1) What is the difference between HashTable and HashMap?

A

Hash Map	Hash Table
1. Hash Map is Non-Synchronized	1. Hash Table is Synchronized
2. It is not thread safe and cannot be shared between many threads without proper Synchronization Code.	2. It is thread-safe and can be shared with many threads
3. Hash map allows null values & null key	3. It doesnot allow null key or value.

2) Difference between LinkedList & ArrayList

ArrayList	LinkedList
1. This class uses a dynamic array to store the element in it	1. This class uses a doubly-linked list to store the elements
2. Manupulation of Arraylists takes more time due to internal implementaion	2. Manupulating Linkend list. takes less time
3. This class implements a List Interface	3. This class implements both the List Interface & Dequee Interface
4. It provides random Access	4. It doesnot provides Random Access.

5. Difference between Arraylist & Vector

Arraylist	Vector
1. It is not synchronized	1. It is synchronized
2. It is not Legacy class	2. It is Legacy class.
3. It increases its sized by 50% of array size	3. It increases its size by doubling the array size.
4. Arraylist is not thread Safe.	4. It is thread Safe.

6. Difference between List, Set & Map

Interface	Duplicates Allowed?	Null Values Allowed?	Insertion Order preserved?	Iterator	Data Structure
List	Yes	Yes, Multiple null values are allowed	Yes can retrive using index	Iterator, ListIterator	Array
Set	No	Yes but only once	No	Iterator	Underlying Map
Map	Not for keys	Yes but only once for keys Can have multiple null Values	No	Through keyset, Values & entrySet	Hashing Teching

5) When to go for List

Ans: If you need to access elements frequently by using the index that List is a way to go & to maintain order. It provides faster access if you know index.

6) When to go for Map

Ans: If you store data in form of key and value the Map is the way to go

7. When to go for Set.

Ans. If you want to creat a Collection of unique elements and don't want any duplicate than we can choose Set Interface

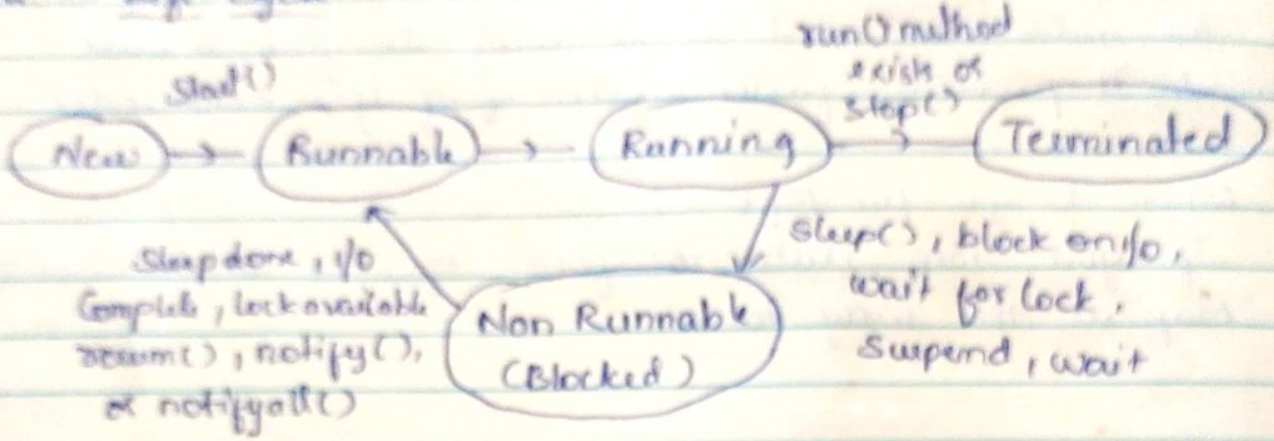
8 What is the difference btw extending Thread class & implementing Runnable interface

Ans:

1. When we extend Thread class we can't extend any other class
   When we implement runnable, we can save a space for our class to extend any other class in future
2. When we extend Thread class, each of our thread creates unique object and associate with it
   When we implement runnable, it shares the object to multiple threads

⑨ Thread Life cycle with all Methods

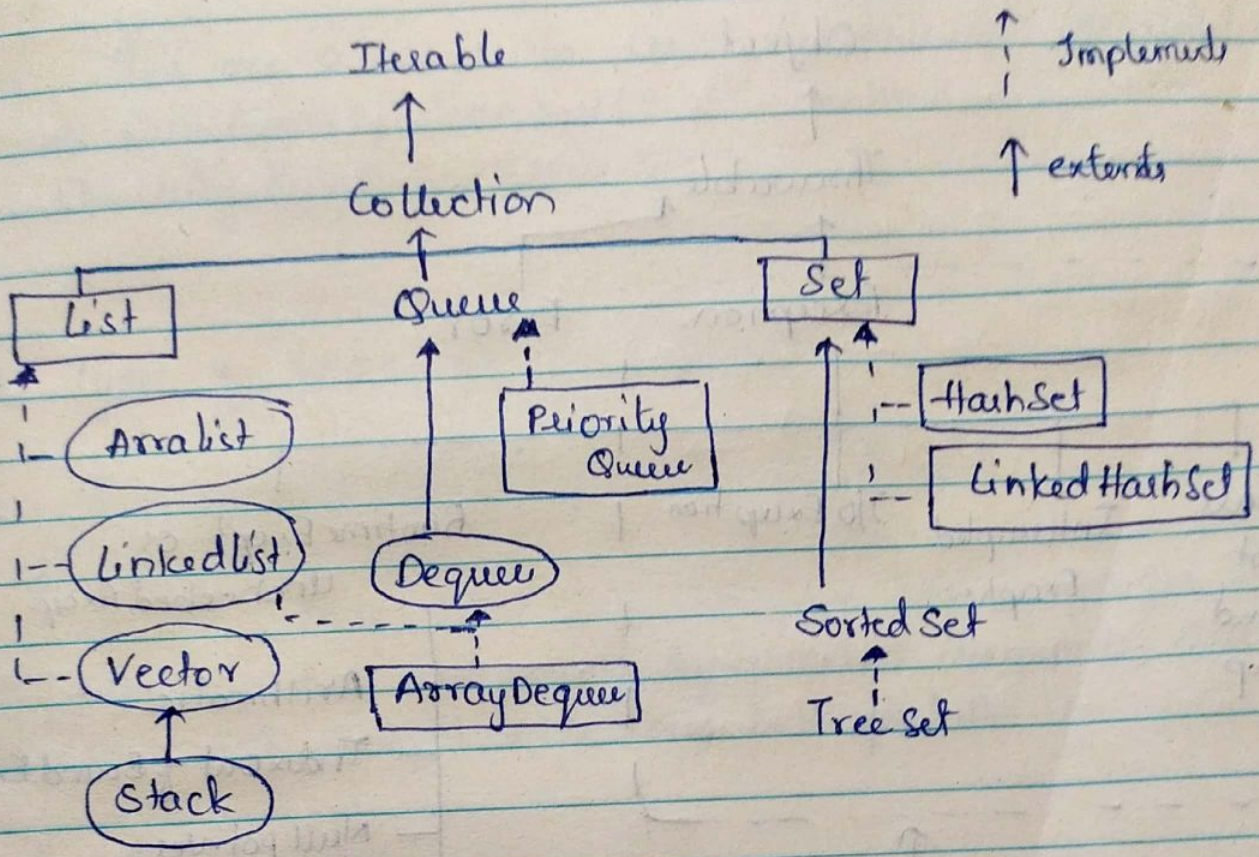
Ans Life cycle



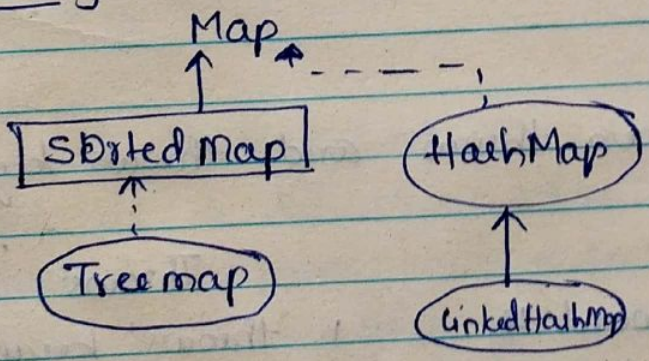
Methods in Thread class:

1) public void run(): used to perform action for a thread
2) Public void start(): starts the execution of thread, JVM calls run() method on the thread
3) public void sleep(long ms):
4) public void join(): ~~word~~ ~~to join~~ waits for a thread to die
5) public void join(long ms):
6) public int getPriority():
7) public int setPriority(int Pri)
8) public string getName():
9) public String setName(String name)
10) public Thread CurrentThread()
11) pulic int getId()
12) public thread state getState()
13) public boolean isAlive()
14) public void yield(): temporarily pause a current thread execution & allow other thread to execute

10) Collection Hierachy

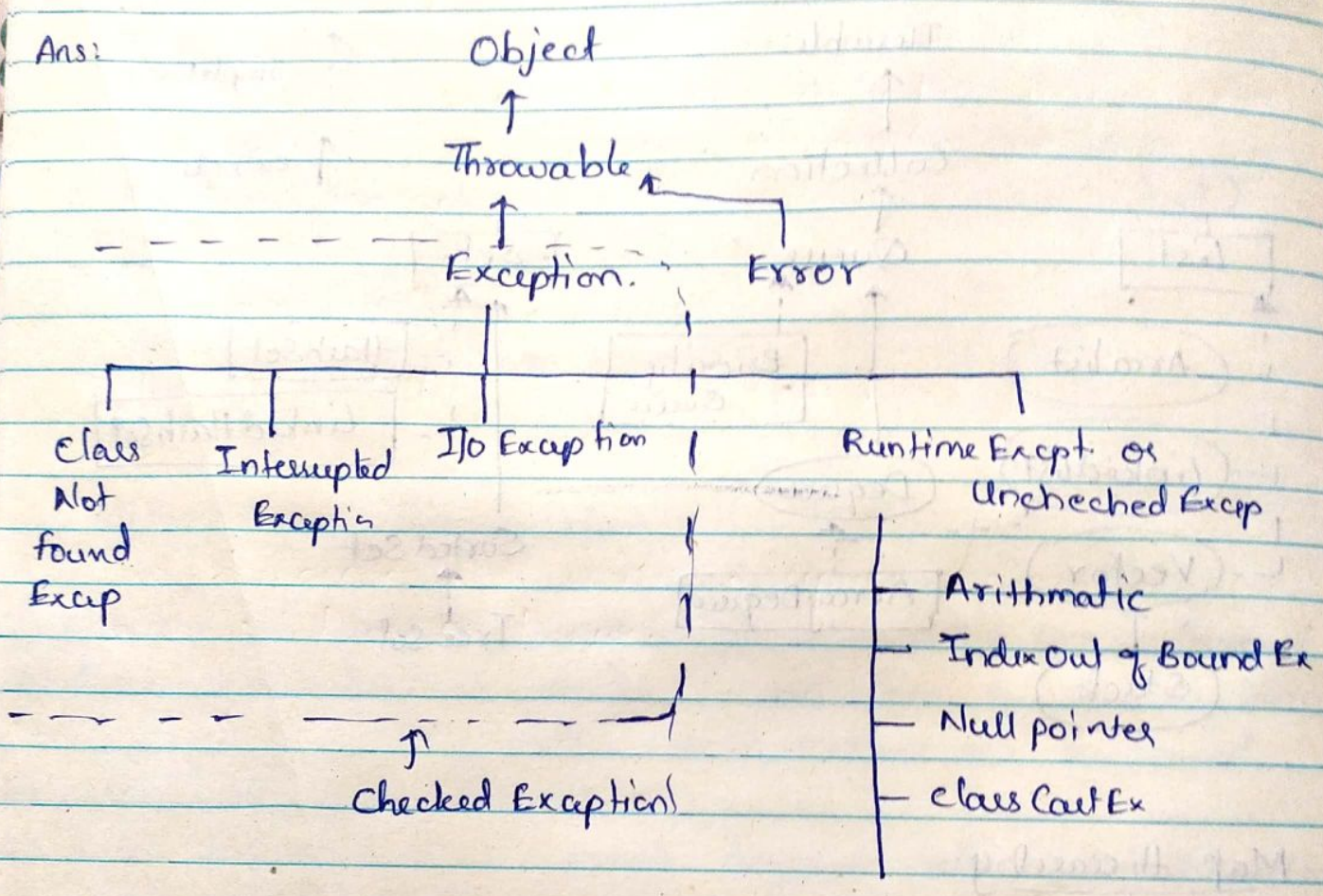


Map Hierachy:



11) Exception Hierarchy.

Ans:



12) Difference between throw and throws keyword

Throw	Throws
1. Java 'throw' keyword is used to explicit throw an exception	1. 'throws' keyword is used to declare an exception
2. checked except^n cannot be propogated using throw	2. checked Exception can be propropagated with throws
3) It is followed by Instance	3. followed by class.
4) It is used with in Method.	4. Throws is used with the Signature

13) How to make ArrayList Synchronized?

Ans: There are 2 ways to Create Synchronized ArrayList
1) Collections.synchronizedList() method
2) Using CopyOnWriteArrayList.

14) Types of locks in Java Thread

Ans: There are two types of locks on threads:
1) Object Level Lock:
Every object lock in java has a unique lock. whenever we are using Synchronized keyword, then only lock concept will come in to picture

2. Class Level Lock:
Every class in java has a unique lock which is nothing but class level lock.

15) What is deadlock Situation?

Ans: Deadlock is a situation where a set of processes are blocked because each process is holding a resource and waiting for another resource. aquired by some other process.

16) Types of Interface in java.

Ans) Class, AbstractClass, Nested Class, Enum, Dynamic proxy.

17) What is Lambda Expression?

A. A Lambda Exp is short block of code which takes in parameters. and return a value.

It is similar to methods, but they do not need a name & they can be implemented right in the body of a method

Eg. Syntax:

parameter → expression

(parameter 1, par 2) → { Code block }

18) How to handle null pointer exception without using try-catch or throw or throws

Ans. To handle null pointer exception without try catch or throw or throws we use below concept.

1) Ternary Operator
2) Use Apache Common 'StringUtils' for String Operations
3) Check Method Argument for null very early
4) Consider primitives Rather than Objects
5) Use String.valueOf() Rather than toString().
6) Avoid returning null from your method.
7) Call String.equals(String) on 'safe' Non-null String.