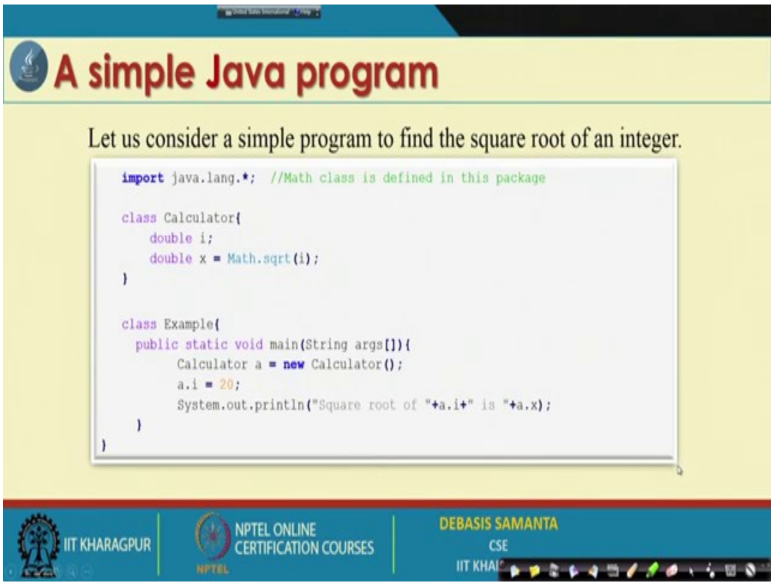
Programming In Java
Prof. Debasis Samanta
Department of Computer Science Engineering
Indian Institute of Technology, Kharagpur

Lecture – 09
Java Programming Insights

So, let us learn in this lecture few detailed things in java. So, there are many things which we have assume that whenever you have to write java program, we have to follow it. But now we are in a position to learn about why we should follow so many things in java programming. So, it basically makes a little bit easy to understand the different concept in java. So, first is the main method. So, main method looks very cryptic compared to the main method as we have learnt in the C programming. It has many things in it.

(Refer Slide Time: 00:53)



A simple Java program

Let us consider a simple program to find the square root of an integer.

```
import java.lang.*;  //Math class is defined in this package

class Calculator{
    double i;
    double x = Math.sqrt(i);
}

class Example{
  public static void main(String args[]){
        Calculator a = new Calculator();
        a.i = 20;
        System.out.println("Square root of "+a.i+" is "+a.x);
  }
}
```

Now, we will discuss about the concept of main method. Now before going to have this discussion, here just I want to start with one another example and the name of the example is here if you see, we create one class call the Calculator. This basically calculates for any value i in it and the results will be returned and store into the value x. And here if you see we use another method, but this method we have not defined anywhere. So, the name of the method is basically square root and here Math dot square root, it indicates that this method belongs to some class; the name of the class in this case is Math.

Now, this class is where actually now if you see this is one import section that we have. Now import is very important whenever you want to use some a special class defined somewhere, then you should use import. So, here actually java dot lang package has been imported. So, it is basically there is a package where many mathematical calculations related to the classes and then methods are defined so, this one. So, this basically the method has been imported here.

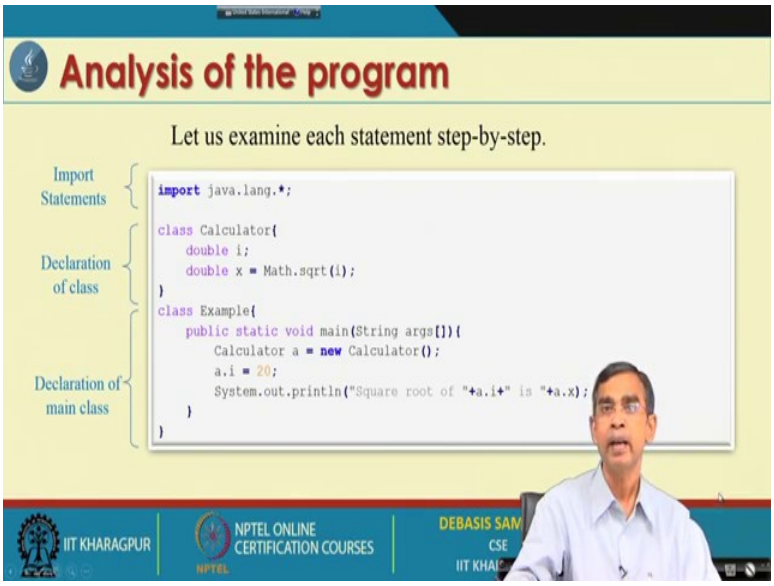
So, that we can we can access the square root. So, we do not have to define the square root method of our own as it is already there in the lang package. So, we can use it. So, this is basically one method that is basically already library method we can say. Now this is a class declaration and the name of the class is Calculator, it is not a main class. Now here Example is the main class in our case because it contains the main method here.

So, if you see the main method it has many things these are the three things are prior to that public static void. Now we should know exactly what it is the implications of all this naming that public, what is the public? Why is static? ,the void and everything and further also you see in this there is an argument; argument is like this. So, the first thing is called the type; that means, the name of the argument is this one which is an array actually.

So, it is an array of type the type is called the String. So, it is gone the array of strings. Now Strings is basically a class. This class is defined in again java. lang. So, these are the 3 few things that is very important public static void and then String args and everything and rest of the things are very simple. Here we can see the Calculator class of the class we have already declared and for this class we create an object and the name of the object is a. And for this object we initialize its member element the value i as 20 and then finally, we use this objects and then for this object we print a.x where x basically store the square root value of i.

So, it is square root of a.i that mean in this case a.i is 20 is 20. So, if you run this program. So, this will basically print the square root of 20.

(Refer Slide Time: 04:24)



Analysis of the program

Let us examine each statement step-by-step.

Import Statements

Declaration of class

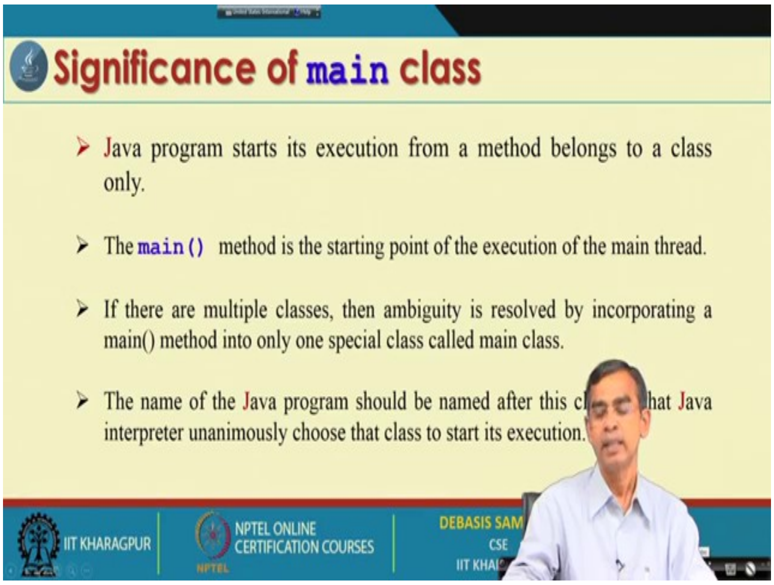
Declaration of main class

```
import java.lang.*;

class Calculator{
    double i;
    double x = Math.sqrt(i);
}
class Example{
    public static void main(String args[]){
        Calculator a = new Calculator();
        a.i = 20;
        System.out.println("Square root of "+a.i+" is "+a.x);
    }
}
```

Now, here few things as I have already told you import is required because you want to use some class and some methods in them. So, we have to use the import lang here and this is the main class which includes the main method.

(Refer Slide Time: 04:41)



Significance of main class

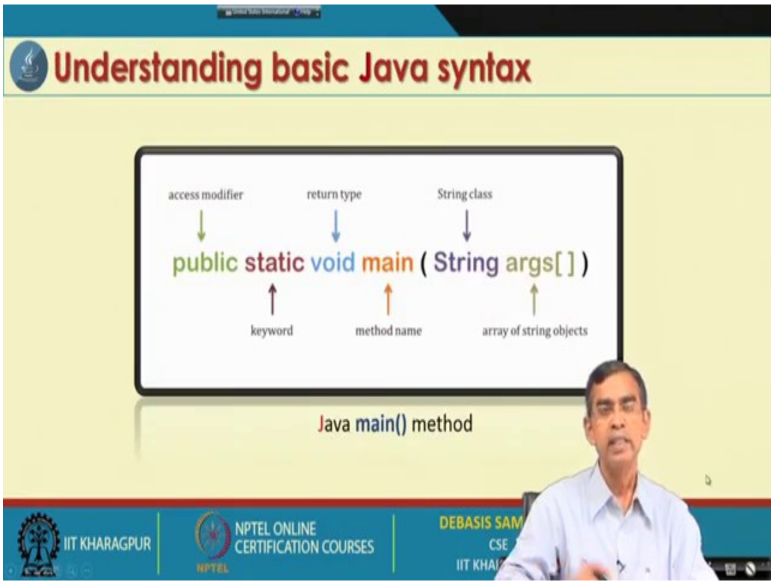
- Java program starts its execution from a method belongs to a class only.
- The main() method is the starting point of the execution of the main thread.
- If there are multiple classes, then ambiguity is resolved by incorporating a main() method into only one special class called main class.
- The name of the Java program should be named after this c hat Java interpreter unanimously choose that class to start its execution.

Now so, main method is very important concept in java. So, there should be one class and the one class having main method and that is the main class and this is because as you see that in a java program there maybe many classes.

So, out of this many classes so, they are there should be some indication that the execution should start from which class. So, if a program file contains a main method, then it gives a signal to the java runtime environment that we have to start its execution from the main method. So, out of the many methods right so, the main methods if it is there in that class so execution will start from invoking this main method and then main method will call some other method, create some many objects and then do whatever the operation it is there.

So, main is basically the starting point of your execution and the java runtime can understand that it is there. If you do not have any main method in any class, then java runtime environment will not be able to understand that which and then it will give an error.

(Refer Slide Time: 05:55)



Understanding basic Java syntax

So, this is why the main method is there and there should be a main method.

Now, I will discuss about this main method having many what is called the elements in it like public static void mean main and all where the String and everything. So, the first the item that you can say the first what is basically public. It is called the access modifier. In fact, in java there are many access modifier, 3 different type of access modifier a one is public, another is private, another is protected.

So, in this case it is public and it is always should be public. If you make it private or some other type, then it will create an error compilation will not be successful. So, this is the one called the access specifier and the it should be public always. So, this is always required indicate that public; that means, the main method should be publicly useable. So, anyone can run a program if you want to restrict that this program should not be executed by anyone, then you can withdraw this public you can write private, but it is no use because you have return one program which no one can run it then it is not useful things right.

So, you should always make the main method as a public and this is a way that public we can declare. And then the void so, void means you know every method should return sometime. This is the concept or method or function or operation any operation can have 0 or more input and always return something like.

Now, in some situation it does not return anything, then in that case we should mention that void. So, in this case as the main method does not return anything because main method is not called by anyone other than the java runtime environment itself. So, the main method is not responsible to return to anything from its caller so, that is why it is void.

So, always this keyword should be specified as the void. So, void is the return type which should be declared as a void null value actually, it should not return anything in that sense. And next is static, this is very important. So, some methods you know what about/above the methods we have discussed earlier say areas, circumference. In order to call this method, we first create an object and for that object, we call the method for an c.area or c. circumference like this. That means if you do not create any object, then you will not be able to access any method in it. Whereas, the main class if you see we usually do not create any object of this class and then without creating any object of this class, we want to access its method in which is define in it.

So, that is why if you declare a static, then no object creation is not required necessary without creating any object we will be able to call this method. So, if a method is declared static. So, no object is required to be created to access this method for that object. It will automatically be accessed like.

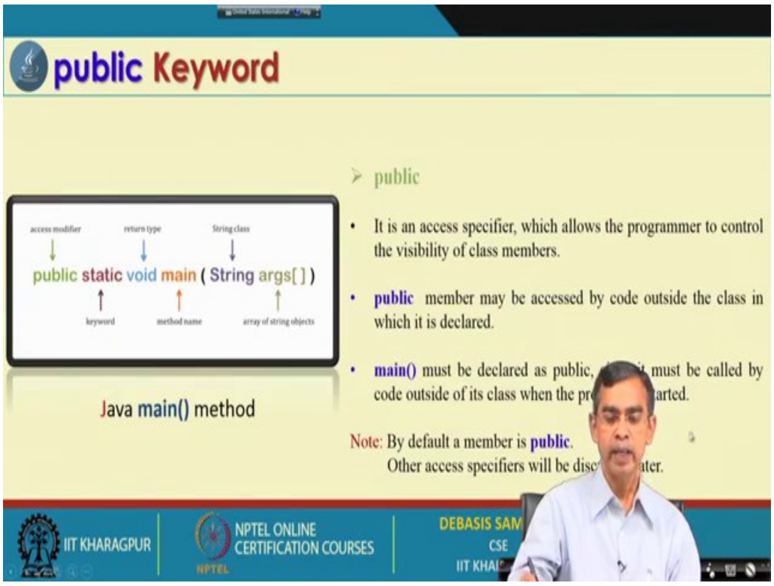
So, here the main class we don’t create any object therefore, the main method should be accessed without any object creation and that is why the static method to be there. So, we have understand about why it is a public. Public is an access specifier, static is the keyword. So, that object instantiation without any we can access this method, void that main method should return anything.

Next is String args, this is argument. Now, these are the argument that needs to be passed to a main method. Now this indicates that a variable number of arguments you can pass. So, that is an array it is there is a point at to array like all the pointer concept is not there java. So, it base there args is basically and array of any size. Now the size is automatically define when you run the program, we will discuss about automatic size declaration of this String args function.

So, here basically the input to the main method of type String and in java you know everything treat as a string whether integer, it is also consider as a string, a floating value is also string, any object is also string and then there is a manipulation where the different objects from can be transferred from string two types say, string to integer string to float like this one.

So, this is why the concept of argument here the string is declared. So, it is basically indicates that array of string objects. So, these are the few things are there we have already used it without knowing what is the reason for that, but this is basically the basic syntax of the java main method that it should be ok.

(Refer Slide Time: 10:59)



public Keyword

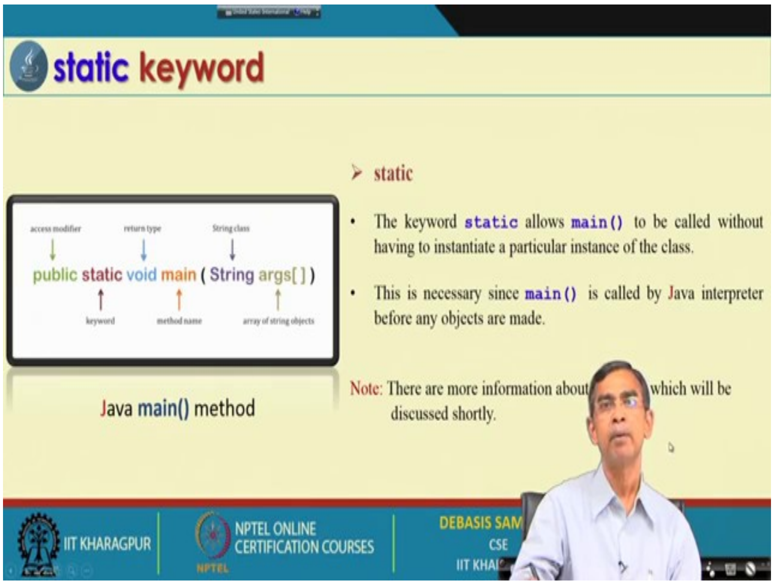
Java main() method

- public
  - It is an access specifier, which allows the programmer to control the visibility of class members.
  - public member may be accessed by code outside the class in which it is declared.
  - main() must be declared as public, must be called by code outside of its class when the pr arted.

Note: By default a member is public.
Other access specifiers will be disc ter.

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | DEBASIS SAM CSE IIT KHA

(Refer Slide Time: 11:01)



static keyword

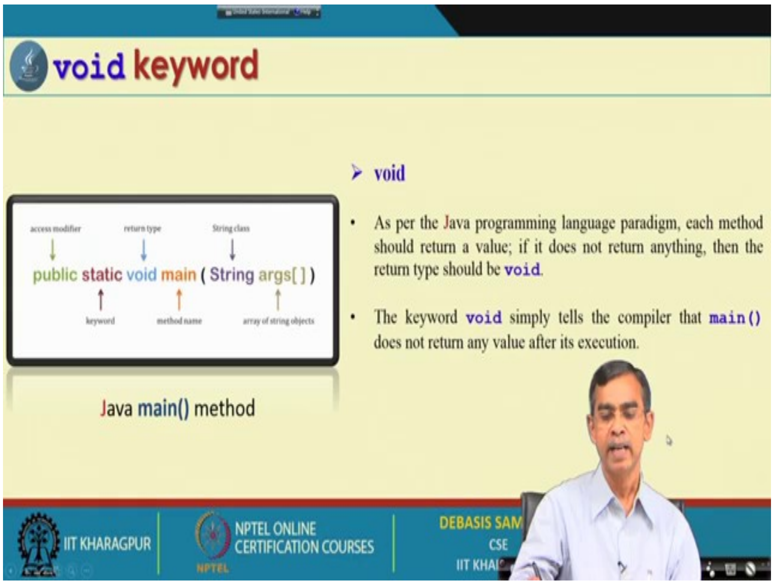
Java main() method

- static
  - The keyword `static` allows `main()` to be called without having to instantiate a particular instance of the class.
  - This is necessary since `main()` is called by Java interpreter before any objects are made.

Note: There are more information about which will be discussed shortly.

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | DEBASIS SAM CSE IIT KHA

(Refer Slide Time: 11:03)



void keyword

access modifier　return type　String class

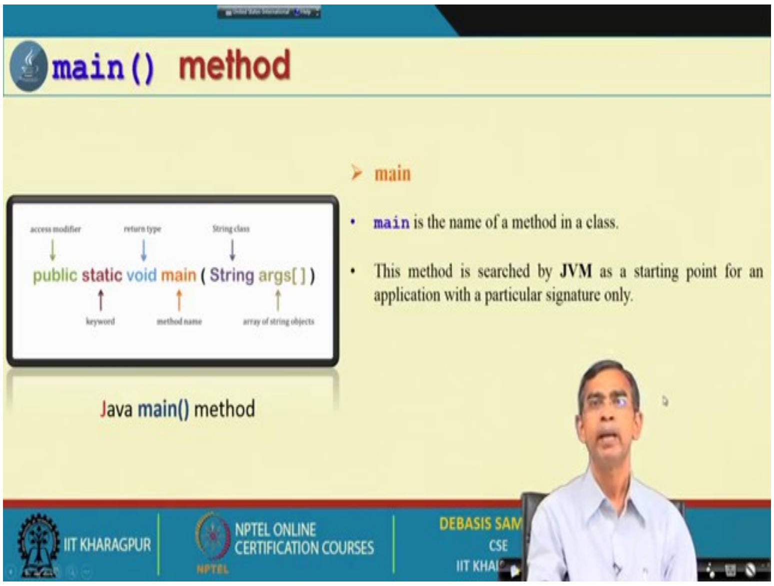
public static void main (String args[ ])

keyword　method name　array of string objects

Java main() method

- void
  - As per the Java programming language paradigm, each method should return a value; if it does not return anything, then the return type should be `void`.
  - The keyword `void` simply tells the compiler that `main()` does not return any value after its execution.

(Refer Slide Time: 11:05)



main () method

access modifier　return type　String class

public static void main (String args[ ])

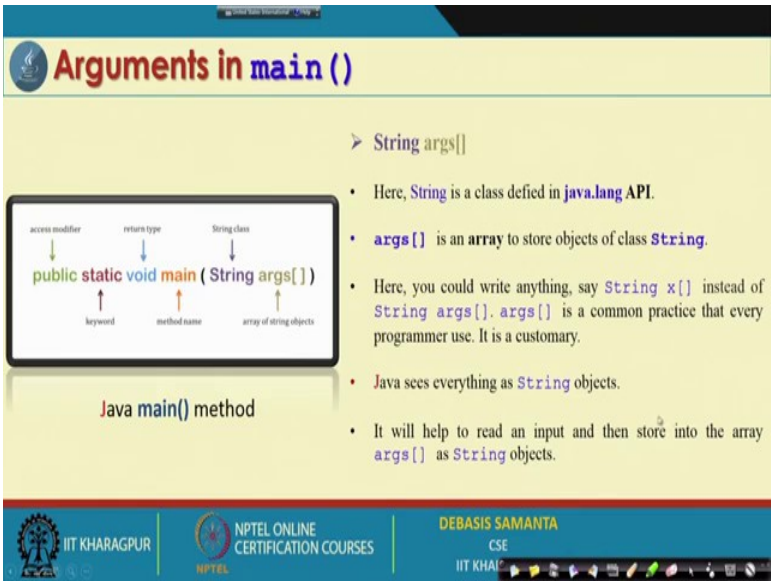
keyword　method name　array of string objects

Java main() method

- main
  - `main` is the name of a method in a class.
  - This method is searched by **JVM** as a starting point for an application with a particular signature only.

So, we have declared about public, concept static and then void. And then main method this one right.

(Refer Slide Time: 11:07)



Arguments in main ()

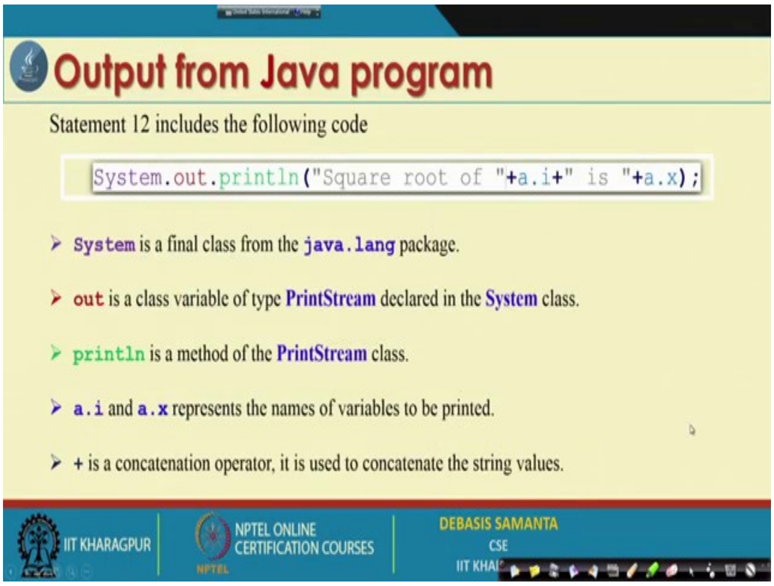
Java main() method

- String args[]
- Here, String is a class defied in java.lang API.
- args[] is an array to store objects of class String.
- Here, you could write anything, say String x[] instead of String args[]. args[] is a common practice that every programmer use. It is a customary.
- Java sees everything as String objects.
- It will help to read an input and then store into the array args[] as String objects.

IIT KHARAGPUR NPTEL ONLINE CERTIFICATION COURSES DEBASIS SAMANTA CSE IIT KHA

And then String args is basically passing input to its argument in a different way.

(Refer Slide Time: 11:13)



Output from Java program

Statement 12 includes the following code

```
System.out.println("Square root of "+a.i+" is "+a.x);
```

- System is a final class from the java.lang package.
- out is a class variable of type PrintStream declared in the System class.
- println is a method of the PrintStream class.
- a.i and a.x represents the names of variables to be printed.
- + is a concatenation operator, it is used to concatenate the string values.

IIT KHARAGPUR NPTEL ONLINE CERTIFICATION COURSES DEBASIS SAMANTA CSE IIT KHA

Now, output, we see in the program that output from a program usually displayed on the screen if you want to display some output on the screen, then we use one method. This method is called println method or its many variations println has its many variation like println, print ,printf ,like this one anyway.

So, println is a method. Now this method is again defined somewhere it this method is defined in the java dot lang package and java dot lang package has one class called the

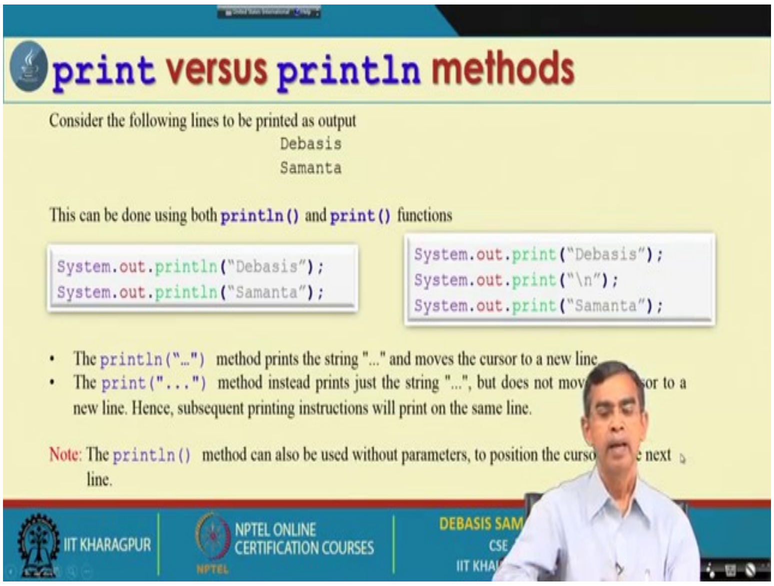
System. In the class, it is defined and in fact, for the System there is an output class and then in this outputs called the method is defined. So, it is the idea is that the println method which we used here it is defined here and the println method has again variable number of arguments as input. Here the argument is basically one string than one value and then another value.

So, the three arguments are there four arguments the other one string one value another string and another value. So, 4 arguments are there and so, the println method is customize to the any arguments actually one argument, no argument, many arguments whatever it is there. And if you want to use multiple arguments all the arguments should be separated by plus sign. So, in this case for example, so, this plus this mean this argument plus this argument plus this argument plus this argument and so on.

So, this is the basic syntax of the System.out.println() and this is the method. As I already told you defined in see java.lang package, in this java.lang package there is a out a class variable and this is the type of Printstream and in this Printstream is declared in a System class. All these things regarding the output system, Printstream and everything we have planned a detailed discussion when we will discuss about the input output stream concept. So, they are belong to input output stream.

So, here basically is an output stream actually because System.out.println(); output some string into screen actually. So, this the concept of the System.out.println() which we have frequently referred in our earlier example.

(Refer Slide Time: 13:43)



print versus println methods

Consider the following lines to be printed as output

```
Debasis
Samanta
```

This can be done using both `println()` and `print()` functions

```
System.out.println("Debasis");
System.out.println("Samanta");
```

```
System.out.print("Debasis");
System.out.print("\n");
System.out.print("Samanta");
```

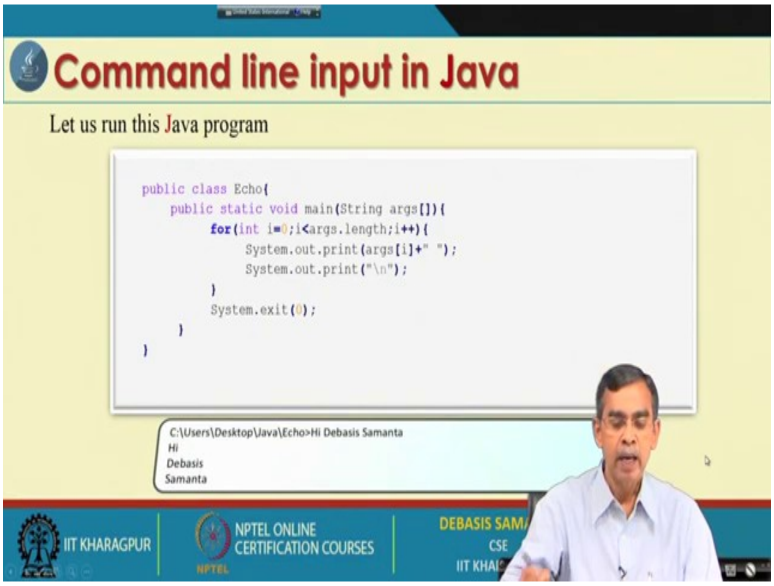
- The `println("...")` method prints the string "..." and moves the cursor to a new line
- The `print("...")` method instead prints just the string "...", but does not mov sor to a new line. Hence, subsequent printing instructions will print on the same line.

Note: The `println()` method can also be used without parameters, to position the curso next line.

The concept is like this and as I told you it has many variation println. Similarly print the difference between the println and print is that after this output the automatically console or the cursor will go to the next line, if we use this one and so, cursor will go automatically to next line. So, that is the concept of println. So, ln stand for next line. And if you use the print then cursor will not go to the next line, but if you use the print again force to your cursor to go to the next line, then you use this “\n” it is same as C printf function.

Now, in addition to this println and print similar to the print f that is we have used in C program also can be used is the same as printf. So, in that case System.out.printf() and then format it because you can customize your output to express into integer format, floating point format and everything. So, in our demo we will discuss about the use of println ,print and print a format in details. So, the idea about the println method it is there and now I will just discuss quickly about how we can fit input to our java program. So, during the execution of the program, how input can be given to the java program? So, this is called the java runtime data input concept.

(Refer Slide Time: 15:21)



Command line input in Java

Let us run this Java program

```
public class Echo{
    public static void main(String args[]){
        for(int i=0;i<args.length;i++){
            System.out.print(args[i]+" ");
            System.out.print("\n");
        }
        System.exit(0);
    }
}
```

```
C:\Users\Desktop\Java\Echo>Hi Debasis Samanta
Hi
Debasis
Samanta
```

Now, here is an example you see the first kind of input that can be given to java program is called the common line input. So, this program if you look it little bit carefully, you will be able to understand few more things which you have already learn earlier something extra here. Now here we have defined one class name; name of the class is the Echo and this is the main method as usual earlier right

Now, here you will be able to understand that what is the usage of stream args as its argument. Now purely you can recall if you declare an array, the name of the array .length return the size of the array. So, here in this example args.length, this is means this is the string array what the length is. So, this is arg length and then here we use a for loop; that means, if this loop will roll for all elements in the args array and then it basically print the arguments that is there in this array; that means, the different array objects the string type of objects, if it is there it will print it like this.

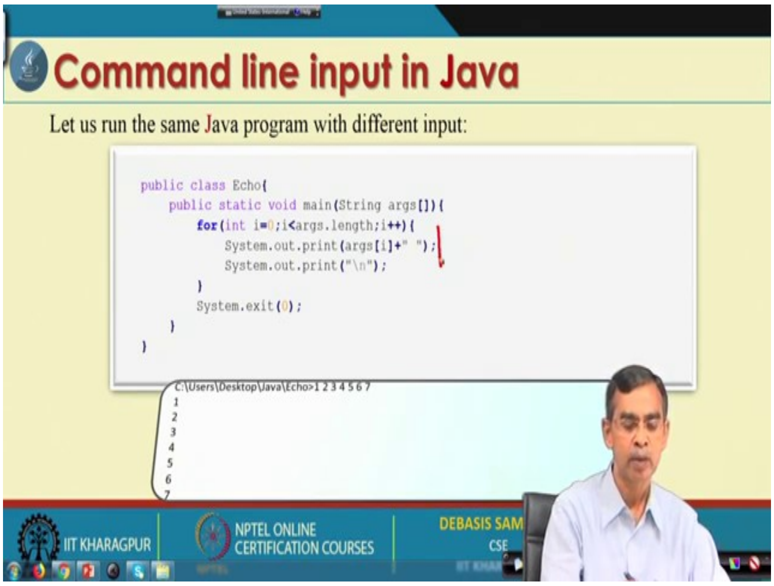
Now, here again question is that how this elements to the array can be given it to it. Now here is idea is that during the execution of this program, we can give the output to this one. For example, suppose the name of this class is Echo.java, it has been compile and then name of the compile class is Echo.class. So, we can run this Echo.class from your directory and then while you run this earlier that java Echo. Echo, then in addition to this java Echo. You can give anything you can type anything. For example, here we can run this program Echo and then giving three input hi, Debasis Samanta then it basically take the three string object.

Now, this first string object will go to the first location of the arguments array that is zeroth location. These will go to the second location of the array and then this will third. So, in this case the so zero zeroth location first location and second location and here the args array will be loaded with three strings hi Debassis and Samanta. They will be stored in the three different array locations and then size of the array is 3 , array index from 0 to 2.

So, args 0 we will store hi args1 will store Debasis and args 2 will store Samanta. So, now you come to this for loop, then this for loop start from i zero and then for the first print first System.out.print() will args 0 and then the space then go to the second i equals to 1, it will print args 1 then space, then second it will print args 2 and send and then finally, the new line n; you will basically go to the next line like.

So, this way if you run this program, it basically gives output which is from here. So, here we can understand that how we can pass input to the array while it is running.

(Refer Slide Time: 19:14)



Command line input in Java

Let us run the same Java program with different input:

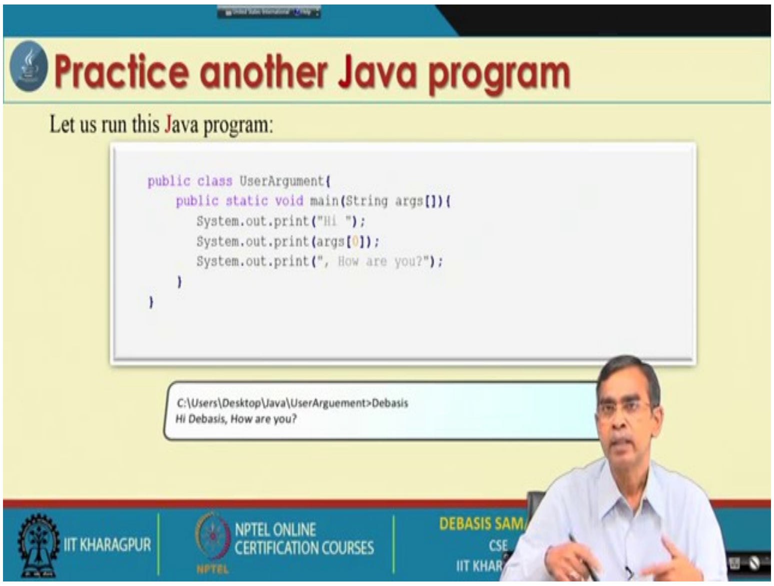
```
public class Echo{
    public static void main(String args[]){
        for(int i=0;i<args.length;i++){
            System.out.print(args[i]+" ");
            System.out.print("\n");
        }
        System.exit(0);
    }
}
```

```
C:\Users\Desktop\Java\Echo>1 2 3 4 5 6 7
1
2
3
4
5
6
7
```

Now, again the same program if you run with different input for example, say this is the program again same the input is like 1, 2, 3, 4, 5, 6, 7. So, here you can see the length is seven and then starting element is at args 0 and the last element is arg 6 and this loop if run, then it will print all the elements one by one as this one. So, this is output that it will print.

Now, if you run this program without any input simple. The earlier way you have run it, then the length will be 0 args dot length will be 0 and then this loop will not roll and it will not give any print statement to be executed. So, this is the concept the common line input to java and it is very useful one concept in java for the java programmer that they can use it.

(Refer Slide Time: 20:08)



Practice another Java program

Let us run this Java program:

```
public class UserArgument{
    public static void main(String args[]){
       System.out.print("Hi ");
       System.out.print(args[0]);
       System.out.print(", How are you?");
    }
}
```

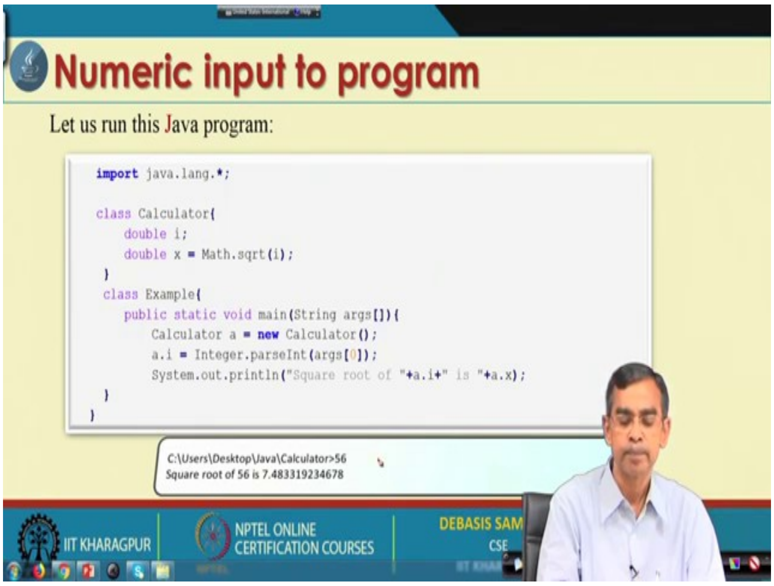
```
C:\Users\Desktop\Java\UserArguement>Debasis
Hi Debasis, How are you?
```

Now, let us see how we can practice another simple program you can understand it very quickly. Here is another is a common line input concept and this is the same thing as we have earlier. It will print the message from this print statement; it will print args 0 and this one.

Now, if you run this program run this program with different output it will look likes. For example, if you run this program say name user argument, then Debasis so, it will just print how Debasis, “How are You”. So, this way you can just take the input and then in your method in your operation belongs to that class all the input can be used for your own purpose that input can be string that input can be number, that input can be any values whatever it is there.

But in case of String args whatever the input you give, it will store in the form of a string type objects and latter on you will just convert it. Regarding this conversion, we have some discussion; we will discuss about it.

(Refer Slide Time: 21:11)



Numeric input to program

Let us run this Java program:

```
import java.lang.*;

class Calculator{
    double i;
    double x = Math.sqrt(i);
 }
 class Example{
    public static void main(String args[]){
        Calculator a = new Calculator();
        a.i = Integer.parseInt(args[0]);
        System.out.println("Square root of "+a.i+" is "+a.x);
 }
}
```

```
C:\Users\Desktop\Java\Calculator>56
Square root of 56 is 7.483319234678
```

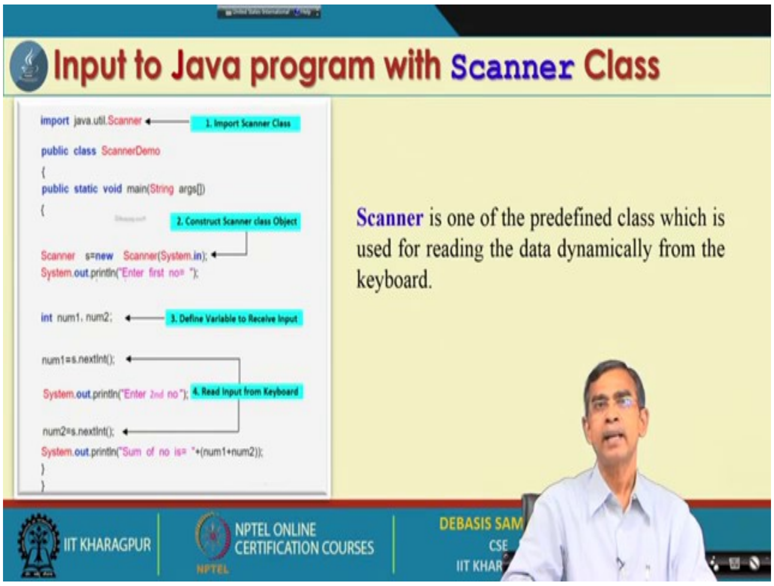
Here is an example, suppose as I told you, input is in the type String class; string objects right and then if it is an input integer then how we can get the integer value from it. So, these an example here. So, lets see this example again Calculator class that we have discussed earlier and then we create an object a of the calculator class like .

Now, the args 0, if we pass an input as an integer then args 0 will take as an input and then this method Integer.parseint() it is declared in again lang function lang packages. So, Integer.parseint() basically, it will take an string objects as an input and parse it and then after parsing it will return an integer value and then it will store in the a.i. So, here we can see that arguments that will be passed from the common line as an input will be passed to the objects value namely i here for a.i. So, the objects will be initialized once the initialized we will be able to call this. For example, for this object we can call this math square root function and then this is the syntax right.

So, here basically you can see the difference there. In the similar example which why we have discussed that we have initialize in the program itself statically, but here we can pass the value to a member elements data fields right while the program is execution .So, during runtime we can pass the input to it. So, this is the concept that is the input to the java program and here if the same program if you run it, then it will execute like here. If you give the input say five 6, then output it will produce which is shown here.

So, so this is the way the input irrespective of its type can be passed by means of while your invoking the program. So, here the think that during first invocation; that means, running the program you have to pass all the input that is required for your program. Now, there are more better way that instead of giving the input at the very beginning of the execution, we can give the input at the time of the execution.

(Refer Slide Time: 23:39)



Input to Java program with Scanner Class

```
import java.util.Scanner

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

Scanner  s=new  Scanner(System.in);
System.out.println("Enter first no= ");

int num1, num2;

num1=s.nextInt();

System.out.println("Enter 2nd no ");

num2=s.nextInt();
System.out.println("Sum of no is= "+(num1+num2));
}
}
```

1. Import Scanner Class
2. Construct Scanner class Object
3. Define Variable to Receive Input
4. Read Input from Keyboard

Scanner is one of the predefined class which is used for reading the data dynamically from the keyboard.

So, there are few more other way of doing this things. So, for this things a very good utility program is there. This is called the Scanner class. The Scanner class is defined in a package. It is also defined in package called util package. java dot util package and in that package one class call the Scanner class.

So, this basically you have to import it; if you want to use this Scanner class and then here is an example giving that how we can take the input; it is not a common line input. So, it can take the input point the program is in execution not just the before the starting of the execution. So, here this method if you look at this here. So, this scanner class has a constructor the Scanner and this constructor is initialized by an object call the System dot in as a parameter.

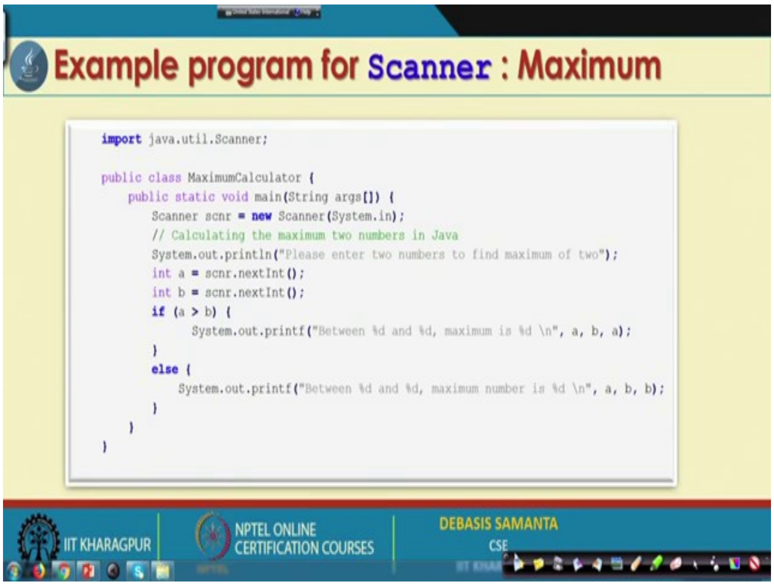
So, it is a default standard the System.in you have to specify because in the System one class dot in is a feel and it basically create a Scanner for this one. So, so once you call this i mean create an object of the Scanner, here the s is the object of type Scanner class which is basically call this and this s basically is responsible for taking input from your user. Now here for example, we give a form that enter a number first number.

Now so, the numbers whatever it will be enter from the keyboard. It will be stored into temporary location; now num1 and num2 we have declared; they are the temporary variable. It is declared here and then the object that s is basically created earlier is now ready to take the input from the keyboard. So, here is a syntax that s.nextint(). So, basically it will read the object as an integer from the keyboard and nextint is basically it will go on reading one after another. So, it is the nextint. So, as the first time is the first integer that is type from the keyboard we will read it and then it basically return this value and in store into the num1

So, num 1 is now initialize at this 1. Similarly it will give prompt that System.out.println() for the second number and then it is also read from the keyboard and then number will be returned and store in num2. So, this way using the Scanner class, we can read two numbers one by one at a time and they can be stored in the two the temporary location shear. And then finally, they can be manipulated for example, this will print the sum of the two numbers just we can read by this program.

So, this is the use of the scanner class and here is the one example that you can discuss about how it is basically execute.

(Refer Slide Time: 26:42)



Example program for Scanner : Maximum

```java
import java.util.Scanner;

public class MaximumCalculator {
    public static void main(String args[]) {
        Scanner scnr = new Scanner(System.in);
        // Calculating the maximum two numbers in Java
        System.out.println("Please enter two numbers to find maximum of two");
        int a = scnr.nextInt();
        int b = scnr.nextInt();
        if (a > b) {
            System.out.printf("Between %d and %d, maximum is %d \n", a, b, a);
        }
        else {
           System.out.printf("Between %d and %d, maximum number is %d \n", a, b, b);
        }
    }
}
```

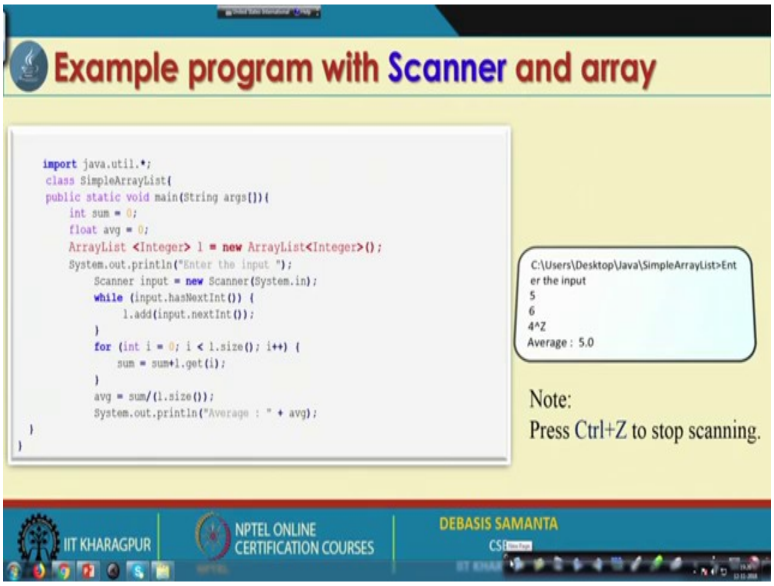
Now, this is the complete discussion of one class actually. Here we use the Scanner. So, you use this statement that you have to use the special class Scanner and this is the one example of a class MaximumCalculator which have the main method it is like this. So, this is the Scanner object we have to create as I told you, if you want to read from the keyboard.

So, this Scanner object is created and then we can read two numbers one by one using this nextint as you have already discussed and these two numbers will be stored into variables a and b and then it will basically print the largest value out of the numbers that we have print a pay a greater than b or others and then it will print like this. So, these are the typical print statement to print the value as for the logic; it is there.

So, this is the one example of is an alternative example to the common line argument that we have discussed here. And so for the things to things are there you have to import it. If you want to use the Scanner class, then you have to create a Scanner object using this one and then use this type of methods that Scanner object.nextint to read as many as the input you want to read it.

Now, definitely you have to mention that when you should stop it, I will tell you one example so, that you can understand.

(Refer Slide Time: 28:09)



Example program with Scanner and array

```
import java.util.*;
 class SimpleArrayList{
 public static void main(String args[]){
     int sum = 0;
     float avg = 0;
     ArrayList <Integer> l = new ArrayList<Integer>();
     System.out.println("Enter the input ");
         Scanner input = new Scanner(System.in);
         while (input.hasNextInt()) {
             l.add(input.nextInt());
         }
         for (int i = 0; i < l.size(); i++) {
             sum = sum+l.get(i);
         }
         avg = sum/(l.size());
         System.out.println("Average : " + avg);
  }
}
```

```
C:\Users\Desktop\Java\SimpleArrayList>Ent
er the input
5
6
4^Z
Average : 5.0
```

Note:
Press Ctrl+Z to stop scanning.

This is an example that how it can work it. So, this is the method that we have discussed. Now in this example we use again Scanner class and using the Scanner class we store a number of elements read from the keyboard and store into an array. So, is a typically this is the new way of defining an array. We will discuss about new definition of this array. So, this array list is already defined in util package and the integer this means array will store integer type of data and this is the name of the array l.

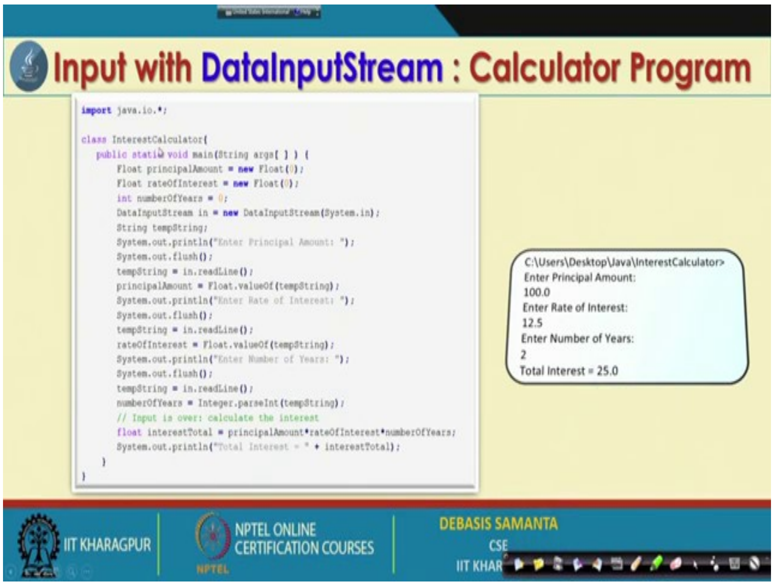
So, we declare using this a an object call the array object the name of the array object is l and using this syntax. So, this a very important syntax usually people use whenever they have to use read list or numbers from the keyboard anyway. So, this is the definition that the number of elements can be temporarily stored in an array like and then size of the array will be decided here shortly. Now here Scanner input, we create the new Scanner object like a s c n r in the last example and for the scanner object we use this input has nextint; that means, if it has the nextin. So, long basically it will go on reading one by one until you stop your reading entering the numbers.

So, it basically check that whether user has entered any number or not. If it has entered, then the same number in an integer format will be add into the l. So, l dot add is a one add method in the objective l and it basically put into their. So, here we can see using this concept, we will be able to read a number of objects number of numbers whether or any type of object share. For example, integer if you not integer is a float type, we can declare the float array whatever it is there. So, the number of objects of different type the user can specify and then can be read from the keyboard and can be stored here.

In this case we read numbers and then the same numbers can be processed here. For example, here you see we just simply read each numbers from the array l using the get method is basically read one number from the array at a time and then because of the find the total of the all i numbers and this basically calculate average. So, this is an example this shows that how a set of numbers can we read from the keyboard, can be stored in an array and then array can be used to process something. For example, here how to calculate the average here.

So, now let us see if you run this program then ok. How it will give the output it is like this? So, it basically ask that the enter the input, then you are typing 5, 6, 4 and stop entering into input you should press control z. So, control z is basically to indicate that termination of scanning. So, once the terminate, then it will do the program and then it will print at the average value for a. In this example for example, the input is 5, 6, 4the average value is 5.

(Refer Slide Time: 31:36)



Input with DataInputStream : Calculator Program

```
import java.io.*;

class InterestCalculator{
   public static void main(String args[ ] ) {
       Float principalAmount = new Float(0);
       Float rateOfInterest = new Float(0);
       int numberOfYears = 0;
       DataInputStream in = new DataInputStream(System.in);
       String tempString;
       System.out.println("Enter Principal Amount: ");
       System.out.flush();
       tempString = in.readLine();
       principalAmount = Float.valueOf(tempString);
       System.out.println("Enter Rate of Interest: ");
       System.out.flush();
       tempString = in.readLine();
       rateOfInterest = Float.valueOf(tempString);
       System.out.println("Enter Number of Years: ");
       System.out.flush();
       tempString = in.readLine();
       numberOfYears = Integer.parseInt(tempString);
       // Input is over: calculate the interest
       float interestTotal = principalAmount*rateOfInterest*numberOfYears;
       System.out.println("Total Interest = " + interestTotal);
   }
}
```

```
C:\Users\Desktop\Java\InterestCalculator>
Enter Principal Amount:
100.0
Enter Rate of Interest:
12.5
Enter Number of Years:
2
Total Interest = 25.0
```

So, this way if the Scanner input will work their and finally, another method which is also very useful method regarding this method, we will discuss in details when will discuss the input output stream in java. So, the method that is there as it is in the same context just I want to discuss shortly up to. So, is a data input stream and using the data input stream is just like a Scanner class like. So, we can create we can read some input from the keyboard here is an example just quickly said it. So, this is basically the data input stream is defined on package io package. So, java.io should be imported.

So, this is why if this is required and we are we are declaring one class or for on its called the interest calculator. Basically this program will read three values principal amount, interested and year and it will calculate the interest this one this is the. So, basically it we have to read three different values of different way amount is floating point value and year is maybe integer interest in a floating and finally, it will calculate interest and result will be printed on the screen. So, this is the concept that it is the program is there.

Now, here you see just how we have written this program. Here we can declare so to float objects on the principal amount and then rate of interest because they are the float type a new way of declaring float object, otherwise you can simply declared as a float type ,double type we have already declared. So, in that case also it can work anyway this is a new technique. This is the new way of declaring float object in java, you just learn it later on you will understand many things about creating the objects actually ok.

So, this the is a one way of creating float object and have also declare another variables call the number of years. It is an integer initialized at 0 and now here you see we declare one object call in object. This object of type data input stream. This is again like System dot in we have used in a Scanner class the standard and this data input stream class is defined in io package. So, we use this and then it basically create an object. So, that it is ready to read something from the keyboard just like Scanner class there.

Now, here is a tempString is a temporary string whatever it is read from the keyboard or something, it is read as a string as I told you. So, we will read the string and store in a temporary called tempString. Now this is prompt that and give the user from that entire principal amount is basically clear the buffer because whatever the user enters it will stored into the keyboard buffer. So, System.out.flush basically clean the buffer. So, that buffer is zero now or null then it basically for the in object that you have created using data input stream readLine.

So, whatever you type from the keyboard the entire things it will read and then read as a string. So, it will store in the tempString form and then this tempString will be returned as a float value and this is a function it is there just like integer dot ParseInt here also Float dot value of tempString. So, it basically scan the string and then its value is converted into the float and stored as a float object.

So, this is the concept. Similarly rate of interest plus and then read the keyboard buffer and rate of interest is read here. So, two value one is that principal amount is read and the rate of interest is from the keyboard. Next the number of years gain the same thing int read line and here you see we have to convert this into an integer, it will read as a string and using this conversion we convert this string into the integer object and number of years is the integer object.

So, the 3 input which we have read here as the principal amount, the rate of interest and number of years. On the 3 inputs are read from the users, we are now ready to calculate. Here we calculate the interest total as a float value and this is the formula it is for this is the simple multiplication. So, the value will be calculated and this value will be printed on the screen

So, idea it is like this and here is a quick demo so, that you can understand about how it works like. So, this is a example. Here we can run the program the number of program is InterestCalculator. So, it will ask user to type it, you will type it and then enter. So, it will take the value and then finally, it will complete its execution.

So, this is the way you can write about the different input ,output, streaming input. In fact, to a java program three way the input is possible we have discussed. These are mainly major way that we can use it. So, regarding other things, we will discussed next in our next class.

Thank you very much.