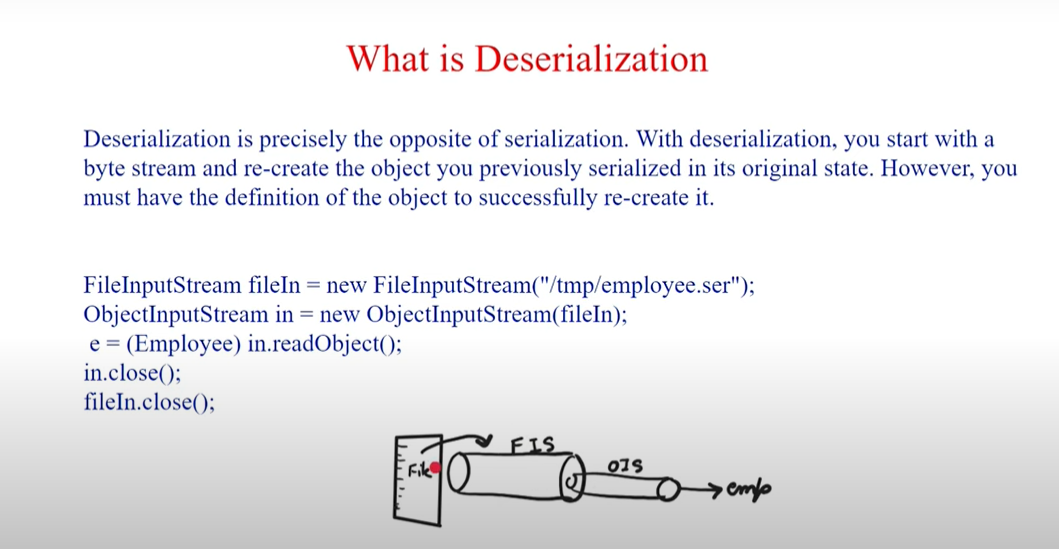
In other words, Serialization is the process of converting an object in your Java code into a format that can be easily saved to a file, sent over a network, or stored in a database. This format is usually a stream of bytes. The primary purpose of serialization is to persist or transmit the state of an object so that it can be reconstructed later. I

Object cannot be transferred directly to the network that’s why we need to serialize the object





<https://www.youtube.com/watch?v=nUFoDfGl1II> for more info



**Meaning of public static void main(String[] args){}:**

String[] args is a collection of Strings, separated by a space, which can be typed into the program on the terminal. More times than not, the beginner isn't going to use this variable, but it's always there just in case.

public : it is a access specifier that means it will be accessed by publically. static : it is access modifier that means when the java program is load then it will create the space in memory automatically. void : it is a return type i.e it does not return any value. main() : it is a method or a function name.

When you create arrays of objects arrays store reference to that objects into it rather then whole objects and reference is always of same memory hence arrays are contagious blocks of memory. I.e every element in the arrays occupies same amount of memory.

**Kafka Notes:**

Kafka solves problems of low throughput in database i.e where we need to insert say 10M records per sec

A blackboard with writing on it

Description automatically generated

It is designed to handle real-time data feeds and is widely used for building real-time data pipelines and streaming applications.

Unlike database kafka cannot store data permenently and run query to get data effeciently.