Collection Framework :

It is a framework or a readymade architecture which is used to manipulate objects or operations.

Operations include sorting,insertion,deletion,searching etc.

It has many interfaces and classes

Hierarchy:

--🡪Interface

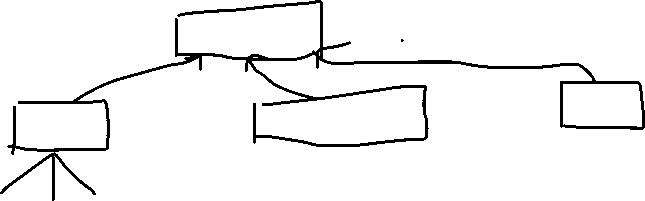


--🡪classses



Iterable

Extends extends extends



1 2 3

Set

List Queue



4

1.Arraylist



2.Linkedlist

3.Vector 🡪Parent class



4.stack 🡪child class

Note:

1.interface to interface relationship is “extends” keyword

2.interface to class relationship is “implements” keyword –

i.e. interface -parent class,class-child class

java collection framework

1.All the collections are available in java.util package in jdk

2.Generics:it is the type of parameter that we are going to add in the arraylist

For.eg. ArrayList<Integer> arl=**new** ArrayList <Integer>();

--🡪<Integer>🡪is the generic here.

If we want to add all the class objects in an arraylist,it can also be added using generics.

Employee e1=**new** Employee("jansi",29);

Employee e2=**new** Employee("raju",34);

Employee e3=**new** Employee("theju",3);

ArrayList<Employee> arl1=**new** ArrayList<Employee>();

//<Employee>🡪generics used to add all employee class objetcs.

arl1.add(e1);

arl1.add(e2);

arl1.add(e3);

//create an iterator to traverse the values

Iterator<Employee> i= arl1.iterator();

**while**(i.hasNext())

{

Employee emp=i.next();

System.***out***.println(emp.age);

System.***out***.println(emp.name);

}

ArrayList ar3=**new** ArrayList ();

ar3.add(100);

ar3.add(200);

ar3.add(300);

ArrayList ar4 =**new** ArrayList ();

ar4.add(100);

ar4.add(200);

ar4.add(600);

ar4.addAll(ar3);//used to add 3 rd array to 4 t array

**for** (**int** j=0;j<ar4.size();j++) //to print arraylist elements

{

System.***out***.println(ar4.get(j));

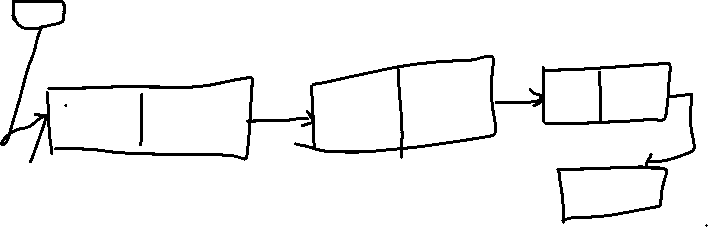
}

Linkedlist:

Linkelist is similsr to arraylist.

Head

Data=30 next



Data=10 next data=20 next

Null

It contains node and each node contains 2 parts:

1.data and next

Next point to data of next node

And next of last node points to null

LinkedList<String> ll=**new** LinkedList<String>();

ll.add("Jansi");

ll.add("Raju");

ll.add("Theju");

//to display the contents

System.***out***.println(ll);

//to get a value from specific index

System.***out***.println(ll.get(2));

//to set a value for specific index

ll.set(2, "Thejaswini");

System.***out***.println(ll);

//add first element and last element in the list

ll.addFirst("Family");

ll.addLast("Atharva");

System.***out***.println(ll);

//to print all the values of the linkedlist:

//1.for loop

//advanced for loop

//iterator

//while loop

//for loop

**for** (**int** i=0;i<ll.size();i++)

{

System.***out***.println(ll.get(i));

}

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*");

//advanced for loop -->is also known as for each loop

**for**(String s:ll)

{

System.***out***.println(s);

}

System.***out***.println("iiiiiiiiiiiiiii");

//Iterator

Iterator<String> i=ll.iterator();

**while**(i.hasNext())

{

String ss=i.next();

System.***out***.println(ss);

}

System.***out***.println("wwwwwww");

**int** x=0;

**while**(x<ll.size())

{

System.***out***.println(ll.get(x));

x++;

}