1. **Maven** - a tool that can be used for building and managing any Java-based project.

- Types of Repositories:

local - /.m2 (C:/Users/superdev/.m2)

central - is located at <http://repo.maven.apache.org/maven2/>

Whenever you run build job, maven first try to find dependency from local repository.

remote – belong to specific company such as Apple and Facebook, any developer in the group can download the repository by remote.

(Git Hub as our remote repository)

add dependency on the pom.xml file. (Maven -> refresh)

- life cycle: validate, compile, test, package (jar), verify, install, deploy. (7 steps)

command line: mvn clean, mvn test, mvn install …

2. **Git** - distributed version control system, handle projects with speed and efficiency.

- 2 ways to push: IntelliJ: Version Control -> Create Git repository …

Terminal: git add .

git commit -m "add read me file"

git push

3. Eight Basic **Data Types**

- primitive type:

byte, short, int, long, float, double, char, boolean

- wrapper class -

Byte, Short, Integer, Long, Float, Double, Character, Boolean

- pri -> wrap: autoboxing, wrap -> pri: unboxing.

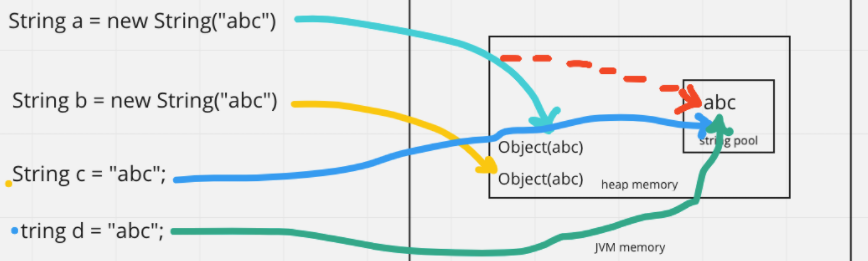
4. **String**/StringBuilder/StringBuffer

- String: **immutable**, thread safe ( + “”; // wouldn’t change content)

- StringBuilder: mutable, not thread safe (.append())

- StringBuffer: mutable, thread safe (.append())

- String constant pool



EX: every primitive data type (Wrapper Class) has constant pool.

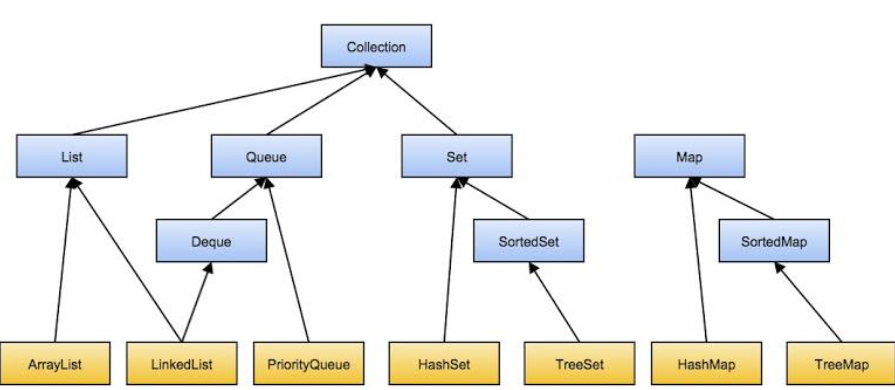
Integer limited range -127~128. Over range, different reference.

- == vs equals(): == compare reference, equals() compare content.

5. **equals** / **hashcode**

- override for return true value of equal(), or just shallow equal (reference).

6. **Collection**



- List : allow duplicated elements. Insertion Order

ArrayList (insert/remove takes O(n) time, retrieve data O(1) since random access)

LinkedList (insert/remove takes O(1) time, retrieve data O(n))

- Set : unique elements

HashSet (don’t keep insertion order)

LinkedHashSet (keep insertion order)

TreeSet (sorted)

- Deque:

ArrayDeque: first [ ] last

replace Stack: deque.offerFirst(), deque.pollFirst()

push(), , pop()

- vector & stack (deprecated)

thread safe, push pop (stack), FILO (stack)

- Map: (not implements Collection)

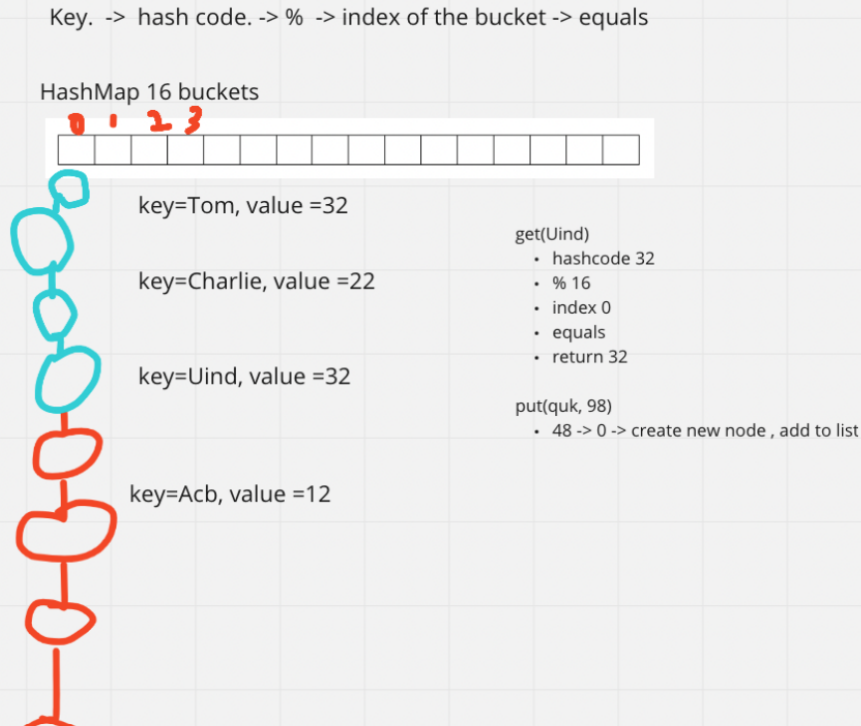
HashMap: not thread safe

**create 16 buckets -> hash(key) generates hashCode -> %16 to cal idx**

**-> equal(key) -> put or get val of key.**

**Same bucket using separate chaining. (Linked List) add new data to the “first” position.**

OR, open addressing which add data to next bucket (NOT in JAVA)

****

After Java 8, size of linked list (O(n)) over 8, will transform to red-black tree to improve performance (O(logn))

LinkedHashMap: Insertion Oder

TreeMap: Sorted (by Key)

HashTable: synchronized, thread safe (1 lock)

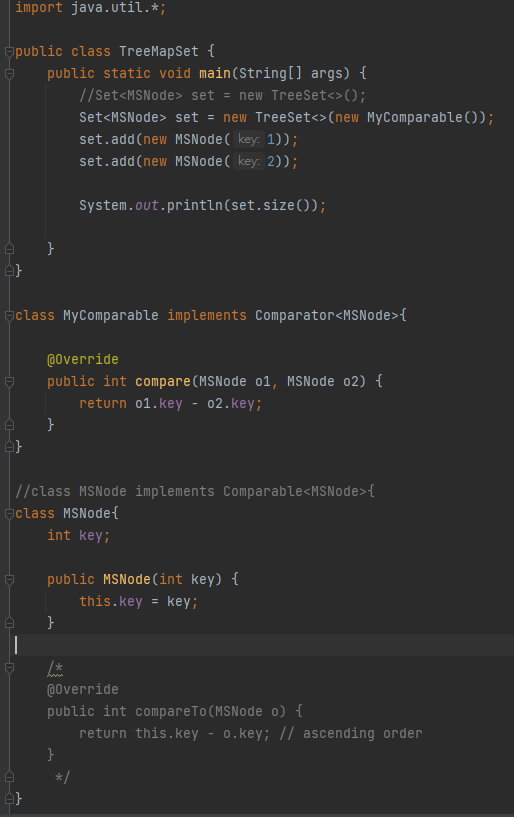
ConcurrentHashMap: synchronized, thread safe (16 lock)

- Queue : FIFO

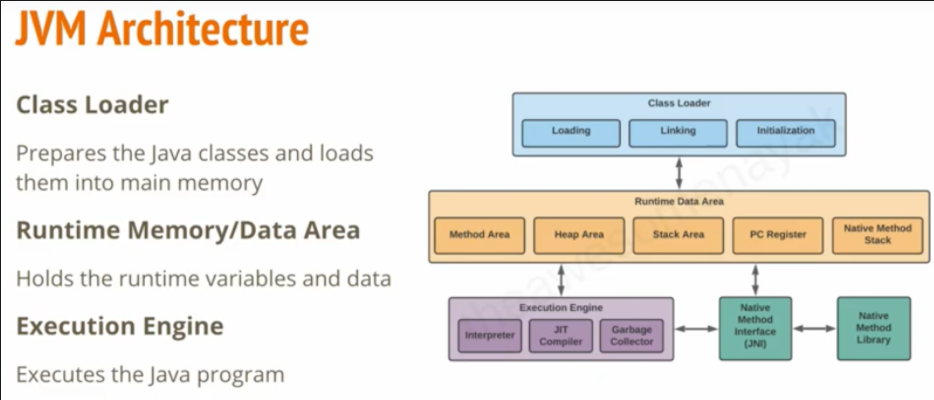
- Heap : PriorityQueue, maxHeap (parent node > child nodes, root max), minheap.

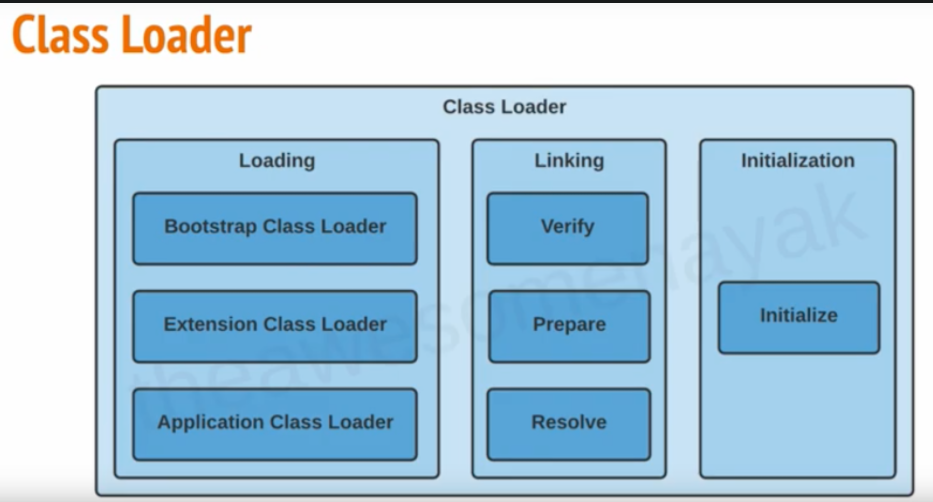
HashSet internal work by HashMap.

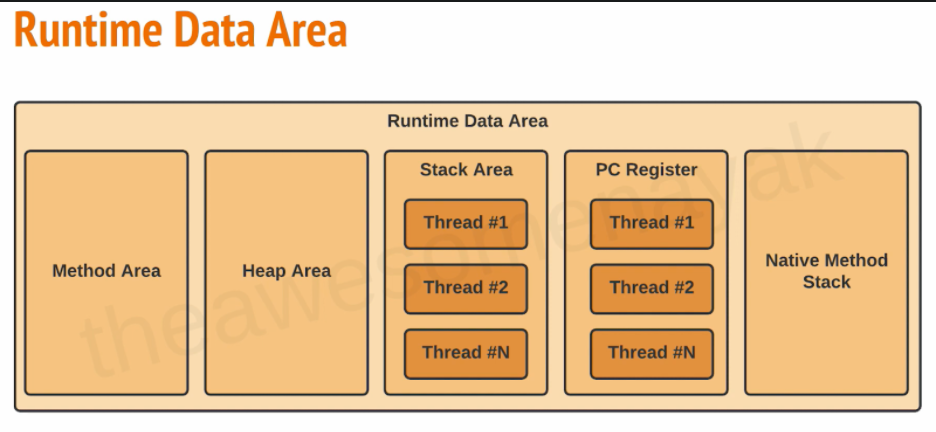
7. **Comparable** vs **Comparator**:



8. **JVM**







一張含有 文字 的圖片

自動產生的描述