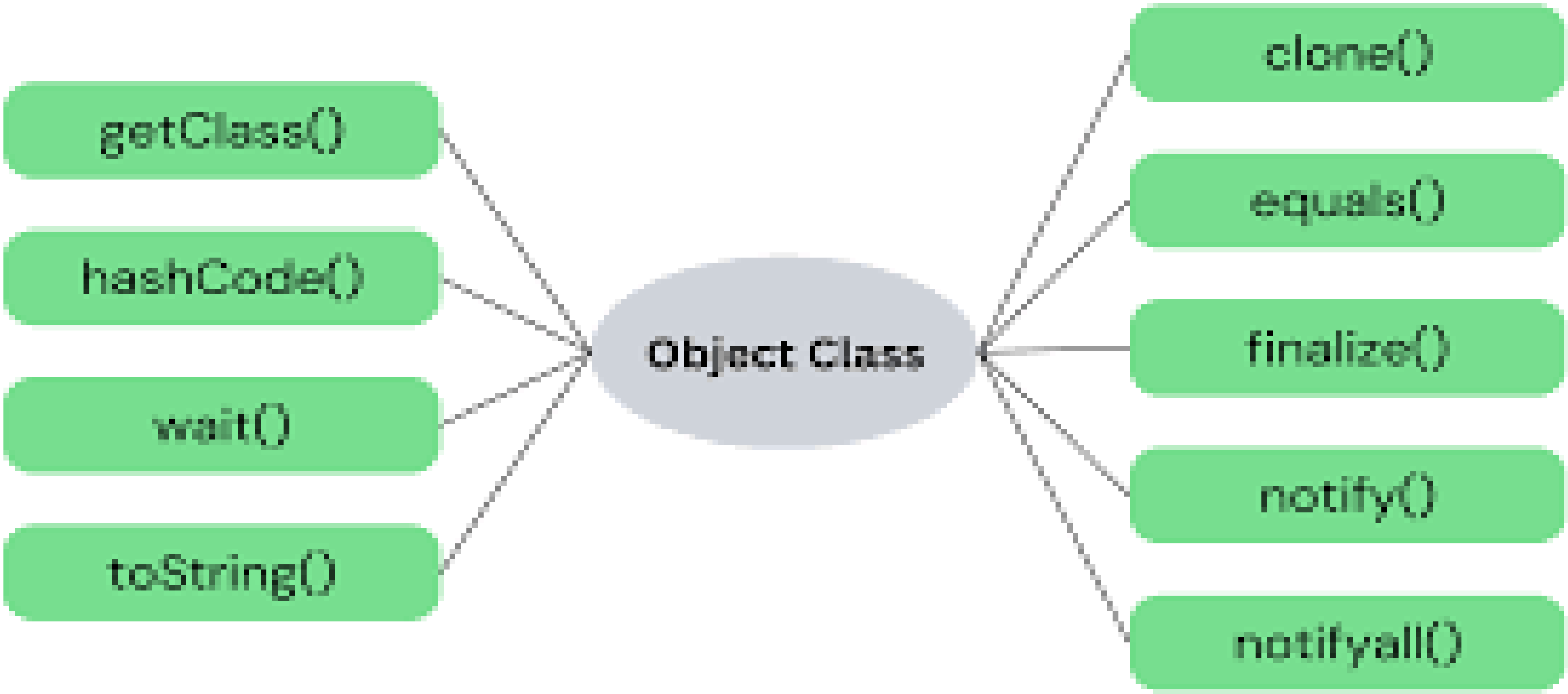
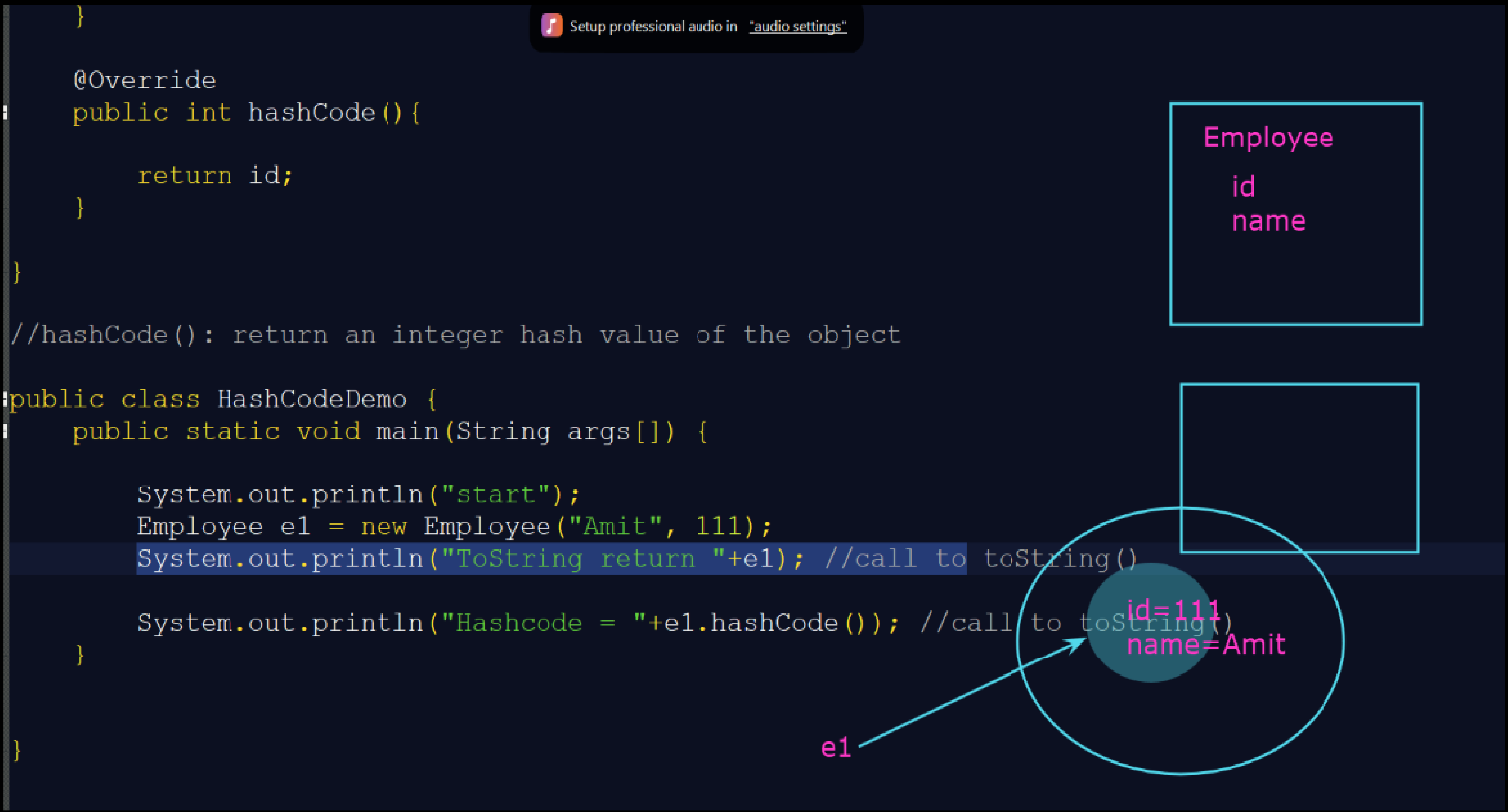
Object Oriented Programming with Java (OOPJ)

Session 13: Object class

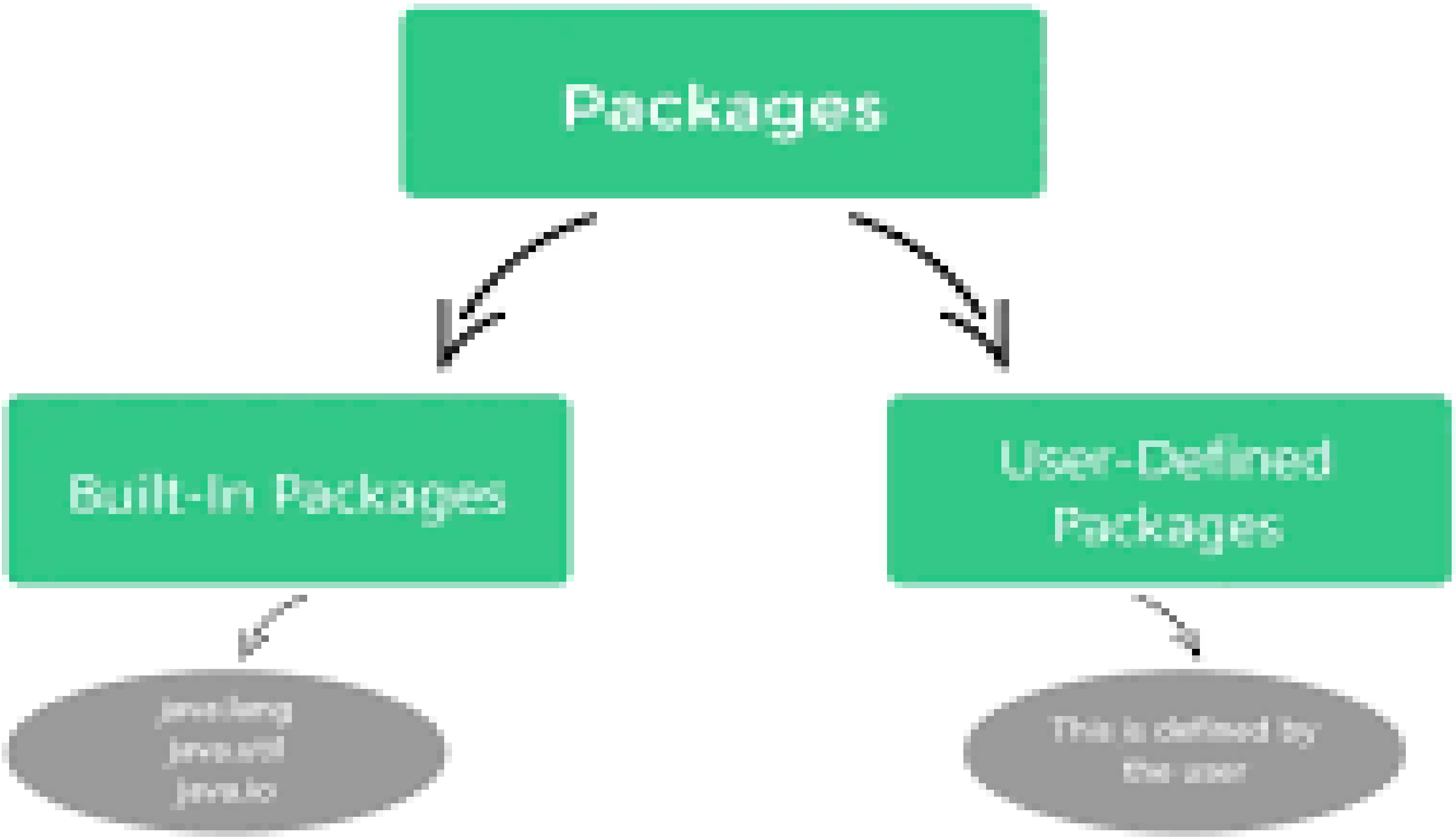
Kiran Waghmare


Object Class
getClass()
hashCode()
wait()
toString()
clone()
equals()
finalize()
notify()
notifyall()



```
    }

    @Override
    public int hashCode(){

        return id;
    }

}

//hashCode(): return an integer hash value of the object

public class HashCodeDemo {
    public static void main(String args[]) {

        System.out.println("start");
        Employee e1 = new Employee("Amit", 111);
        System.out.println("ToString return "+e1); //call to toString()

        System.out.println("Hashcode = "+e1.hashCode()); //call to toString()
    }


}
```

Packages in Java



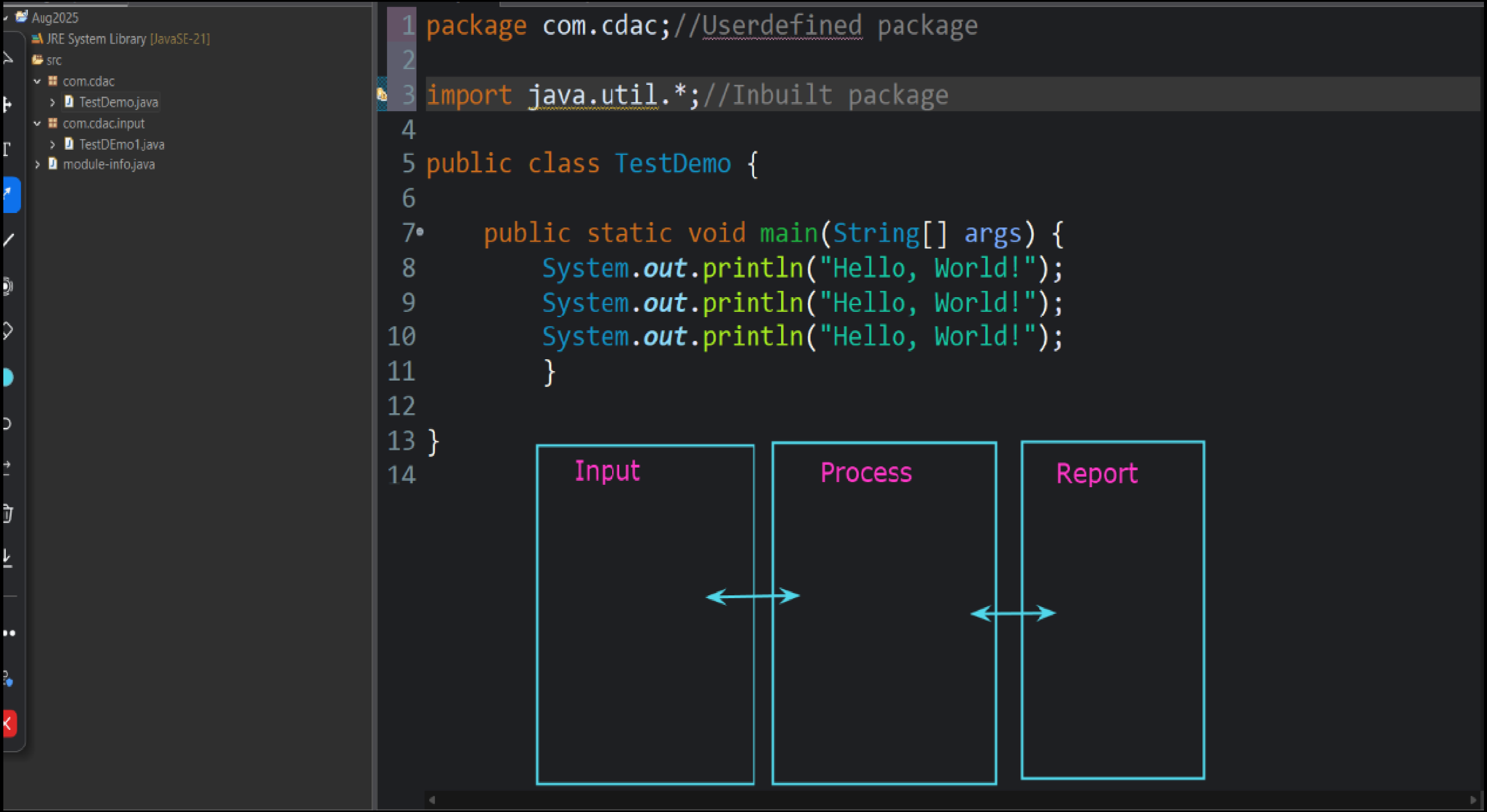
Packages

Built-In Packages

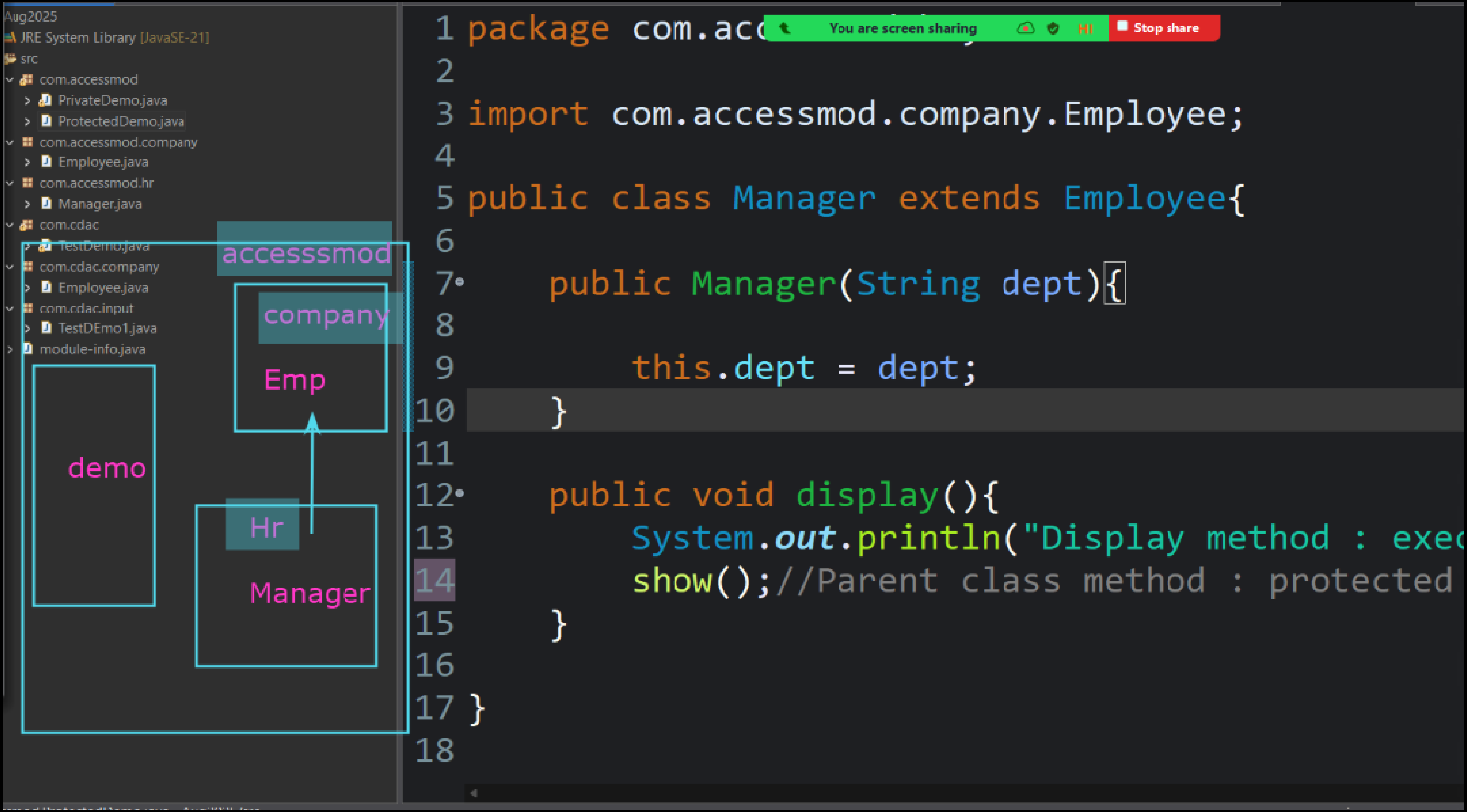
User-Defined Packages

java.lang
java.util
java.io

This is defined by the user

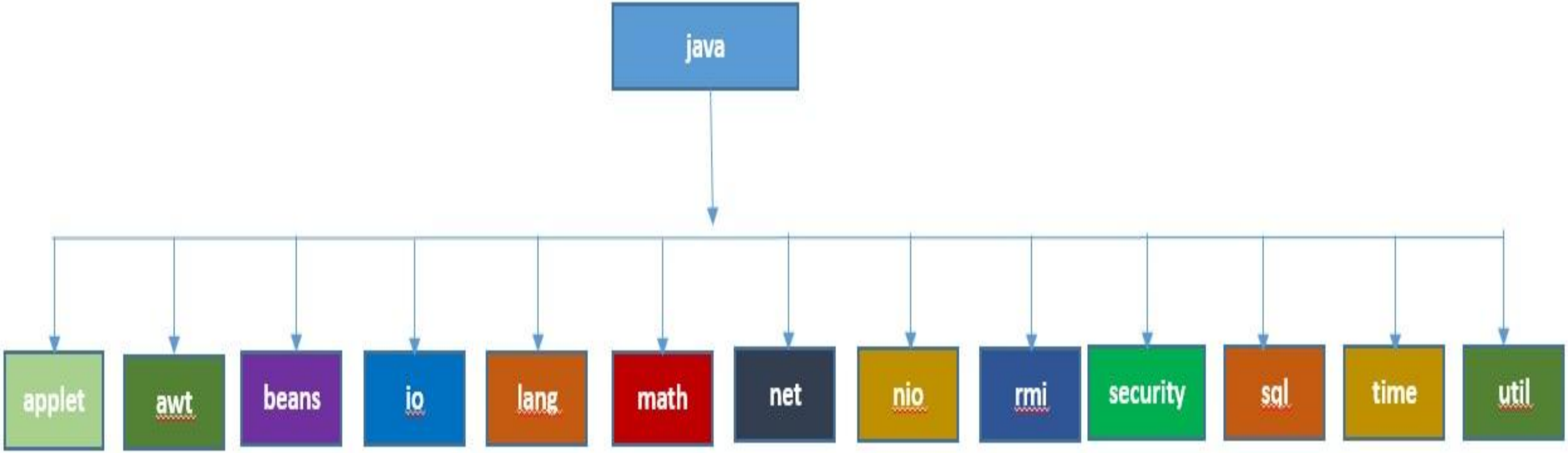


```java
package com.cdac;//Userdefined package

import java.util.*;//Inbuilt package

public class TestDemo {

    public static void main(String[] args) {
        System.out.println("Hello, World!");
        System.out.println("Hello, World!");
        System.out.println("Hello, World!");
        }

}
```



```
Aug2025
  JRE System Library [JavaSE-21]
  src
    com.accessmod
      PrivateDemo.java
      ProtectedDemo.java
    com.accessmod.company
      Employee.java
    com.accessmod.hr
      Manager.java
    com.cdac
      TestDemo.java
    com.cdac.company
      Employee.java
    com.cdac.input
      TestDEmo1.java
    module-info.java
```

```java
package com.acc

import com.accessmod.company.Employee;

public class Manager extends Employee{

    public Manager(String dept){

        this.dept = dept;
    }

    public void display(){
        System.out.println("Display method : exec
        show();//Parent class method : protected
    }

}

```



Java Most Common Use Built-in Packages

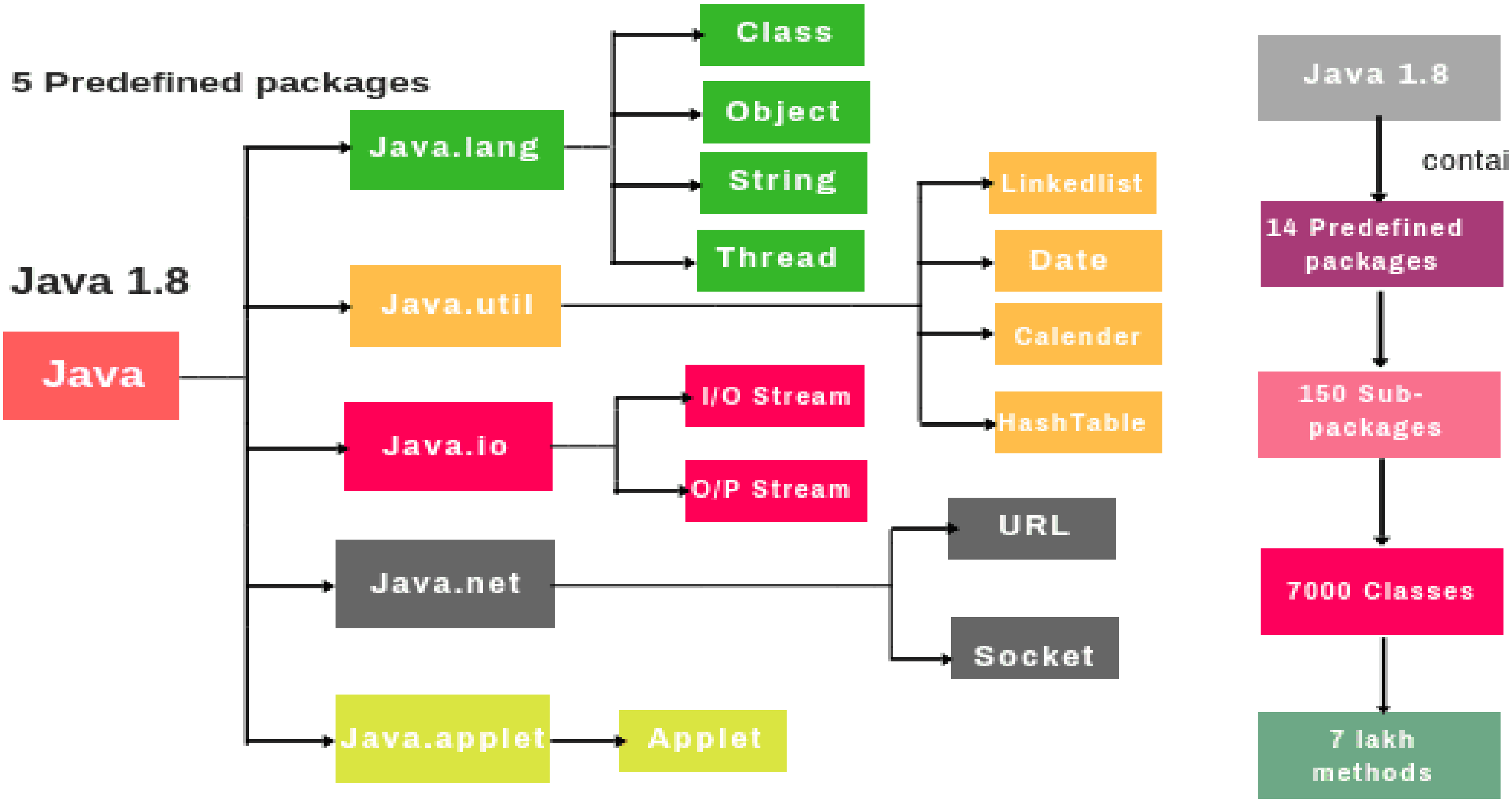
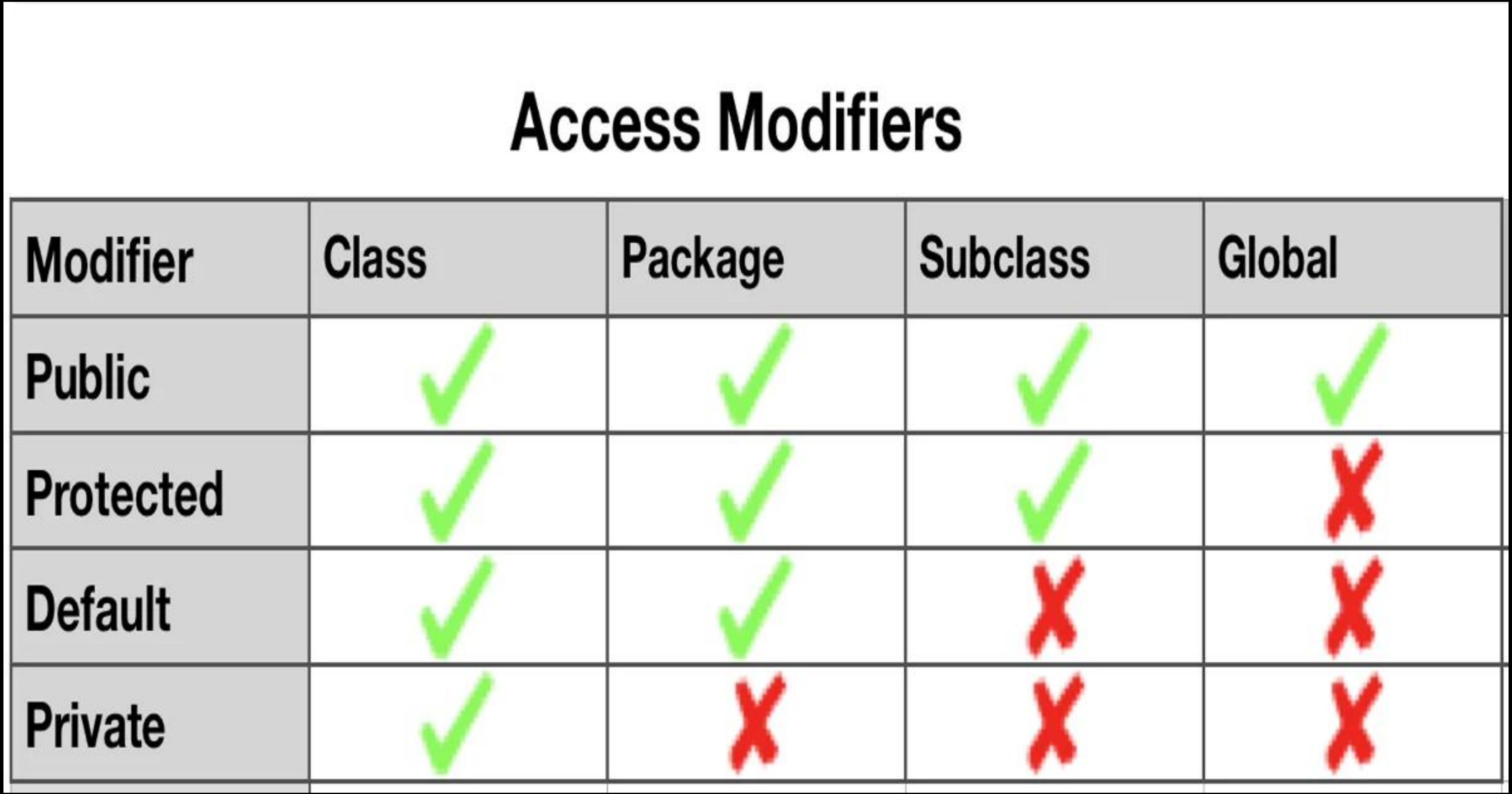
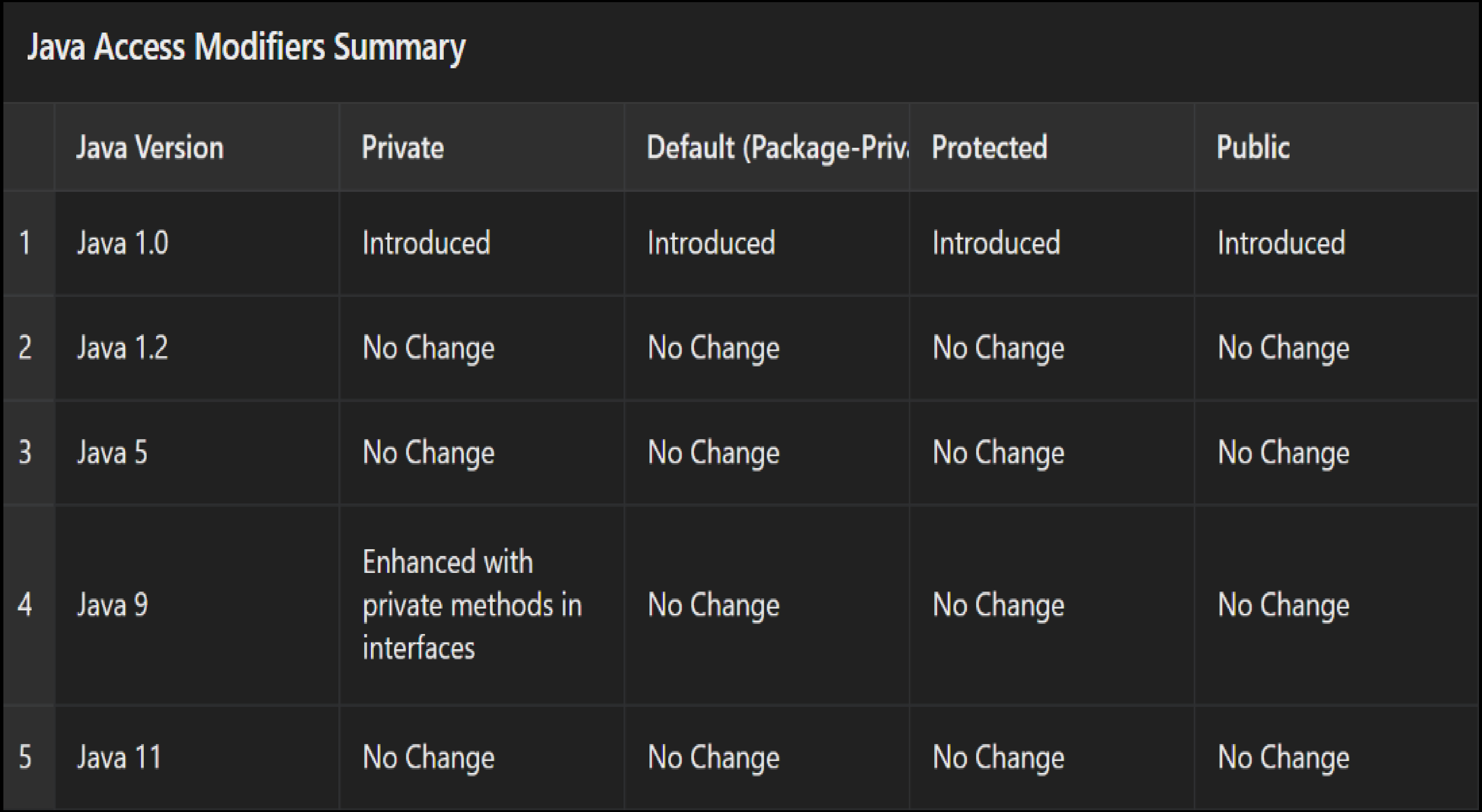


Fig: Predefined packages in Java



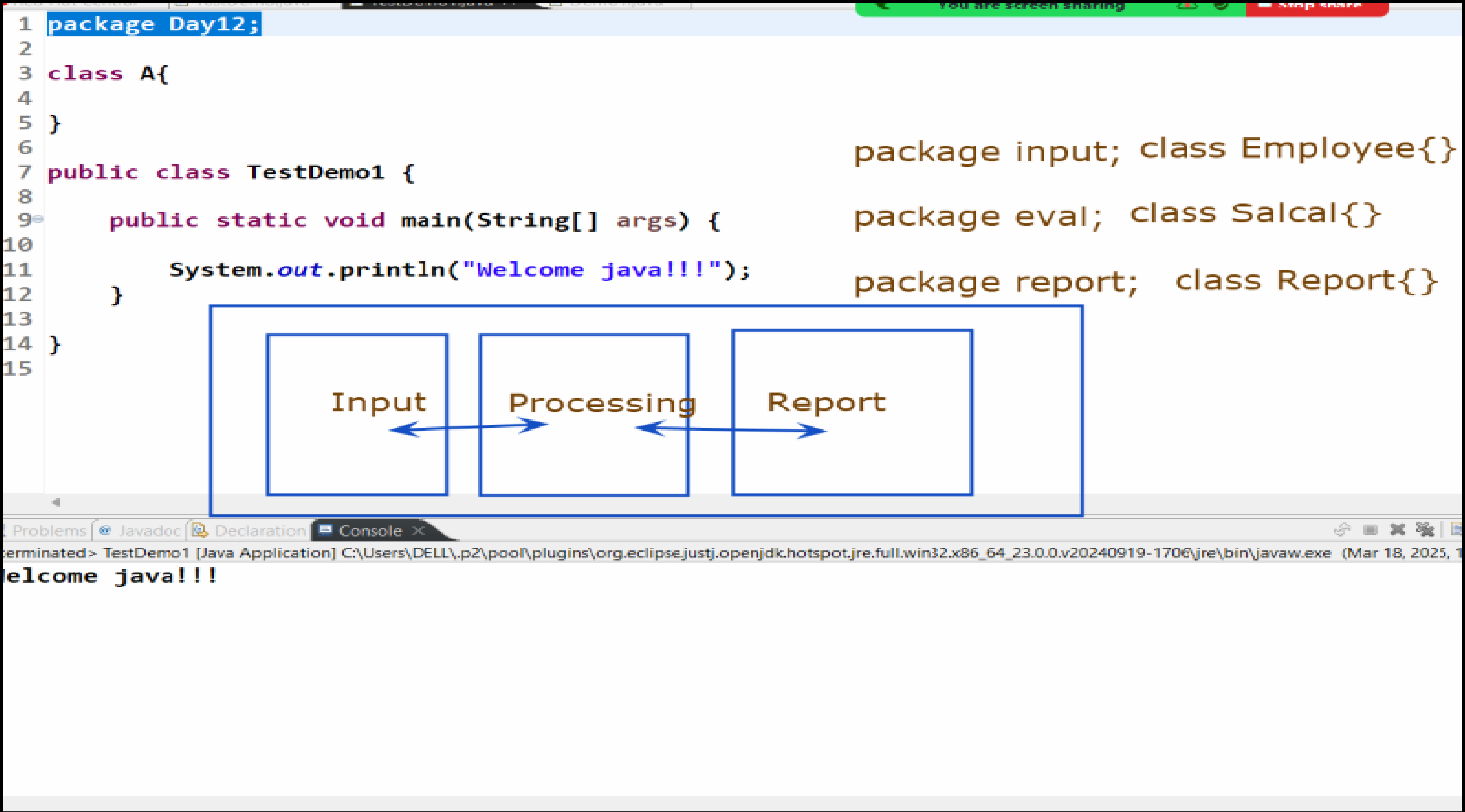
Access Modifiers

Modifier	Class	Package	Subclass	Global
Public	✓	✓	✓	✓
Protected	✓	✓	✓	✗
Default	✓	✓	✗	✗
Private	✓	✗	✗	✗



Java Access Modifiers Summary

	Java Version	Private	Default (Package-Priv	Protected	Public
1	Java 1.0	Introduced	Introduced	Introduced	Introduced
2	Java 1.2	No Change	No Change	No Change	No Change
3	Java 5	No Change	No Change	No Change	No Change
4	Java 9	Enhanced with private methods in interfaces	No Change	No Change	No Change
5	Java 11	No Change	No Change	No Change	No Change



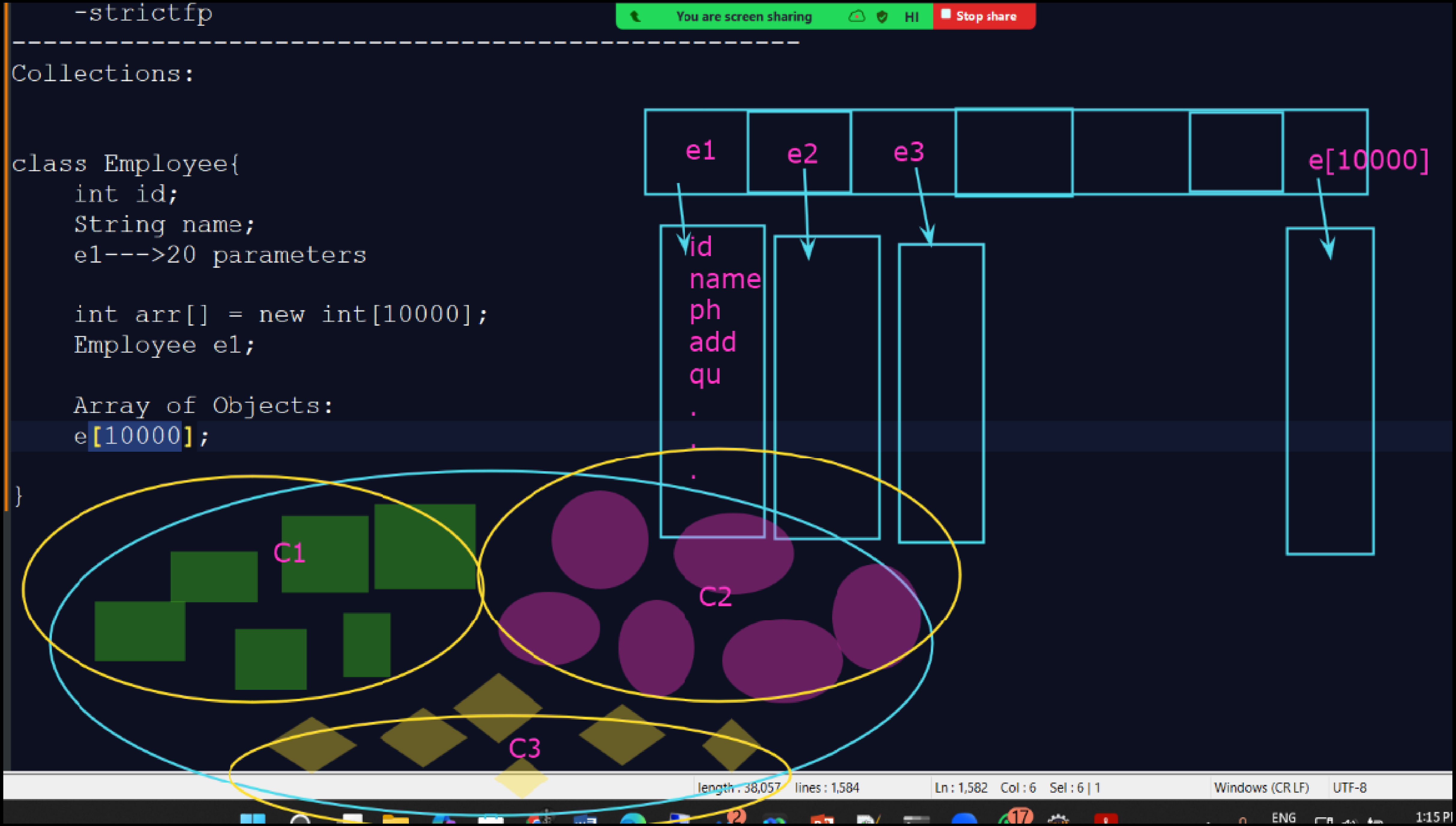
```java
package Day12;

class A{

}

public class TestDemo1 {

	public static void main(String[] args) {

		System.out.println("Welcome java!!!");
	}

}
```

package input; class Employee{}

package eval; class Salcal{}

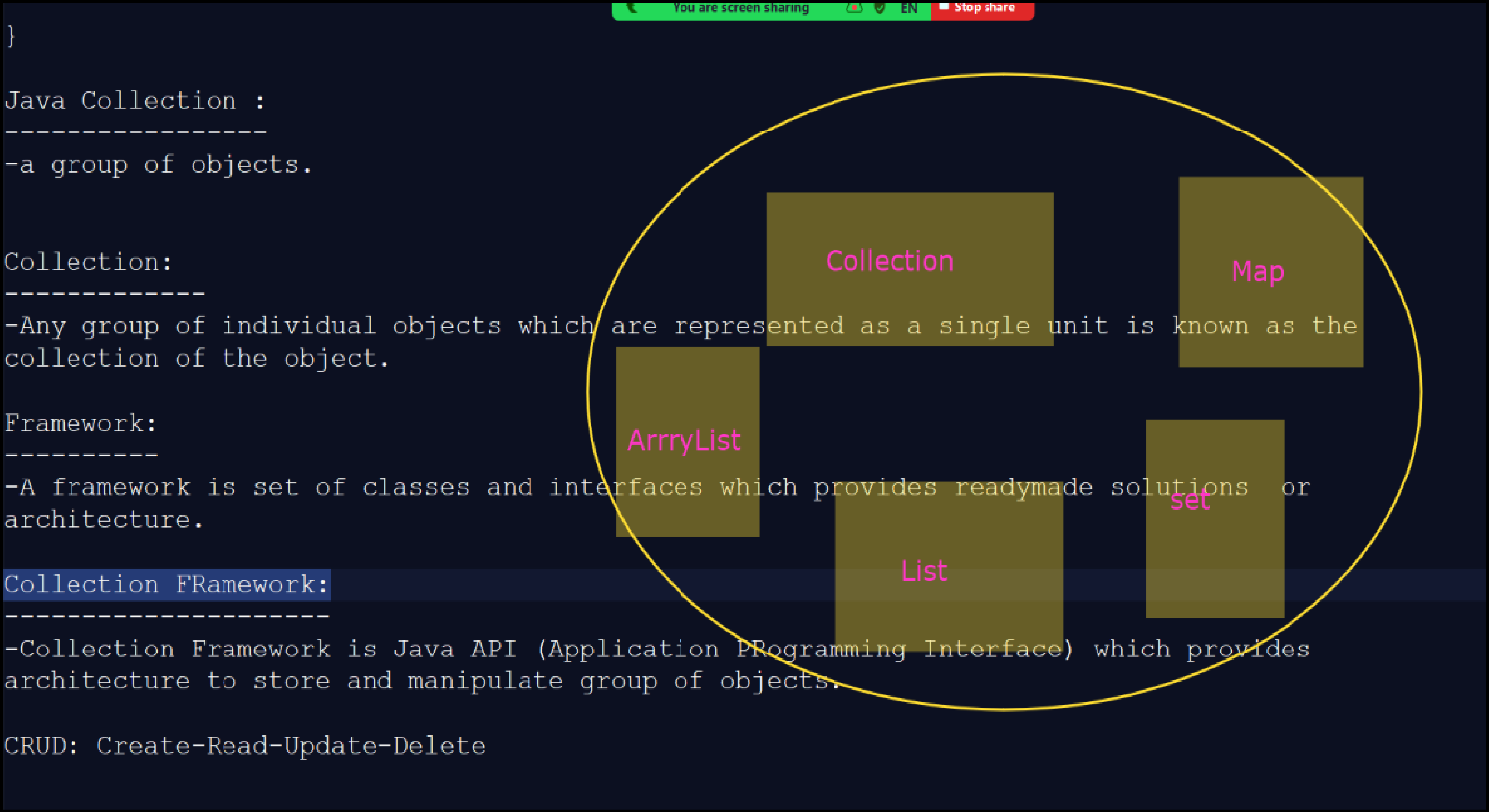
package report; class Report{}

```
erminated> TestDemo1 [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_23.0.0.v20240919-1706\jre\bin\javaw.exe (Mar 18, 2025, 1
elcome java!!!
```

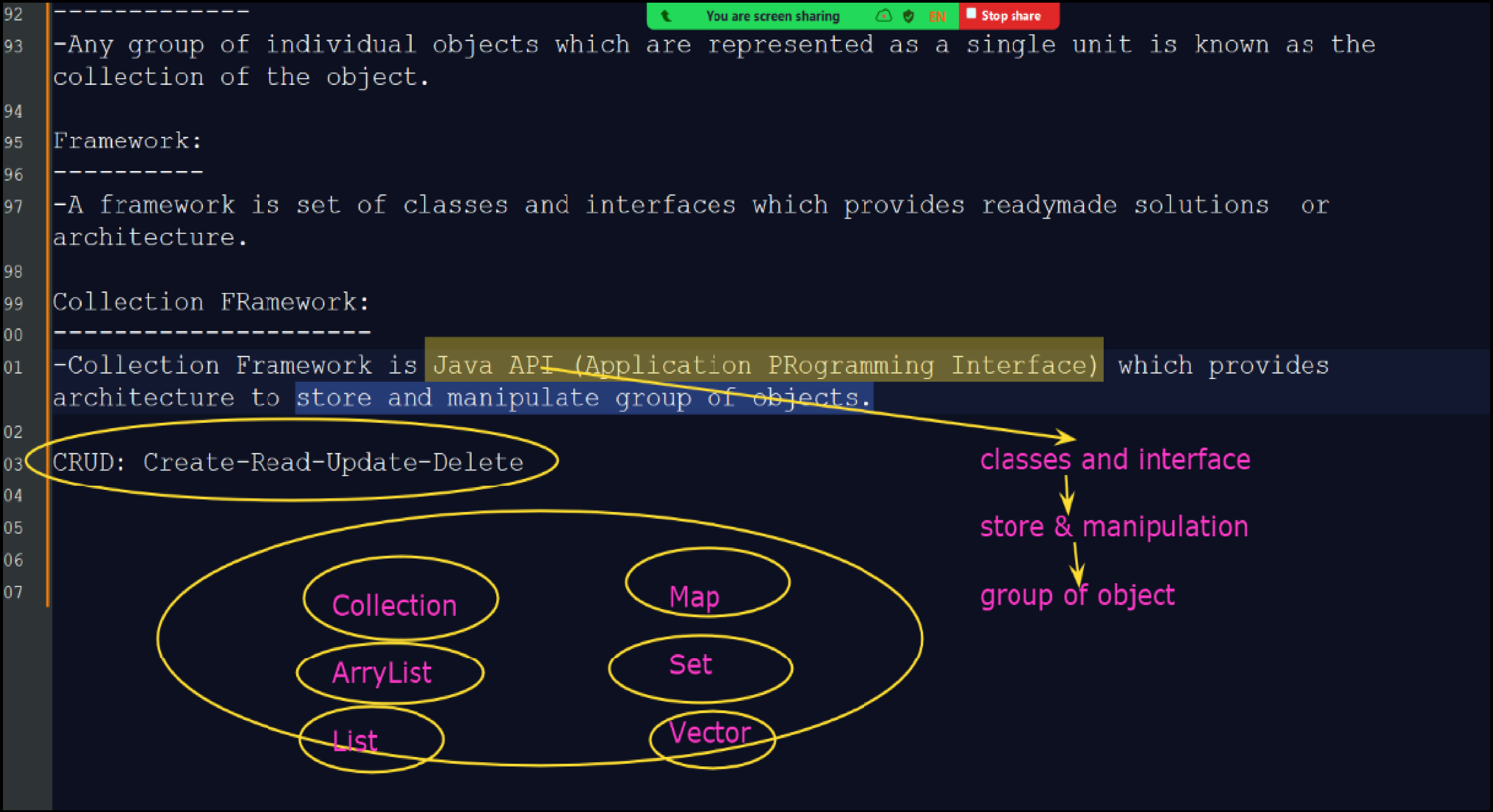


```
    -strictfp
----------------------------------------------------
Collections:


class Employee{
    int id;
    String name;
    e1--->20 parameters

    int arr[] = new int[10000];
    Employee e1;

    Array of Objects:
    e[10000];

}
```

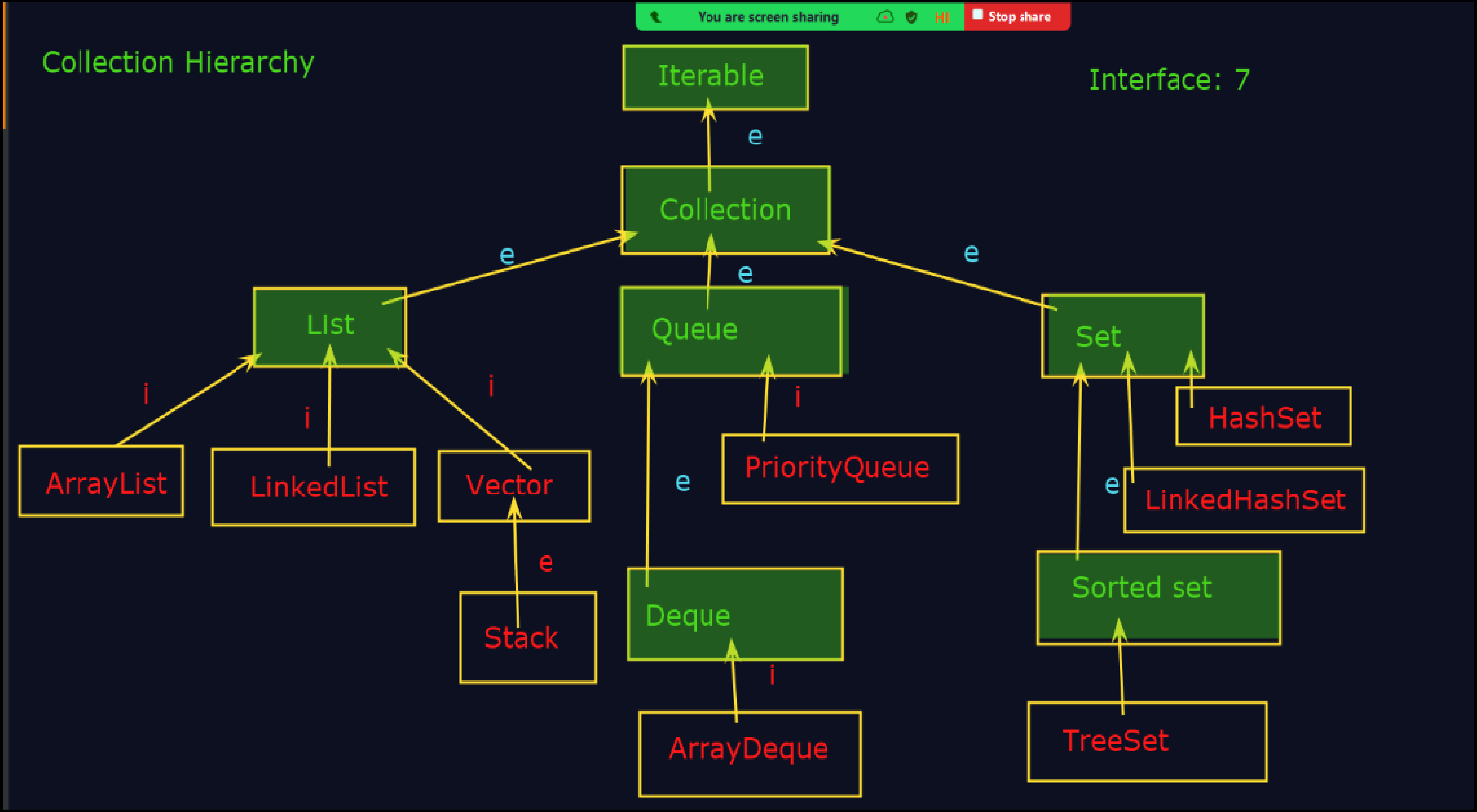
```
length : 38,057   lines : 1,584   Ln : 1,582   Col : 6   Sel : 6 | 1   Windows (CR LF)   UTF-8
```



```
}

Java Collection :
-----------------
-a group of objects.


Collection:
-------------
-Any group of individual objects which are represented as a single unit is known as the
collection of the object.

Framework:
----------
-A framework is set of classes and interfaces which provides readymade solutions  or
architecture.

Collection FRamework:
--------------------
-Collection Framework is Java API (Application PRogramming Interface) which provides
architecture to store and manipulate group of objects.

CRUD: Create-Read-Update-Delete
```

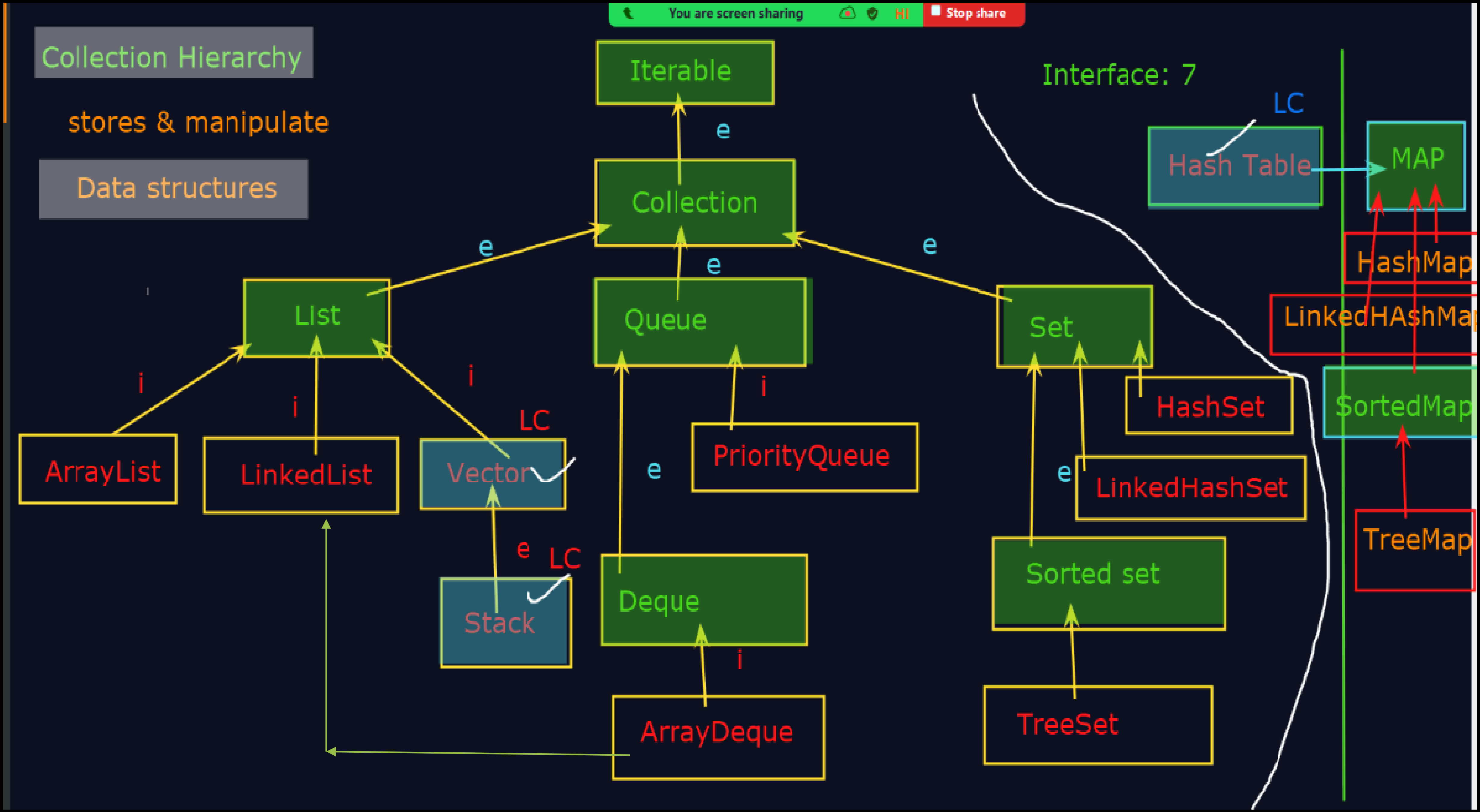


```
-------------
-Any group of individual objects which are represented as a single unit is known as the
collection of the object.

Framework:
----------
-A framework is set of classes and interfaces which provides readymade solutions  or
architecture.

Collection FRamework:
---------------------
-Collection Framework is Java API (Application PRogramming Interface) which provides
architecture to store and manipulate group of objects.

CRUD: Create-Read-Update-Delete
```



Collection Hierarchy

Interface: 7

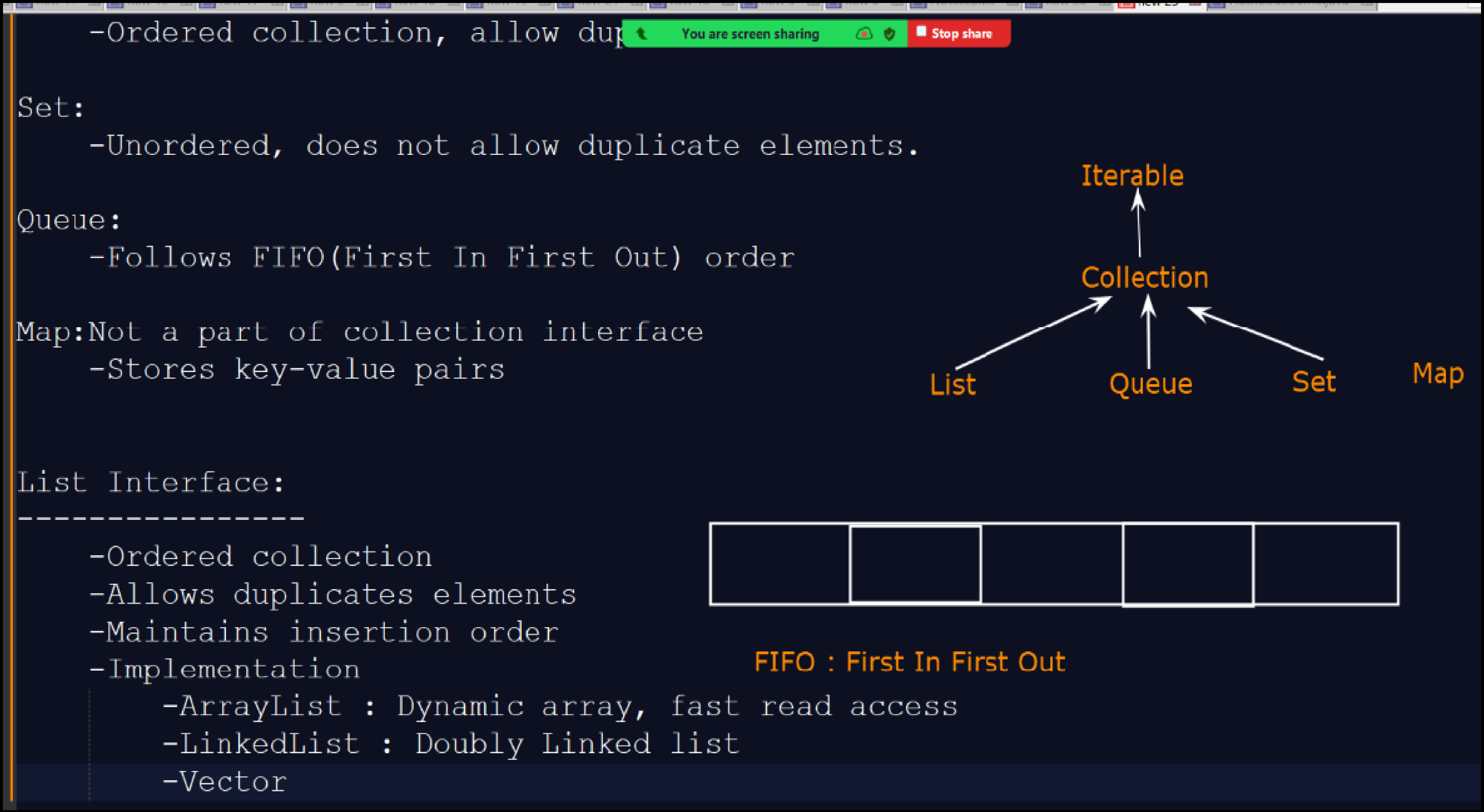


Collection Hierarchy

stores & manipulate

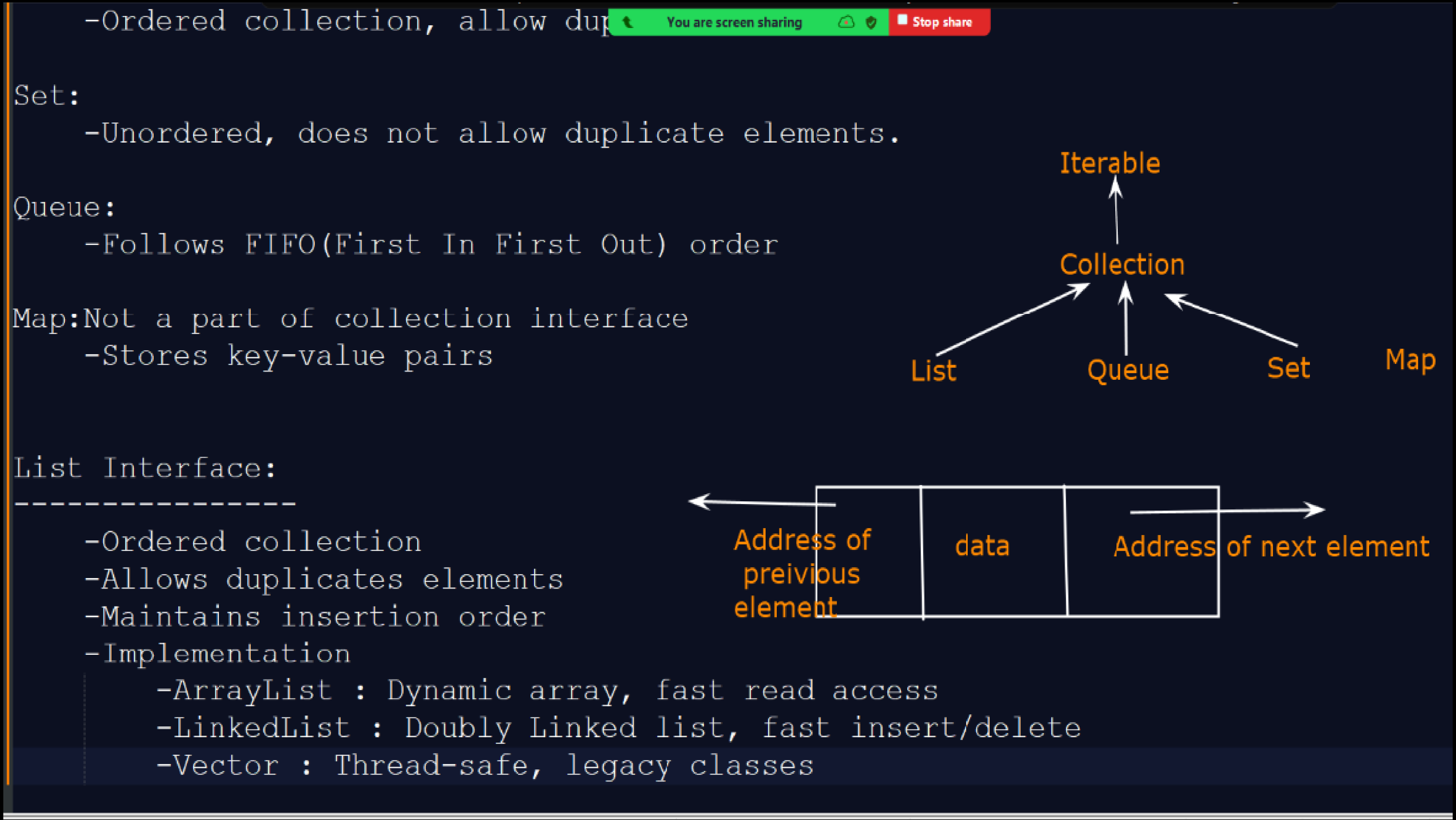
Data structures

Interface: 7



```
    -Ordered collection, allow dup

Set:
    -Unordered, does not allow duplicate elements.

Queue:
    -Follows FIFO(First In First Out) order

Map:Not a part of collection interface
    -Stores key-value pairs


List Interface:
----------------
    -Ordered collection
    -Allows duplicates elements
    -Maintains insertion order
    -Implementation
        -ArrayList : Dynamic array, fast read access
        -LinkedList : Doubly Linked list
        -Vector
```

FIFO : First In First Out



```
    -Ordered collection, allow dup

Set:
    -Unordered, does not allow duplicate elements.

Queue:
    -Follows FIFO(First In First Out) order

Map:Not a part of collection interface
    -Stores key-value pairs


List Interface:
----------------
    -Ordered collection
    -Allows duplicates elements
    -Maintains insertion order
    -Implementation
        -ArrayList : Dynamic array, fast read access
        -LinkedList : Doubly Linked list, fast insert/delete
        -Vector : Thread-safe, legacy classes
```