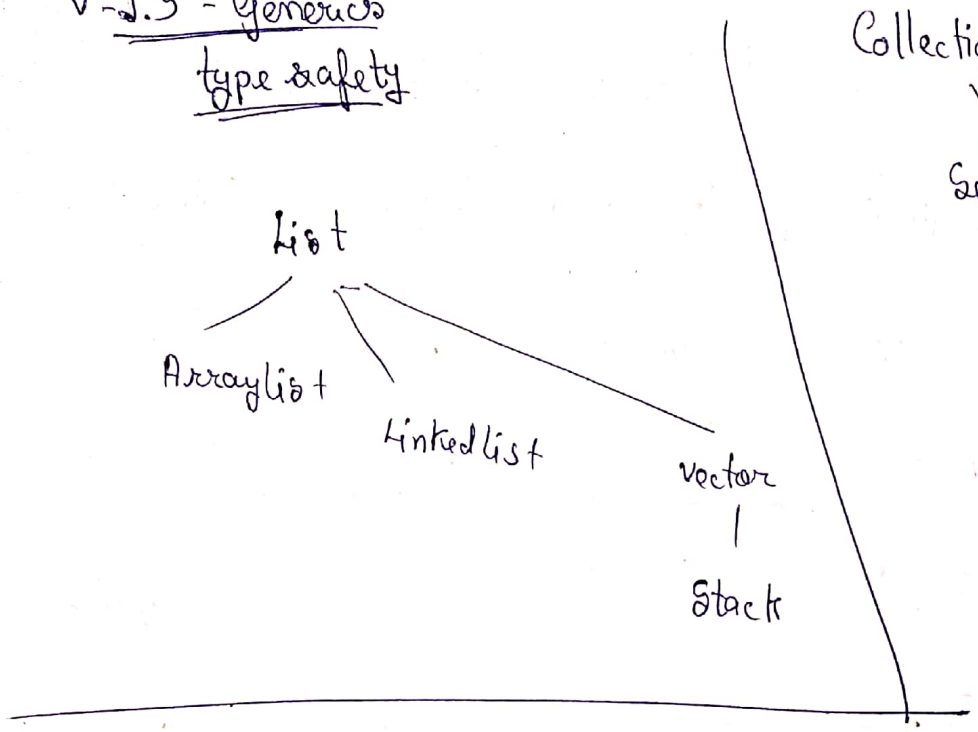
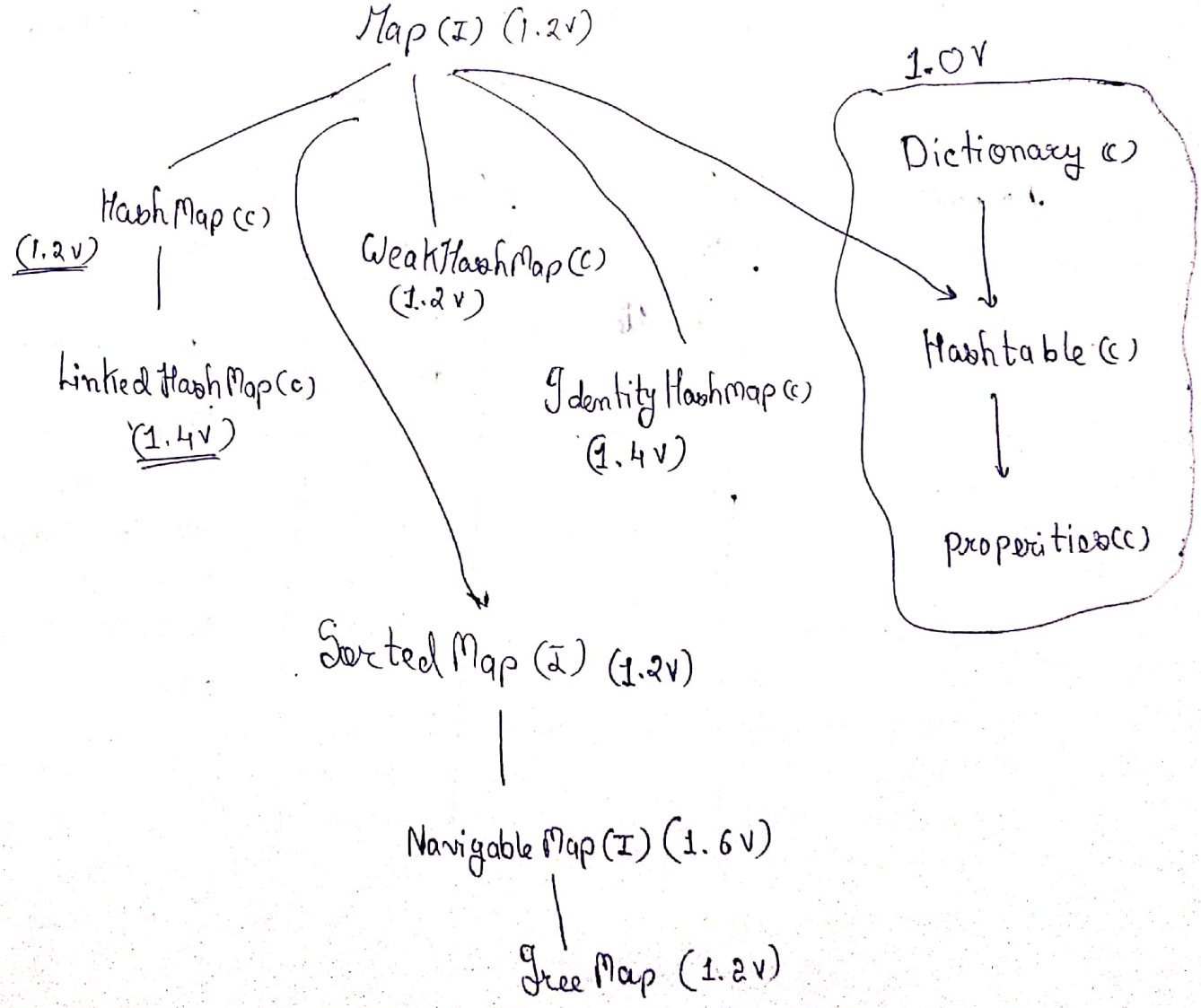
V-J.S - Generics

type safety



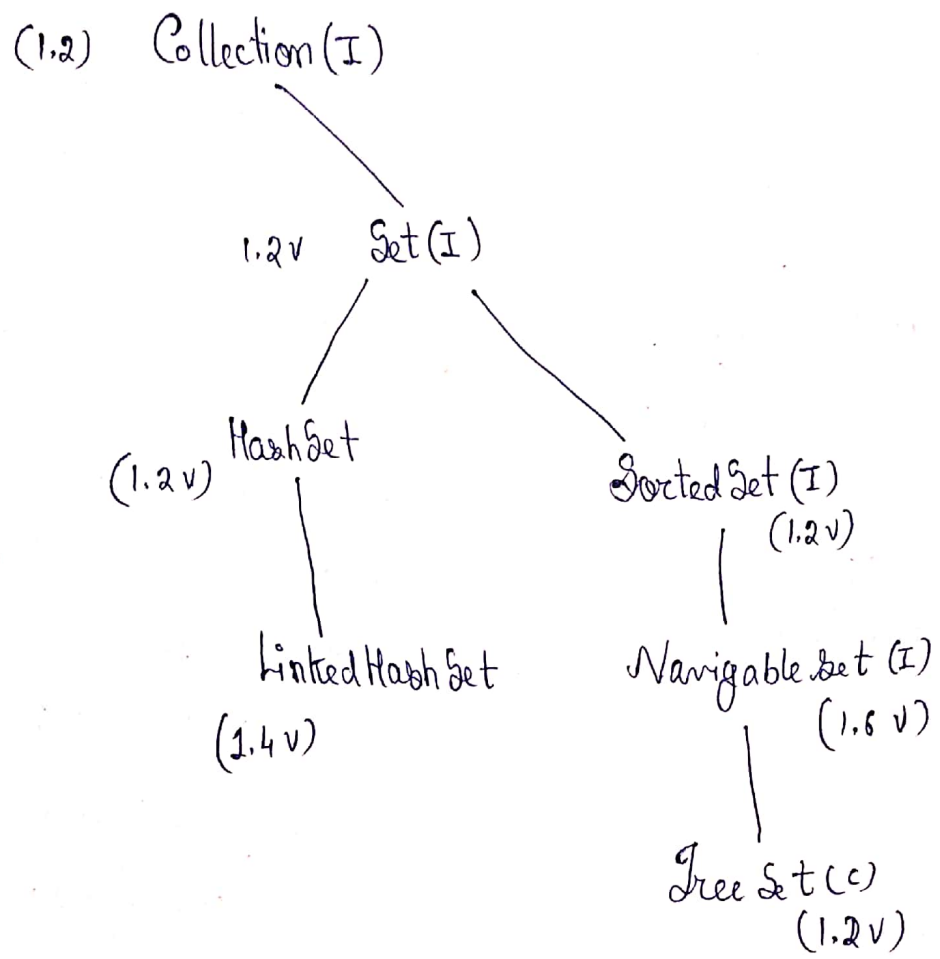


Scanned with Cam

Property	Enumeration	Iterator	ListIterator
Where to apply	legacy Class	Any collection object	Only for List Object
Is it legacy	yes (1.0 v)	No (1.2 v)	No (1.2 v)
Movement	Forward	Forward	Bidirectional
Accessible	only Read	Read, Remove	Remove, read, replace, add
How we can get	By using elements of vector class	By using iterator() of collection(I)	By using listIterator() of List (I).
Methods	(2) HasMoreElements NextElements()	(3) Has next() next() remove()	(9) Has next next next Index --- Has previous previous previous Index --- remove add

Scanned with Cam

Set (I)



	HashSet	LinkedHashSet
①	Hashtable	Linked list + Hashtable
②	Insertion order not preserved	Insertion order preserved
③	1.2v	1.4v

Scanned with Cam

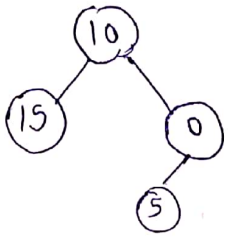
Reverse of alphabetical order.

→ Default sorting → compareTo()

SortedSet s = new TreeSet(new myComparator()); → Customized Sorting →

TreeSet t = new TreeSet(new myComparator());

```
t.add(10);
t.add(0);
t.add(15);
t.add(5)
```



compare(5,10)
+1
compare(5,0)
-1

```
Class myComparator implements
                    Comparator
{
  public int compare(obj1, obj2)
  {
    Integer I1 = (Integer) obj1
    Integer I2 = (Integer) obj2
     if (I1 < I2)
        return +1;
     else if (I1 > I2)
        return -1;
     else
        return 0;
    }
}
```

→
```
public int compare(Object obj1, Object obj2)
{
  Integer I1 = (Integer) obj1;
  Integer I2 = (Integer) obj2;
```

① return I1.compareTo(I2); // Ascending
② return -I1.compareTo(I2); // Descending
③ return I2.compareTo(I1); // Descending
④ return -I2.compareTo(I1) // Ascending
⑤ return +1; Insertion order [. . . . After.]
↳ Duplicate will allowed.

Scanned with Cam

Reverse of alphabetical order.

Default sorting → compareTo()

```
TreeSet t = new TreeSet(new myComparator);   → Customized sorting → compare(o1, o2)
t.add("Ra");
t.add("Sh");
t.add("Raja");
t.add("Ram");
t.add("Ramu");
```

```
class myComparator implements Comparator
{
  String s1 = (String) obj1;
  String s2 = (String) obj2;
  return s2.compareTo(s1);   // Desc;
}
```

Default Sorting → Comparable → method: compareTo();

Customized Sorting → Comparator → method: compare(obj1, obj2);

---

① Pre-defined Comparable class [String] ⇓ Comparator

② Pre-defined non-comparable class [StringBuffer] ⇓ Comparator

③ Our own class → Comparable (compareTo();) / Comparator (compare(,);)

Scanned with CamScanner

Comparable	Comparator
Default natural Sorting order	Customized sorting order.
Java.lang.	Java.util

Only one method
objs.compareTo(obj2)
	2 - methods
	compare(obj1, obj2);
	equals(obj1);

String class Wrapper Class	Only implemented classes - Collator - RuleBasedCollator } GUI classes

property	HashSet	LinkedHashSet	TreeSet
Data-Structure	Hashtable	LinkedList + Hashtable	Balanced Tree
Duplicate	Not allowed	Not allowed	Not allowed
Insertion order	Not preserved	preserved	Not preserved
Sorting order	N.A.	N.A.	Applicable
Heterogeneous Object	Allowed	Allowed	Not allowed.
Null acceptance	Allowed	Allowed	For empty TreeSet as first element null is allowed. Till 1.6 V only From 1.7 V not allo

Scanned with Cam

Same as ArrayList & Vector

	HashMap	HashTable
①	Not Synchronized	Synchronized
②	Not thread Safe	Thread Safe
③	performance is high	performance is low
④	Null key, value allowed	Null key, value not allowed
⑤	1.2 V Not legacy class	1.0 V Legacy class

Convert Non-Synchronized to Synchronized

- HashMap m = new HashMap();
- Map m1 = Collections.synchronizedMap(m);

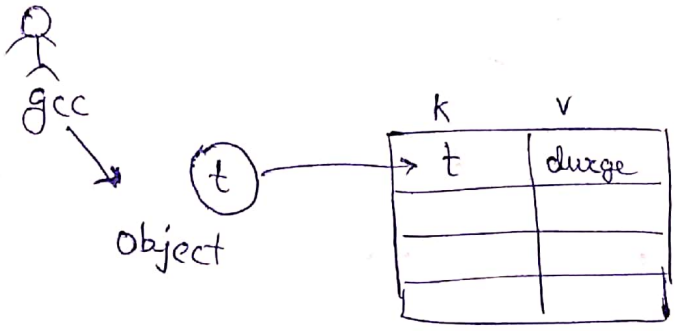
m1 → Synchronized.

m → Non-Synchronized

Scanned with Cam

Weak HashMap

→ Garbage collector → called finalize method.
in WeakHashMap.




```
Hashmap m = new HashMap();
Temp t = new Temp();
m.put(t, "durga");
Sopen(m);
```

```
class Temp
{
  public String toString()
  {
    return "temp";
  }
  Public void finalize()
  {
    Sout("Finalize method
          called");
  }
}
```

→ object does not have reference, even though gc garbage collector does not delete object bcoz object is related with HashMap.

```
t = null;
System.gc();
```

→ HashMap dominates Garbage Collector. (GC not called)

→ Garbage collector dominates WeakHashMap. (GC called. But after finalize method)

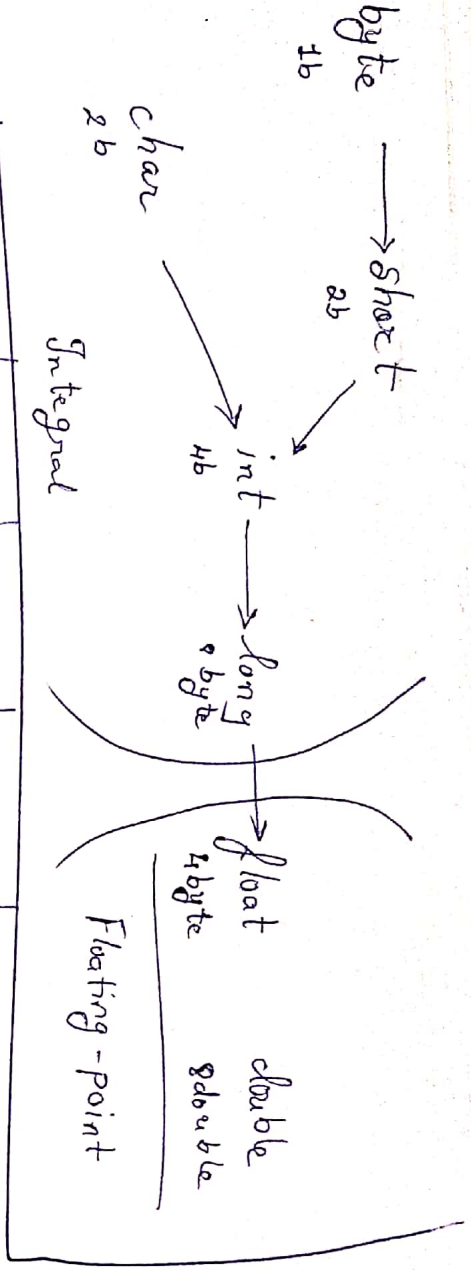
Scanned with Cam

length V/s length()

array — String

```
int[] a = new int[6];
```

Array Type	Allowed Element types
primitive Arrays.	Any type which can be implicitly promoted to declared type.
Object type Arrays.	Either declared type or its child class objects.
Abstract class type Arrays.	Its child class object.
Interface type Array.	Its implementation class objects are allowed.



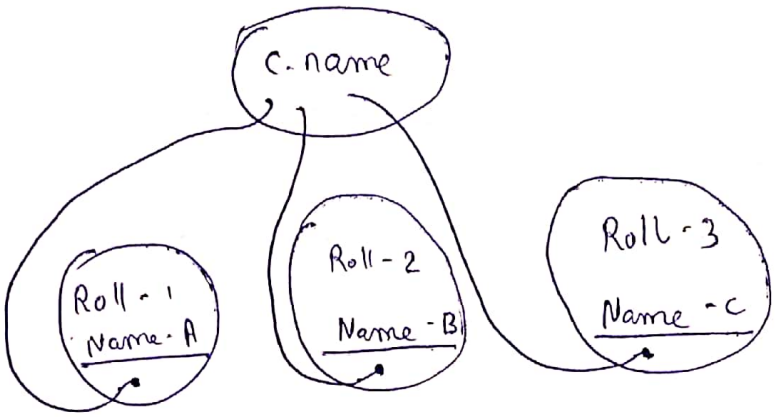
Scanned with Cam

→→ Instance variable

```
class Student {
    String name;
    int roll-no;
    static String c-name;
```

// If the c-name value is constant every time. Then use access modifier as static.

For every object only one copy will created.



→ Static variable will be created at the time of class loading. & destroy at the time of unloading.

→ Scope of static variable - Same as scope of object.

Java Test ↵

① Start JVM
② Create & Start main Thread
③ Locate Test.class file
④ Load Test.class → Static variable creation
⑤ Execute main() method
⑥ unload Test.class → Static variable destruction
⑦ Terminate main Thread
⑧ Shutdown JVM.

Scanned with CamScanner

① Static variable stored in method area

↳ local variable are stored in stack area.

```
class Test
{
    static int x = 10;
    PSVM (String[] args)
    {
        Sopln (x);
    }
    public void m1()
    {
        Sopln (x);
    }
}
```

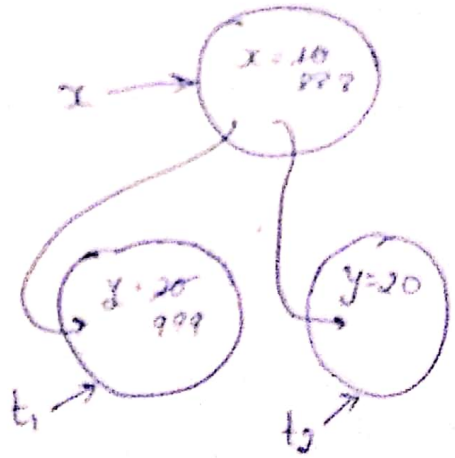
① Static variable can be access from both instance area and static area.

② for static variable JVM provide default value for instance variable.

---

②

```
class Test
{
    static int x = 10;
    int y = 20;
    PSVM (String[] args)
    {
        Test t1 = new Test();
        t1.x = 888;
        t1.y = 999;
        Test t2 = new Test();
        Sout (t1.x, t2.y)
    }
```



$t_1.x = 888$

$t_1.y = 999$

$t_2.x = 888$

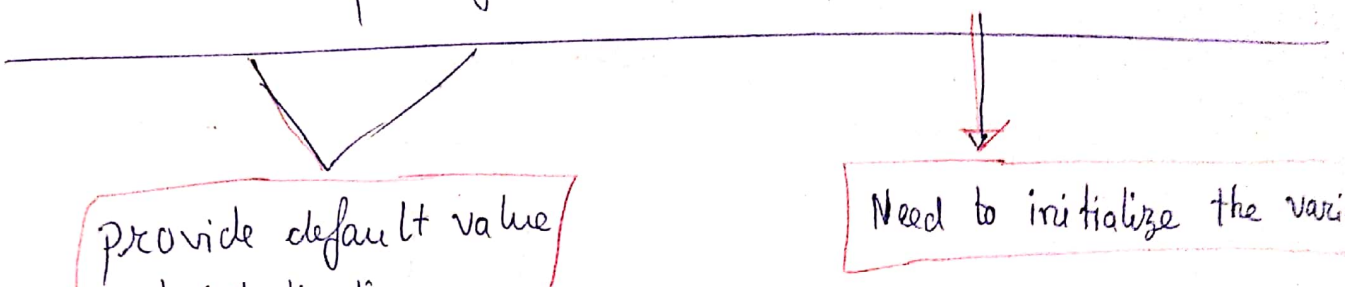
$t_2.y = 20$

Scanned with CamScanner

Static Variable → class level

Instance variable → object level

Local Variable → block level



provide default value at initialization

Need to initialize the variable.

Scanned with Cam