Name: Arun Kumar K

Java

Dictionaries in Java:

- Java doesn't have a direct equivalent to Python's dictionary type, but the Map interface and its implementations (HashMap, TreeMap, LinkedHashMap, Hashtable) serve as the dictionary-like data structure.
- HashMap is the most common and widely used Map implementation for general-purpose dictionary-like use cases, providing fast lookups and flexible handling of keys and values.

Graph:

- In Java, a graph is a data structure that consists of a set of nodes (also called vertices) and edges that connect these nodes.
- Graphs are widely used in various fields like computer networks, social networks, routing algorithms, etc.
- A graph can be represented in Java using various approaches, including using adjacency lists, adjacency matrices, or edge lists.

Interface And Class:

Class:

- A class is a blueprint or template for creating objects.
- It defines the properties (fields) and behaviors (methods) that the objects of the class will have.
- A class can also implement multiple interfaces and extend one class.

Interface:

- An interface is a contract that defines a set of abstract methods (methods without implementation) that a class must implement.
- Interfaces are used to represent behaviors or actions that can be shared across different classes.

Why interface is base:

- An interface is often considered a "base" because it serves as a foundational contract that defines a common set of behaviors or capabilities that multiple classes can adopt.
- It promotes polymorphism, decoupling, flexibility, and extensibility in your code, allowing different classes to adhere to a shared set of rules while retaining their individual implementation details

throw (Used to throw an exception):

- The throw keyword is used to explicitly throw an exception from a method or block of code. When you throw an exception, you are creating an instance of an exception class and passing it to the runtime environment.
- It can be used inside a method, constructor, or any block of code where an exception needs to be thrown due to some condition or error.

throws (Used to declare exceptions a method can throw):

- The throws keyword is used in a method signature to declare that the method might throw one or more exceptions. It tells the caller of the method that it is responsible for handling or declaring these exceptions.
- throws is used in the method declaration to specify that the method might throw one or more exceptions but doesn't handle them directly.

Why HashMap is not in collection:

- In Java, HashMap is not part of the Collection interface, and this design choice is due to the specific role HashMap plays in the Java Collections Framework.
- Here’s a detailed explanation of why HashMap is not a direct subclass of Collection

Interfaces Name:

→ Comparable
→ Clonable
→ Drawable
→ Writable
→ Readable
→ Loggable
→ Inspectable
→ Validatable
→ Observing
→ Flyable

→ Cancelable
→ Schedulable
→ Identifiable
→ Connectable
→ Executable

Why _able is in interface:

The suffix "able" in interface names is commonly used to denote that an object or class can perform a certain action or exhibit a specific behavior. It's a linguistic convention that helps make interface names self-explanatory, meaning they describe what the class is capable of doing.