

1.What do you mean by Generic?

Answer: Generic provides information for the compiler about the type of collection used. Hence, type checking is resolved automatically at run time. With the addition of autoboxing of primitive types, you can use generics to write simpler and more understandable code.

Example: `ArrayList<String> stringList = new ArrayList<String>();`

stringList only accept String type object. Which is detect by the compiler.

2.What is Threads?

Answer: A *thread* is the flow of execution of a single set of program statements. The *Thread Class* allows *multitasking* (ie running several tasks at the same time) by instantiating (ie creating) many threaded objects, each with their own run time characteristics.

3.What are the two ways of creating thread?

Answer: There are two way to create a thread:

- 1. using the thread class:** Define the thread by writing your class that extends the Thread class and by overriding its run() method and instantiate the thread by instantiating your class. Then start the thread by executing the start() method.
- 2. Using the runnable interface:** Write the class that implements the Runnable interface, and implements the run() method of the Runnable interface. Then make an object of the thread class by passing instance in the argument of thread constructor of your runnable class. And start the thread by invoking the start() method.

4.what is the different between wait() and sleep()?

Answer: wait(): wait() is a method of Object class.wait() allows thread to release the lock and goes to suspended state. The thread is only active when a notify() or notifAll() method is called for the same object.

sleep(..) method is a static method of the class Thread. It puts the thread into sleeping state for the specified number of milliseconds. After the sleep

time expires, thread goes into the runnable state, and eventually is put back into the running state by the scheduler.

- **wait()**-It tells the calling thread to give up the lock and go to sleep until some other thread enters the same monitor and calls notify().
- **notify()**-It wakes up one single thread that called wait() on the same object. It should be noted that calling notify() does not actually give up a lock on a resource.
- **notifyAll()**-It wakes up all the threads that called wait() on the same object.

5.What is use of synchronized keyword?

Answer: When multiple threads access shared data, there is a possibility for the data to be corrupted. A synchronization mechanism is used to prevent these kinds of errors.

The use of the synchronized keyword for provides the Java programming language with a mechanism that enables a programmer to control threads that are sharing data.

public void synchronized run(){}; // class lock.

synchronized(this){}; //object lock.

6. What is deadlock?

Answer: When two threads are waiting for each other and can't proceed until the first thread obtains a lock on the other thread or vice versa, the program is said to be in a deadlock.

7.What is difference between ArrayList and vector?

Answer: Vector and ArrayList both class are implement the List interface. Vector and ArrayList both uses Array internally as data structure. Difference is in the way they are internally resized. By default, Vector doubles the size of its array when its size is increased. But, ArrayList increases by half of its size when its size is increased.

only main difference is, Vector's methods are synchronized and ArrayList's methods are not synchronized.

8. What is different between comparable & comparator interface?

Ans: **Comparable Interface:** The comparable interface are useful for ordering collection . The comparable interface imparts natural ordering to classes that implements it. The comparable interface is a member of the Java.lang package. That is used compareTo() method.

Comparator Interface: The comparator interface are useful for ordering collection . The comparator interface is used to specify order relation. comparator interface is a member of the Java.util package. That is used compare() method.

9. What is Stream?

Ans: A stream is a flow of data from a source to a sink. Sources called input stream & we can only read from it. Sinks called output stream & we can only write to a stream.

10. What is an event handler?

Answer: A function or method containing program statements that are executed in response to an event. An event handler typically is a software routine that processes actions such as keystrokes and mouse movements.

11. What is Adapter class?

Answer: An adapter class is a term for a class that implements a listener interface with methods that have no content, so they do nothing. The idea of this is to enable you to derive your own listener class from any of the adapter classes that are provided, and then implements just the methods that you are interested in.

12. What is Stream? How you classify them?

Answer: A stream is a flow of data from a source to a sink. Sources called input stream & we can only read from it. Sinks called output stream & we can only write to a stream.

There are two classify of stream:

- Byte Stream
 - InputStream
 - OutputStream
- Character Stream
 - Reader
 - Writer

13. What is Serialization & Deserialization?

Answer: Serialization: The process of writing an object to somewhere is called object serialization. The basic concept of object serialization is the ability to read & write objects to the byte streams. To make the objects of a class serializable, the class must implement the **Serializable** interface.

Deserialization: The process of reading a serialized object back into the program is called deserialization.

14. Write the Lifecycle method of a thread?

Answer: The lifecycle method of a thread is given below:

- New- (Instantiating an thread object.)
- Runnable-(By calling start from thread object add to thread schedule wait for run.)
- Running-(By calling run method.)
- Nonrunnable states
 - Blocked
 - Sleeping
 - Waiting
- Dead-(After complete the run method the thread become dead.)

15. What do you mean by MVC?

Answer: **MVC** is a design pattern used in most of the enterprise applications. The MVC (Model-View-Controller) architecture divides the web-based application into three parts:

The **model** that store the data.

The **view** that creates the visual representation of the component.

The **controller** that deals with user interaction.

16. What is Look And Feel? Write the default class of the Look and Feel.

Answer: The swing component provide a feature called pluggable look and feel that means it possible to change the appearance of a component. We can programmatically select the look and feel of a component. The pluggable look and feel has been facilitated by designing the classes in a particular way, called the Model-View-Controller architecture.

the default class of the Look and Feel.

- a. MetalLookAndFeel
- b. MotifLookAndFeel
- c. WindowsLookAndFeel
- d. WindowsClassicLookAndFeel

17. What are the type of events?

Answer: There are two type of Event:

- Low-Level Event (Source: Window, keyboard, mouse):
 - FocusEvent
 - MouseEvent
 - KeyEvent
 - WindowEvent
- Semantic Event (Source: button, menu, checkbox etc.):
 - ActionEvent
 - ItemEvent
 - AdjustmentEvent

18. What are the Listener?

Answer: The event object corresponding to the button click will be passed to any listener object that has previously registered an interest in the kind of event-a listener object being simply an object that listens for particular events. A listener is also called a target for an event.

19. What are the daemon threads?

Answer: daemon threads are created by the JVM. The daemon threads provides services to the user-created threads and they run in the background.

Hence to set them, have to write `threadname.setDaemon(true)` and to check whether a thread is a daemon thread or not you write `isDaemon()`. Daemon threads are terminated by the JVM when there are no longer any user threads running, including the main thread of execution.

20. What are the differences between Swing and AWT?

Answer: Swing is a more powerful alternative for awt. Swing is a set of classes that provides more powerful and flexible components than awt. The main difference between awt and swing is that swing components are not implemented by platform-specific code, instead they are written entirely in Java and therefore are platform-independent . The java.awt components depend on native code.

21. What is the difference between `yield()` and `sleep()`?

Answer: The **`yield()`** method is a static method of the class Thread. It must call in the following way – `Thread.yield()`; Yield places the calling thread into the runnable pool and allows another runnable thread to run.

The **`sleep(...)`** method is also a static method of the class Thread. It puts the thread into sleeping state for the specified number of milliseconds. After

the sleep time expires, thread goes into the runnable state, and eventually is put back into the running state by the scheduler.

22. What is collection? Write the type of collection.

Answer: A collection is a single object managing a group of object. A group of objects known as elements. Collection allows us to a specific ordering and duplicate are permitted.

There are two type of collection: **Set**-An unordered collection & no duplicate are permitted.

List-An ordered collection & duplicates are permitted.

Another type of collection : **Map**-Another collection of map which store pair keys against values.

23. What is buffer Stream? Why do you use?

Answer: Buffer stream is a high-level stream. We can read and write as a formatted data like as String by `readLine()` or `writeLine()` method.

We use it to reduce overhead cost/expense.

24. What is different between JFrame and JWindow?

Answer: The basic difference between a JFrame object and a window object is that a JFrame object represents the main window for an application, whereas a Window object does not. Always need a JFrame object before you can create a Window object.

JFrame: A resizable, movable window with title bar and close button. The entire application is usually a JFrame.

JWindow: A window without a title bar or move controls. The program can move and resize it, but the user cannot. It has no border at all. It optionally has a parent JFrame.

25. What is layout manager?

Answer: A layout manager is an object that implements the `LayoutManager` interface and determines the size and position of the components within a container. The components can provide size and alignment hints.

Exa: `FlowLayout`, `BorderLayout`, `CardLayout`, `GridLayout`, `GridBagLayout`, `BoxLayout` and `SpringLayout`.

26. What is the command-line-argument?

Answer: When the java technology program is launched from a terminal window, we can provide a program with zero or more command-line-arguments. These command-line-arguments allows us to a specify the configuration information for the application.

Exam: `java TestArgs arg1 arg2 "another arg"`

`args[0]` is 'arg1'

`args[1]` is 'arg2'

`args[2]` is 'another arg'

27.What is thread scheduler?

Answer: A thread cannot directly put into the running stage. By the `start()` method of a `Thread` class add a thread to runnable state under the thread scheduler. Then the thread scheduler a runnable thread put into the running state. The scheduler follows some kind of algorithm to determine which thread should go into the running state. There are two main categories of scheduling algorithms: a) Preemptive scheduling b) Time sharing.

28. Write down the basic features of Swing?

Answer: The basic features of Swing component are as follow -

- a. Supports pluggable look-and-feel for components.
- b. Support for tooltips.
- c. Support for automatic scrolling.
- d. Special debugging support.
- e. Component class can be easily extended.

29. What do you know about Container and Component?

Answer: Container: Container class is a subclass of the component class; every container object is a component, too. The container class is an abstract class, you cannot create instances of container.

Component: A component represents a graphical entity of one kind or another that can be displayed on the screen. A JFrame is a component. The General relationship between the group of classes that represent components.

30. What is the advantage of adapter class? Write down three adapter class.

Answer: An adapter class defines a set of empty methods for one or more low-level event interface. Can create a class defining a low-level event listener by deriving your class from an adapter class and then implementing the event-handling methods in which you are interest.

Three adapter class- i) FocusAdapter ii) WindowAdapter iii) KeyAdapter
iv) MouseAdapter

31. Life cycle of a Thread (Thread States)

1. [Life cycle of a thread](#)
 1. [New](#)
 2. [Runnable](#)
 3. [Running](#)
 4. [Non-Runnable \(Blocked\)](#)
 5. [Terminated](#)

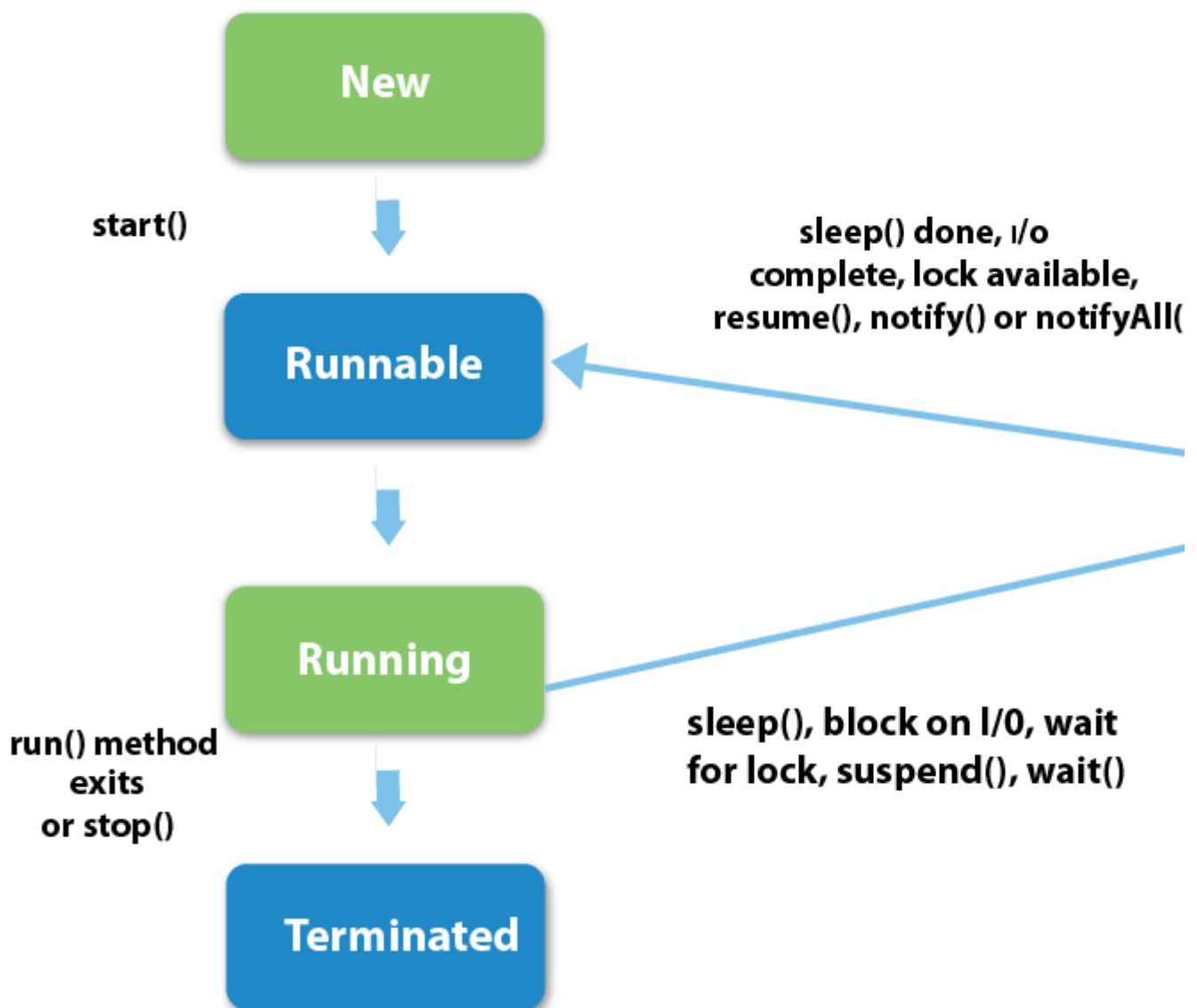
A thread can be in one of the five states. According to sun, there is only 4 states in **thread life cycle in**

java new, runnable, non-runnable and terminated. There is no running state.

But for better understanding the threads, we are explaining it in the 5 states.

The life cycle of the thread in java is controlled by JVM. The java thread states are as follows:

2. New
3. Runnable
4. Running
5. Non-Runnable (Blocked)
6. Terminated



1) New

The thread is in new state if you create an instance of Thread class but before the invocation of start() method.

2) Runnable

The thread is in runnable state after invocation of start() method, but the thread scheduler has not selected it to be the running thread.

3) Running

The thread is in running state if the thread scheduler has selected it.

4) Non-Runnable (Blocked)

This is the state when the thread is still alive, but is currently not eligible to run.

5) Terminated

A thread is in terminated or dead state when its run() method exits.

32. What is collection? Write the type of collection.

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

All the operations that you perform on a data such as searching, sorting, insertion, manipulation, deletion, etc. can be achieved by Java Collections.

The Java Collections Framework supports two types of containers:

- One for storing a collection of elements is simply called a *collection*.
- The other, for storing key/value pairs, is called a *map*.

Maps are efficient data structures for quickly searching an element using a key. We will introduce maps in the next chapter. Now we turn our attention to the following collections.

- **Sets** store a group of nonduplicate elements.
- **Lists** store an ordered collection of elements.
- **Stacks** store objects that are processed in a last-in, first-out fashion.
- **Queues** store objects that are processed in a first-in, first-out fashion.
- **PriorityQueues** store objects that are processed in the order of their priorities.