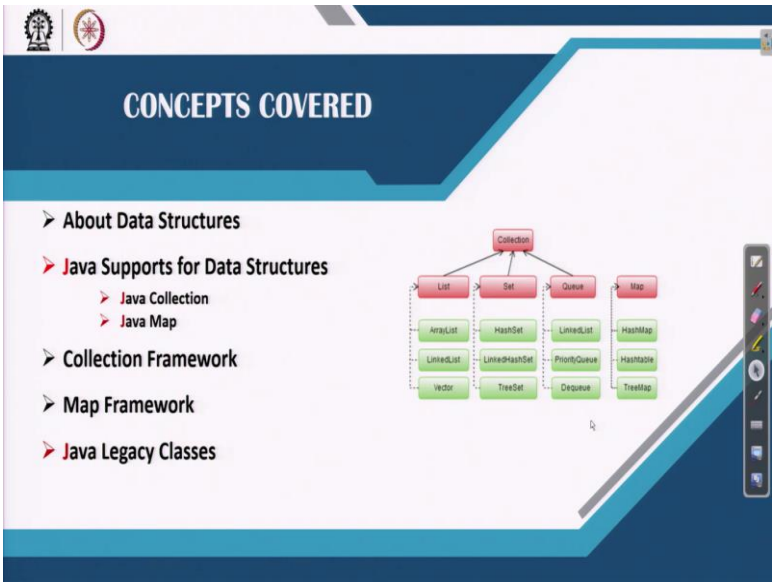
Data Structures and Algorithms using Java
Professor. Debasis Samanta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Lecture 06
Basics of JCF

Let us start with a new topic, topic related to Java Data Structure Programming. The Topic is Java Collection Framework. There are many concept needs to be cleared in this, on this topic. So, today we will focus on basic concepts, basic concept of Java collection framework.

(Refer Slide Time: 00:53)

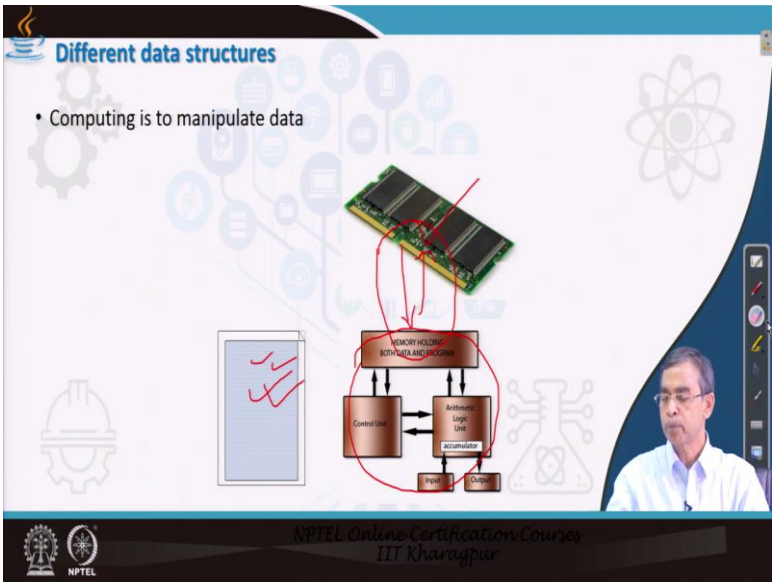


In this lecture, we will try to cover first of all, what is the usefulness of the Java collection framework in the context of data structures. So, a brief we will start this lecture, with a brief about data structures which mostly, all us we know about it and then we will discuss about, the Java supports for handling data structures.

So, there are two framework, java has provided, java developer has provided to us, it is called the java collection framework and the map framework, the two things will be discussed one by one to give an, just an introduction that why this collection framework, what are the things are there, what is there usefulness and everything.

Now, this Java collection framework, a relatively new in Java family. It is basically prior to this Java collection framework, the Java developer followed one concept called the Java legacy classes to handle data structure related task. But later on, Java 2 onwards they radically have introduced a new concept, this is the Java collection framework. It basically allows a programmer to handy with all sort of data structures related activates.

(Refer Slide Time: 02:24)



Now let us first see, what are the different data structures that, Java can supports to us. Now as we know, we have already tried to give an introduction about this concept of data structures in the very first lectures of this course, the introduction in the lecture one. Now, data structures is an important concept. It allows a programmer to define a structure in such a way that any type of data user can handle it efficiently, that efficiency handling means, storing it efficiently once it is stored in memory, retrieving it efficiently.

Now, why this storing and retrieving is an issue? This is because using our conventional computing concept, whatever the program that you write, this is the program if you write, this program basically solve the certain problem. Now, this solving a certain problem means, is a manipulating data.

Basically, computer is nothing but data manipulator engine, data churching engine. So, basically data. So, all data are to be there somewhere, where the data are stored, data all data that needs to be manipulated are stored in computer memory. Now, this program is basically includes a set of instructions, how to manipulate data.

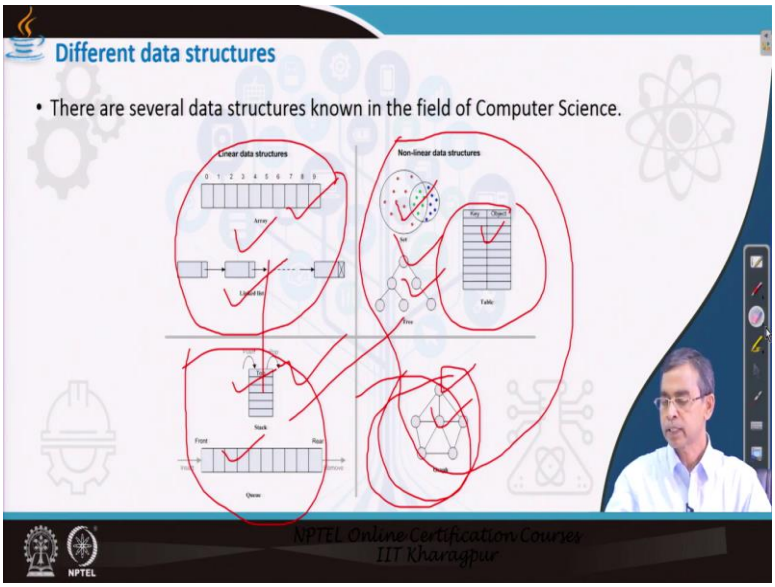
Now, so far the data manipulation is concerned, we have to create the data, that somehow you created from the different source we can obtain the data, those data ones it is created to be stored there and then while you are doing manipulation is a program you have to retrieve the data from the memory, do whatever the operation, addition, subtraction, comparison whatever it is there, do it and once the result is obtained again result can, needs to be push back to the memory again.

So, here data needs to be retrieved, data needs to be stored and this is a constant process that you will go between the computers, that is basically processing unit. Now, so this basically the idea about computing and here is the need of a very efficient mechanism by which your data that you want to process should be stored in a very efficient manner, so structured manner.

That is why difference structure needs to be consider depending on that, how efficiently we can manage. Now, here is the basically problem, that how we can define our own data. So, that we can manage it efficiently. So, defining your own data is basically declaration of your custom data. Now whatever be the custom data or the primitive data or whatever it is there, we basically see all data not a single one atomic component like, it is basically a sea of data. All this things also more, convenient way we can say is a collection of data or a set of data.

So, data is basically not a single component it is basically is a collection. So, in Java a data is viewed as a collection that is why the framework that is required in order to manage this collection and that framework is called the collection framework.

(Refer Slide Time: 5:53)



Now, here are the different type of data that can be considered to store a collection. So, here the very simple one is the arrangement it is called the array is a collection. Another collection is basically list, so that means all the data will be stored and the link of one data with other data can be maintained somewhere.

Obviously, there is some advantage, of this, over this or advantage of this over this and so on. Those things are the different issue will be discussed, when we will discuss all these data structure separately and manipulating those collections storing this kind of data structure using the Java collection framework support.

The other data structure that we usually deal with, is basically stack and then queue. Now, another also type of data it is called the set, there is a table, there is a tree and then graph. So, these are the different way you can see, these are the basically collections. So, graph is also collection of some data, they are stored in a little bit peculiar way.

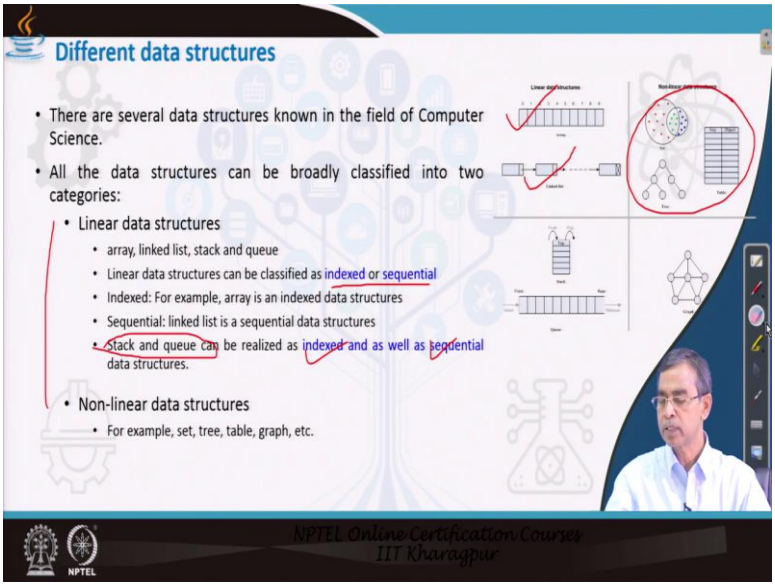
Table is also a collection, it is also stored in a different way, then all these things are there. Now, so these are the different what is called the type of data or we can say collection type of different collections and here you can see all collections can be broadly categorized into four different parts. This is basically one category, they belongs to call the linear collection. So, linear data structures and here another is called non-linear collection.

So, this basically belongs to non-linear collection and this is one thing, it can be realized using both linear as well as non-linear. So, this is the particular, they are stack in queue line. So, overall all the type of the data can be broadly classified into this four categories, what is

called this is also non-linear, but in Java this graph stack collection is not given any importance because if we understand all this three types then you will be able to build your own collection of this type.

So, there is no support from the Java developer, to maintain the graph collection. So, but otherwise all these are the three collections are manageable in the Java support, with Java supports.

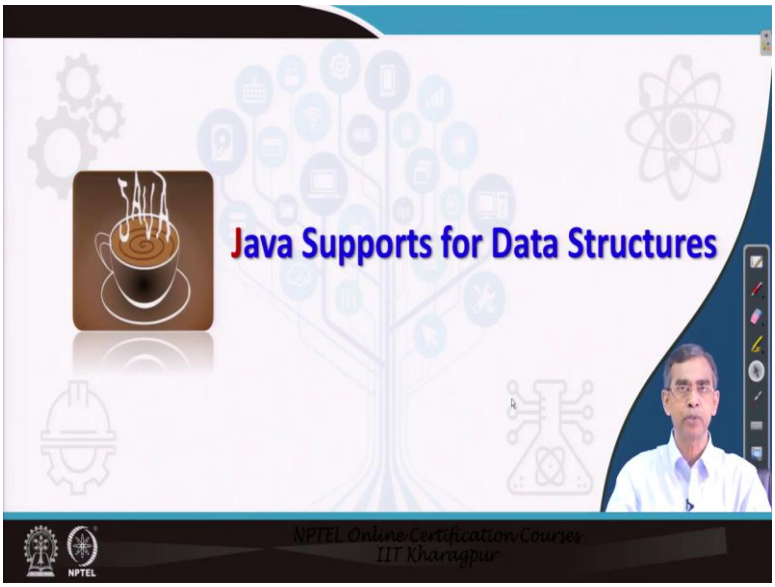
(Refer Slide Time: 08:29)



Now, so this is basically the idea about different data structures and there concepts that is, so far the manipulation is concerned. As I have already told you, that all the data structure can be broadly classified linear and non-linear. Now linear again are called indexed or sequential. So, this is an example of indexed and this is an example of what is called the sequential and as I told you, stack and queue can be considered as a both indexed and sequential.

Now, these are the data structure, this basically is a non-linear, they can be viewed either in a, it is basically is a linked manner actually. So, it is called the not the sequential, it is a linked manner that we will discuss in detail, so in we will discuss about this according the, how they can be maintained and they can be stored it is there. Now, so these are the different type of data structures, linear or non-linear, linear can be indexed or sequential and non-linear is non-linear of its own merits.

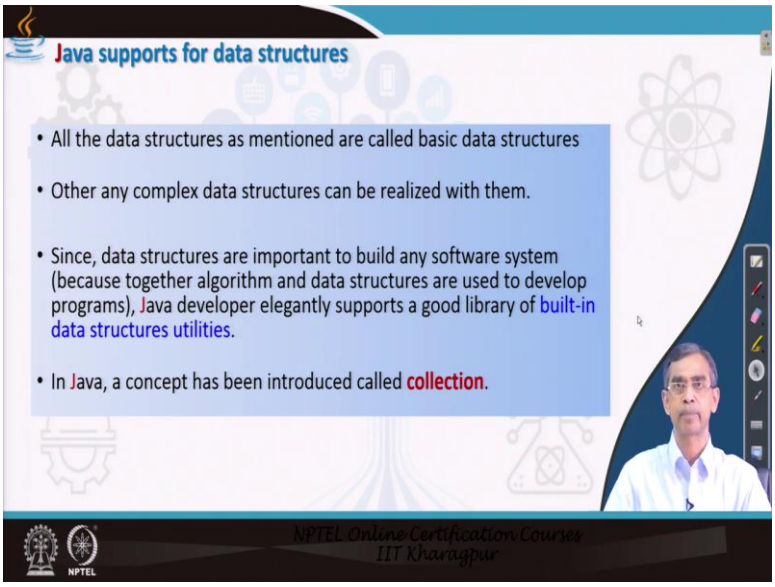
(Refer Slide Time: 09:29)



Now let us see the Java supports handling all type of data structures those are there. So, in this introductory lectures for the Java collection framework, we will quickly just mention what are the different supports are there and in our next few lectures three lectures, we will try to cover each framework in details.

Then, for each data structures, like say array, linked list, tree, graph and everything will be discussed individually in separate discussion where we will discuss the concept of those data structures and then the Java support in details, explicitly related to that particular structure. So, this is basically plan of this course.

(Refer Slide Time: 10:17)

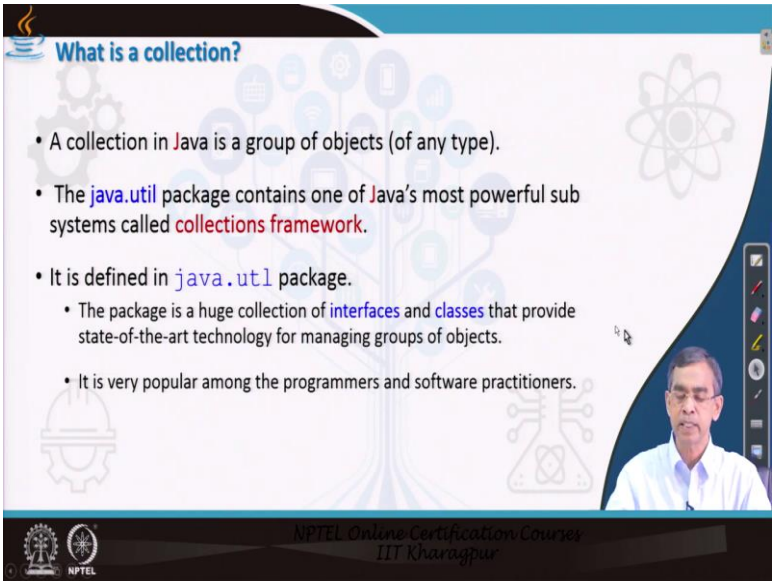


Now let us first see, what are the different supports from the Java developer, so far data structure manipulation is concerned. Now, I told you, Java provides a framework, a framework is basically nothing but a built in, built in means is a library, readily it made by the Java developer itself and this is available as a part of utility.

All this utilities, that means how to process different data structures, that mean how to store? How to retrieve? How to perform certain operations on all those data structures, all methods regarding all this things are readily available. So, readymade programs are there, available as a library, this library is stored in a very vast package.

The name of the package is called Java. util, U T I L. The util is a very extensively used one package after Java . lang package. So, overall all this Java . util the facilities those are there, they are called collection. So, all these utility programs related to the data structures are called collection.

(Refer Slide Time: 11:31)



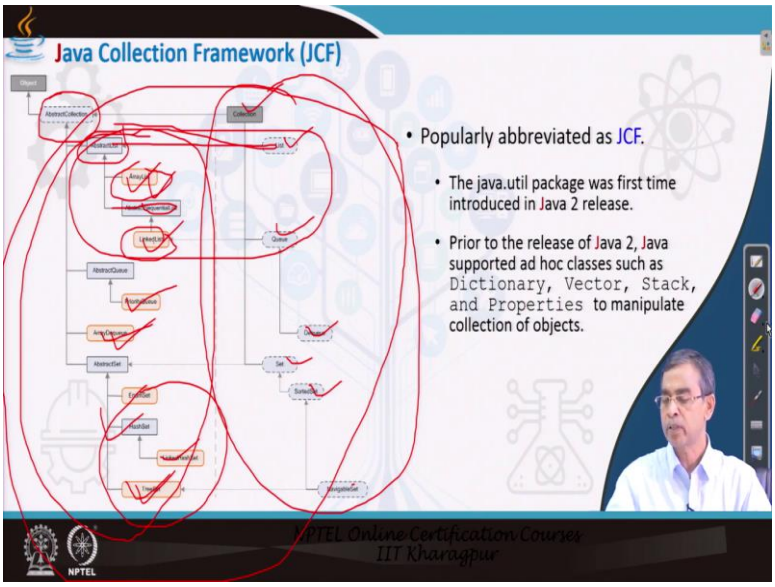
And overall this collection are stored in a very systematic manner for which Java developer proposed or introduced a very nice concept that is called the collection framework concept. So, this is also more popularly called JCF, Java Collections Framework concept and as I told you it is defined in Java . util package as a part of definition part and parcel of definition this Java . util package is basically is a collection of a large set of interfaces and classes.

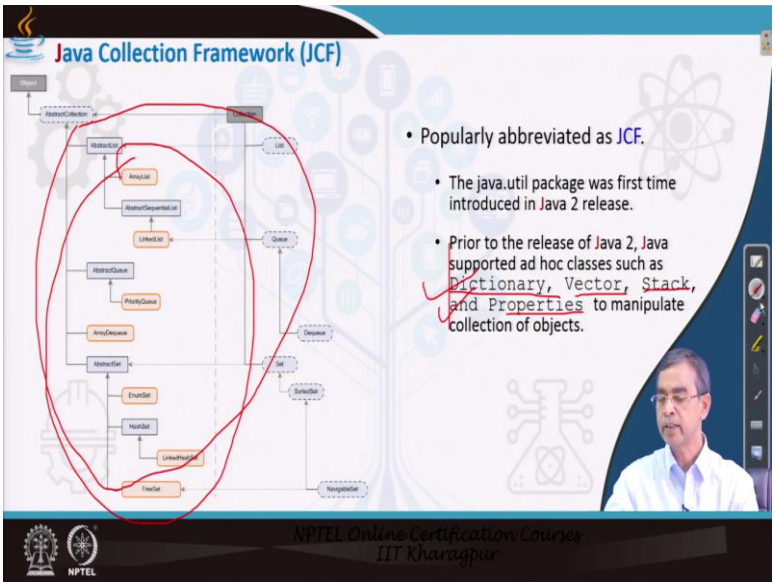
All classes are basically is the utility in a real life form. If you can create an object of this class that means you will be able to access methods of all those classes there. Interfaces are not, because interface of, given a interface you will not be able to create any object of that

because interface is not suitable for creating any object because it is basically is a black box sort of thing, it is a template.

So, that interface is there but interface gives an idea about what are the different methods are declare. The definition is basically, completely done in the respective classes, which extends interface. Now in our few next slide, we will see exactly the composition of this framework.

(Refer Slide Time: 12:52)





Now this picture if you can show little bit clearly, it basically gives a purview of all the facilities that means all classes and interface. Those are defined there in Java . util package as a Java collections framework. So, whole the things basically, this whole the things is called the collection framework. Now, in this collection framework, as we see there is a class call collection. This class is an abstract class.

As you know abstract class has only some fields and method it is called the abstract methods, but they do not have any real methods, so that code is not there. Now, those are the method within the dotted line, they are basically interface. So, this collection under this collection. Basically, there are certain interfaces are declare which basically extend the collection.

So, this is one interface extend collection queue is an interface dequeue, set, sorted set, navigable set all these things are there and there is an hierarchy, hierarchy that least basically extend collection, queue extend collection, dequeue again extend queue which basically extend collection like this one, so these are the hierarchy.

Now as I told these are all interfaces, interfaces is basically is a design concept that these are the things are to be defined somewhere, if they want to have that kind of concept. So, it is basically things are there. Now all these interfaces are to be implemented.

Though, so they are, so there is a need of each class to implement a number of interfaces like. Now here is the interface, these are the interfaces and see whatever the classes are there, which basically implements all the interference. So, these are the set of interfaces, these are the set of classes which implements all the interfaces which is defined here under the collection class. Now if we see all the classes, they again have certain class hierarchy.

That means some class inherited from the other class and whatever it is there. Now, there is an abstract collection class is basically superclass of all the collection classes, those are there. Again, it basically derive is called a Abstractlist, Abstractlist basically provides Arraylist and Linkedlist or in other words, what you can say, Linkedlist is a class which basically has a super class is an abstract class, which basically has derived from Abstractlist. Arraylist is another class which basically define Abstractlist. Now Abstractlist, is a class which basically implements list class.

So, this is a very complex hierarchy, which basically used to realize array structure and Linkedlist structure. Likewise, there is a again number of classes, which realize the queue structure mainly the priority queue and then array dequeue and there is also another type of this things called the set, Linkedheadset and Treeset, they are basically to implement the set.

Now by these things you can understand about the tree structure, the queue structure, then stack and queue is a part of the same thing and then set structure array structure, all these are the data structure, which basically we have learned about has been current very carefully and meticulously maintained in the Java . util package.

So, overall what I wanted to say is that Java collection framework is a very vast, is a collection of many classes and many interfaces. Interfaces, we have to learn only, but not need to use them, but we have to learn more systematically about all the classes related to particular data structure that it can supports.

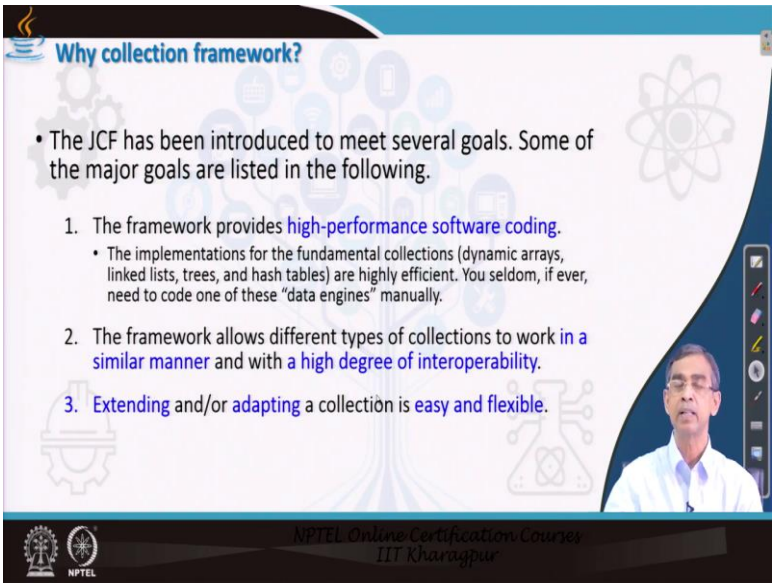
Now, these are the collection framework. I already told you prior to this Java collections framework in Java 2 release, there was one collections, it is known and they are called Legacy classes at the moment because they are not useful nowadays or they made deprecated or obsolete because the new things are already covered everything.

But all those things, although deprecated but still it is in use. That means you can use if you want. Actually what is the idea is that the earlier collection supports. Those are there, like say they are called Dictionary class, Vector class, Stack class and Properties whatever it is. So, these are basically the prior implementation about data structure facilitates.

Now later on Java 2 release, they supported these are the latest one parts. But in order to implement all these collection, that means to define different methods are there. They basically follow all these depletion. So, that is why they are not so obsolete, they are basically still it is in use because they are on the back of all these things are there.

So, they are very old, but not obsolete we should say, they are still very useful to have it and as a programmer, you can even access them in your program also we will see exactly how, all those Legacy classes also can be utilized while you are writing your program handling the data structures. So, this is about JCF the full form of it is basically Java collection framework.

(Refer Slide Time: 18:31)



Now, let us proceed little bit further, why these Java collections framework that the Java developer supports to all programmers. Now Java collections framework, the main motivation of this Java collections framework is to provide a very good sophisticated user friendly programming environment. That is the most what called the motivated, motivation behind this.

In fact, this Java framework, Java collections framework provides high performance software coding if you write your own program to manipulate data structure, I am not sure whether they are fast, so far execution is concerned, that means storing and retrieving manipulating different things whether fast or not.

Actually, whenever you have to deal with programs and reading the data and whatever it is, there it should be made as fast as possible. So, first execution is an important criteria for any software engineer and to facilitate this, Java developer supports this high-performance software coding by virtue of Java collections framework.

Another advantage, another what is called that goal of the Java framework is that, there are, these Java framework because there are different type of data structures, we have already mentioned, now they needs a different way to deal with and not only this, there may be requirement that, there will be an operability from one type of collection to another type of collection.

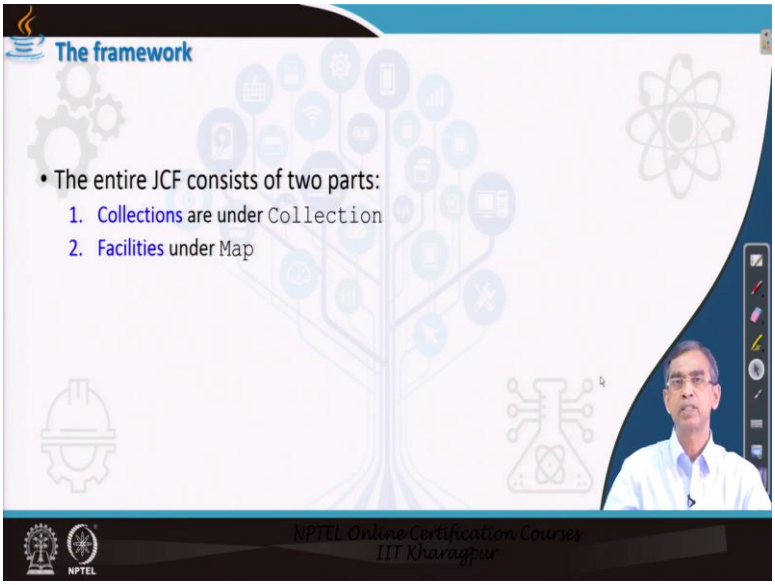
For example, you have stored some collection in the form of array you want to convert the same collection, but stored in the sequential form like Linkedlist or say array, we want to

store into a queue or vice versa. Now if you want to do it, that means collect, changing from one type of data structure to another type of data structure, storing the information in the form of an array converting into the in the form of a tea, for some reason then that you can do.

So, that Java collections framework highly essential for this interoperability issue. That means you can switch from one collection to another collection with a very near of a simple one code, that is not so much difficult for this and extending and/or adapting a collection is very easy. For example, if you want to adapt a particular structure in your program, if you are familiar to the Java collections framework, it is very easy for you.

So, maintaining a tree structure for your data or maintaining a graph structure for your data or a link or a stack or queue absolutely very easy because there is no need to discuss about the details inside how they are managed, it is just only to learn about what to do, not how to do. That means, you know that what is your need and according to you can search, what are the methods are there fulfilling you need and that is enough so far the program is concerned. Programming related (())(21:44) is concerned program related data structure using Java collections framework is concerned. So, this is basically the benefits of the Java framework.

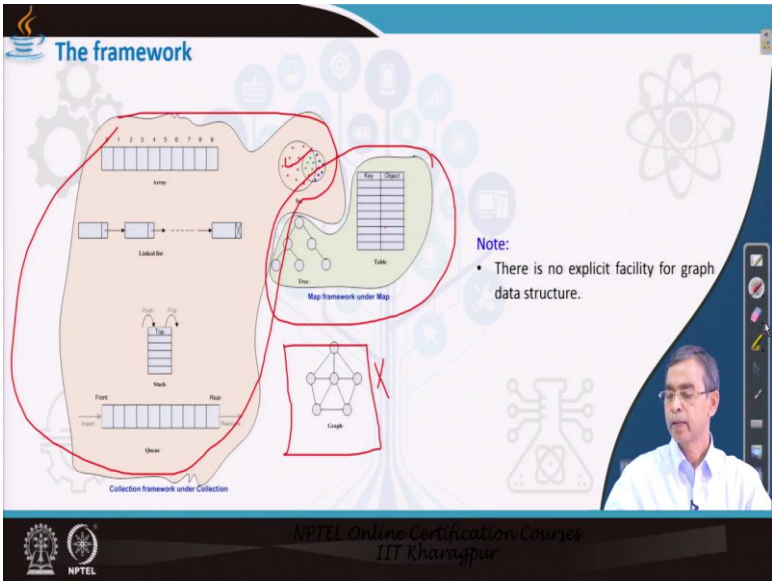
(Refer Slide Time: 21:51)



Those are basically we are going to learn throughout this course we will deal with only these different features the different benefits advantage whatever the pros and cons or limitation or careful precaution everything we are going to learn about it. Now, let us start about the two things those are there, as it is very vast. So, Java developer planned it to divide into little bit two different types, one is called the simple collection framework, another is map framework.

So generally, the collection framework itself is called JCF, but there is one part about it is a map framework.

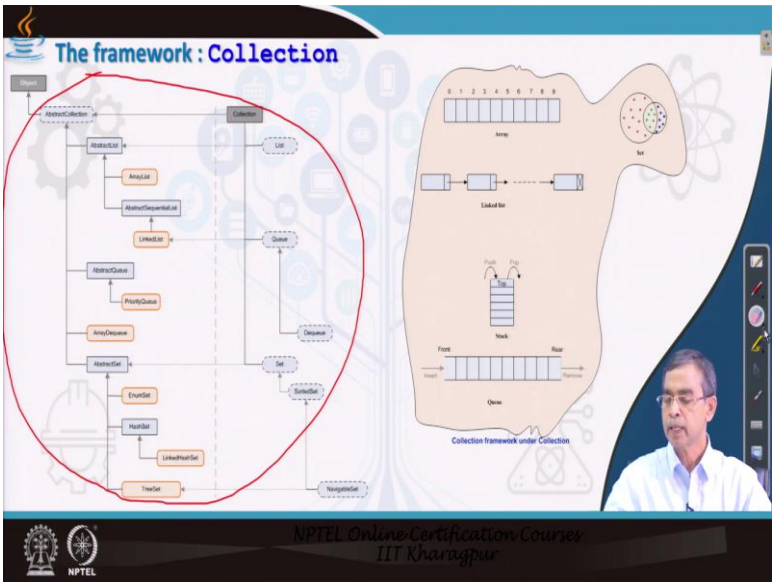
(Refer Slide Time: 22:31)



Now the Java collections framework and map framework is basically cater to the need of different type of structures. So, this part is basically, is basically deal with collection framework that means array, Linkedlist, stack, queue and set these are the part of collection framework or collection and these are rest of the part as I said, that is the tree and table they are belongs to the map framework and graph is basically is the one part, where the Java does not have any support, so it is not included here.

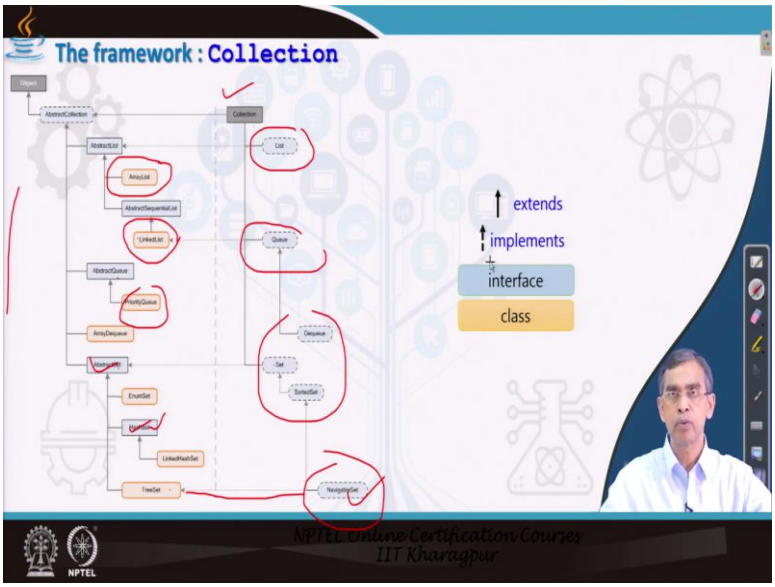
So, these are the basically the concept of framework, those are defined they are in Java . util namely the different data structure support it is there. Now, we will just quickly discuss about the different facilities quickly we will discuss in detail one by one.

(Refer Slide Time: 23:23)



Now let us first start with the collection framework. Collection framework as I said this is a complex one, it basically consider this thing. So, including all this things say it Linkedlist, stack and queue, the difference class and interface, which are defined there, they are basically part of the collection framework.

(Refer Slide Time: 23:48)

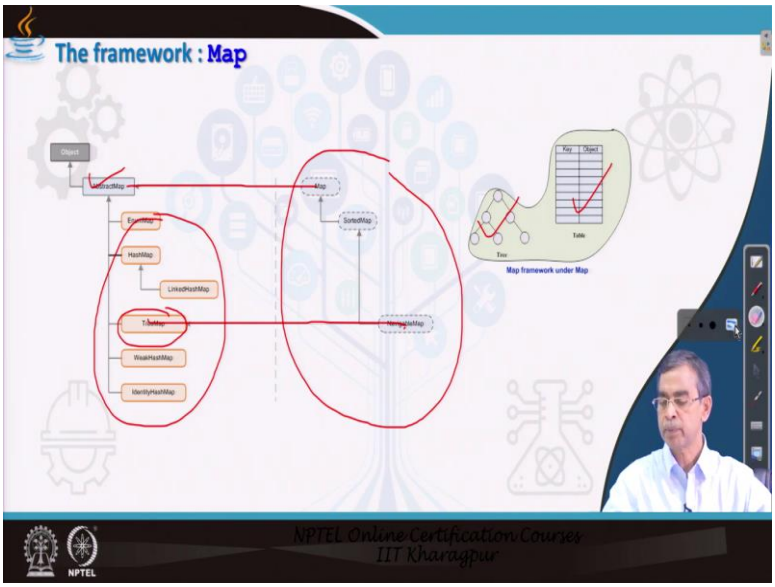


Now, let us come to the different type it is there, as a sample of notation, the (())(23:52) that you can follow for your easy understanding, I want to say those are the dotted link is there, these are the dotted link like. So, these are the dotted link like, so these are dotted link implement that this class implement this interface it is like this.

So, this class Abstractset implement set and like this. So, it is Linkedlist, implement queue and like this one and those are the, this color, here this color is basically the actual class that is basically beneficial in your programing. So, these are the classes and those are the, these are the right seted right with dotted border, they are basically interface.

So, these are the interface, interfaces are there and those are the basically within a difference set, they are basically abstract class. So, these are the abstract class, it is there and collection is a super abstraction, it is there because all these are basically under this collection. So, this little bit notation we will follow overall discussion. So, we have to little bit memorize this one. Anyway, so constantly if you use it and then you can understand about it.

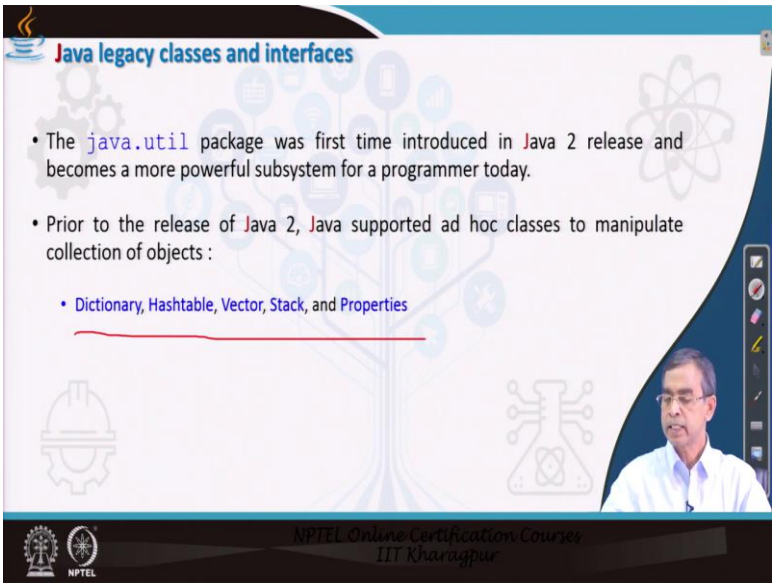
(Refer Slide Time: 25:09)



Now so this is the collection and then likewise the map framework it has basically towards the implementation of tree and table and it has the number of interface the, it is like maps stored map and navigable map and they basically have been implemented by different classes for example tree map is a one concept implemented by this class implements navigable maps. Then map is basically implemented by abstract map and these are classes dealing with all the facilities related to tree and table.

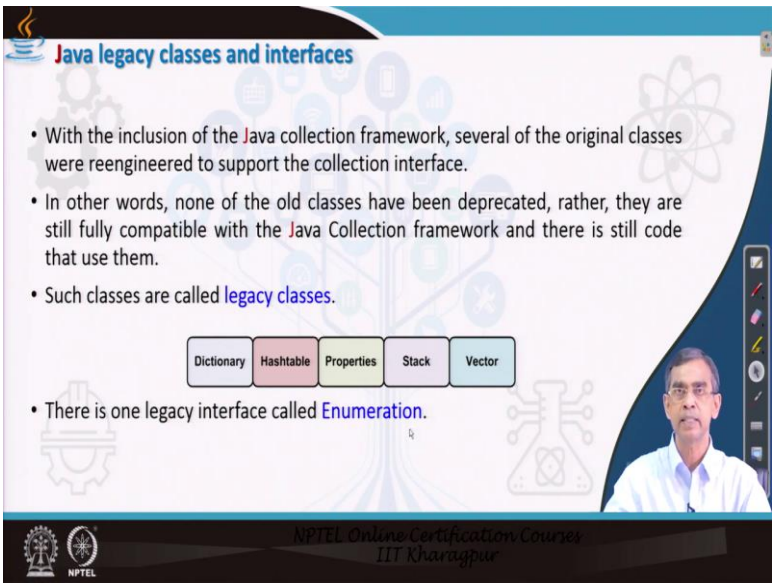
So, this is basically the concept that is there in the map framework. Now, so these are the concept and again notation we follow the simple notation, it is there.

(Refer Slide Time: 25:55)



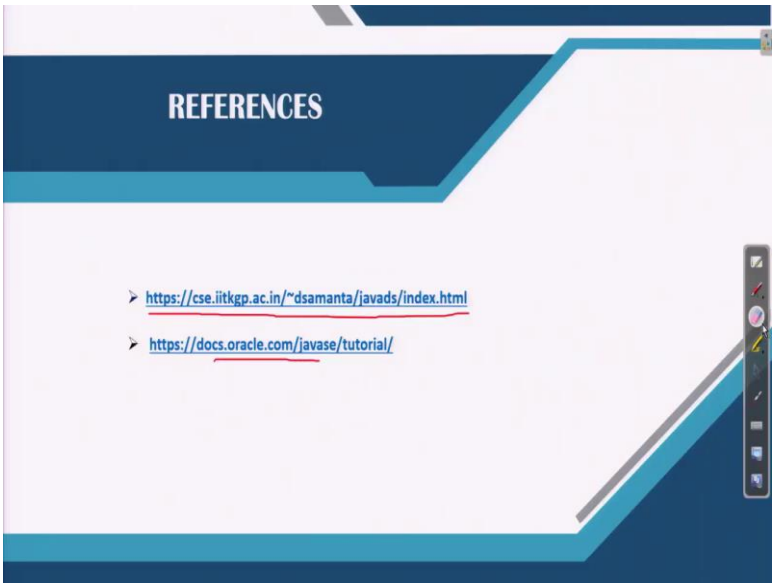
Now let us come to the Java legacy class as I told you, Java legacy class still it is in use. But Java 2 release onwards they have been deprecated but they are again defined in Java . util package and these are the few classes, those are still it is in use are there. So, this is the overall concept of the different what is called the classes are there in, as a part of the Java collections framework towards the managing, the data structures and the details about these concepts and other information.

(Refer Slide Time: 26:36)



So, this is basically the classes and classes, those are there in Legacy class and in this Legacy class only one interface enumeration, enumeration concept will be discussed while we will discuss in details about all those class separately.

(Refer Slide Time: 26:55)



Anyway, so this is the different concept that we have discussed in and the basically the overall story of different, different classes facilitating the different data structures those are there in your, I mean, your in your programming as a programming environment. Now again regarding the details about the different, different classes and interfaces. There is a again material that we have already provided for you.

You can check this link where you can find a detailed discussion about the different classes related to this Java Collections framework, along with all programs and examples and everything and we will follow in our next few video lectures classes, the different classes related to a particular data structure one by one and then we will discuss each data structure individually in details, exhaustively they are application and utility.

So, this is the part of, this is the topics that is due and we will discuss and more details, if you want to have the developers document. Obviously, Oracle doc is the nice point to conserve. So, you can check this link also the second link, so that you can understand the more detail reference to this material. Thank you. Thank you very much.