JAVA PROGRAMMING COURSE

FOREACH IN JAVA

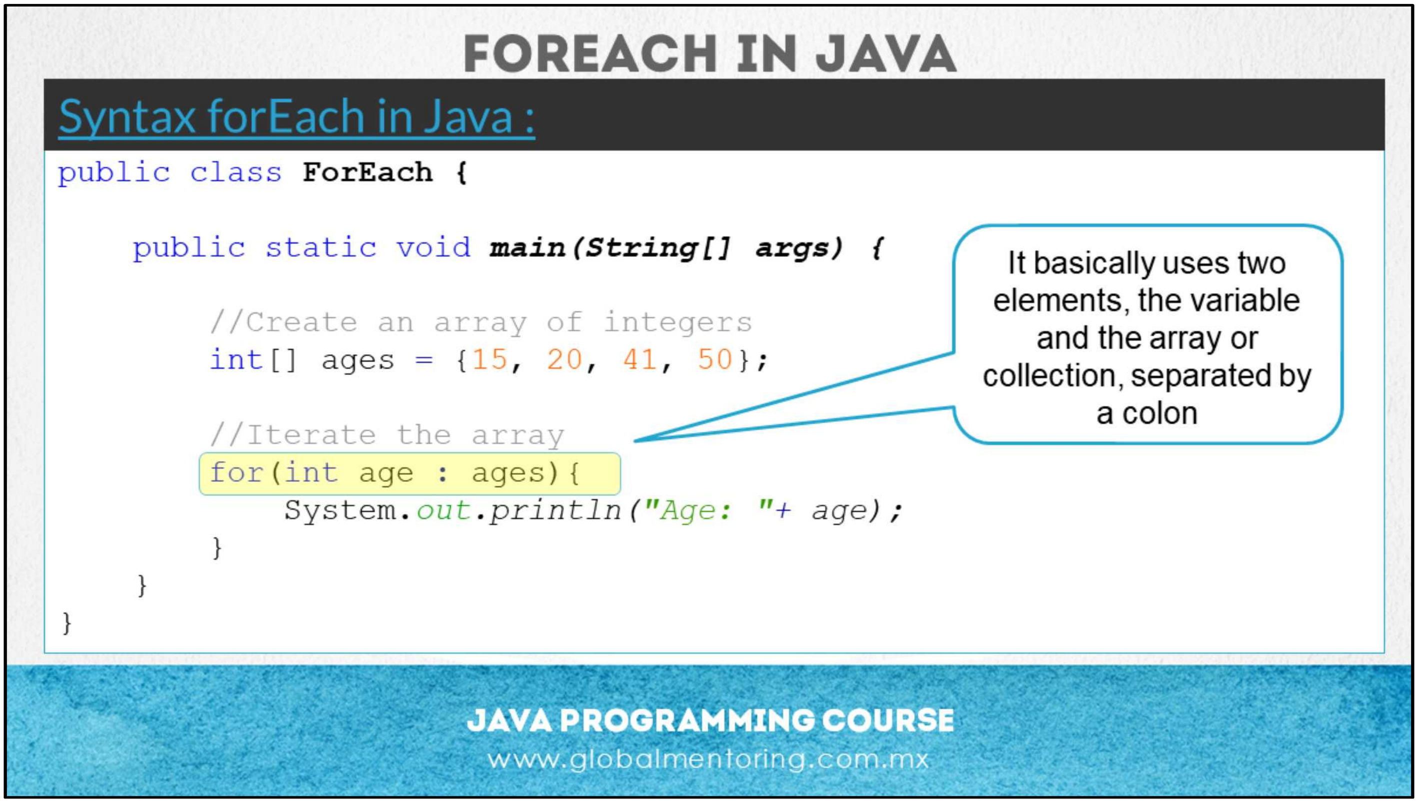
By the Expert: Ubaldo Acosta

www.globalmentoring.com.mx

Hello, Ubaldo Acosta greets you. Welcome again to this Java Programming Course.

In this lesson we are going to review the topic of forEach in Java.

Let's start immediately.



FOREACH IN JAVA

Syntax forEach in Java :

```
public class ForEach {

    public static void main(String[] args) {

        //Create an array of integers
        int[] ages = {15, 20, 41, 50};

        //Iterate the array
        for(int age : ages){
            System.out.println("Age:  "+ age);
        }
    }
}
```

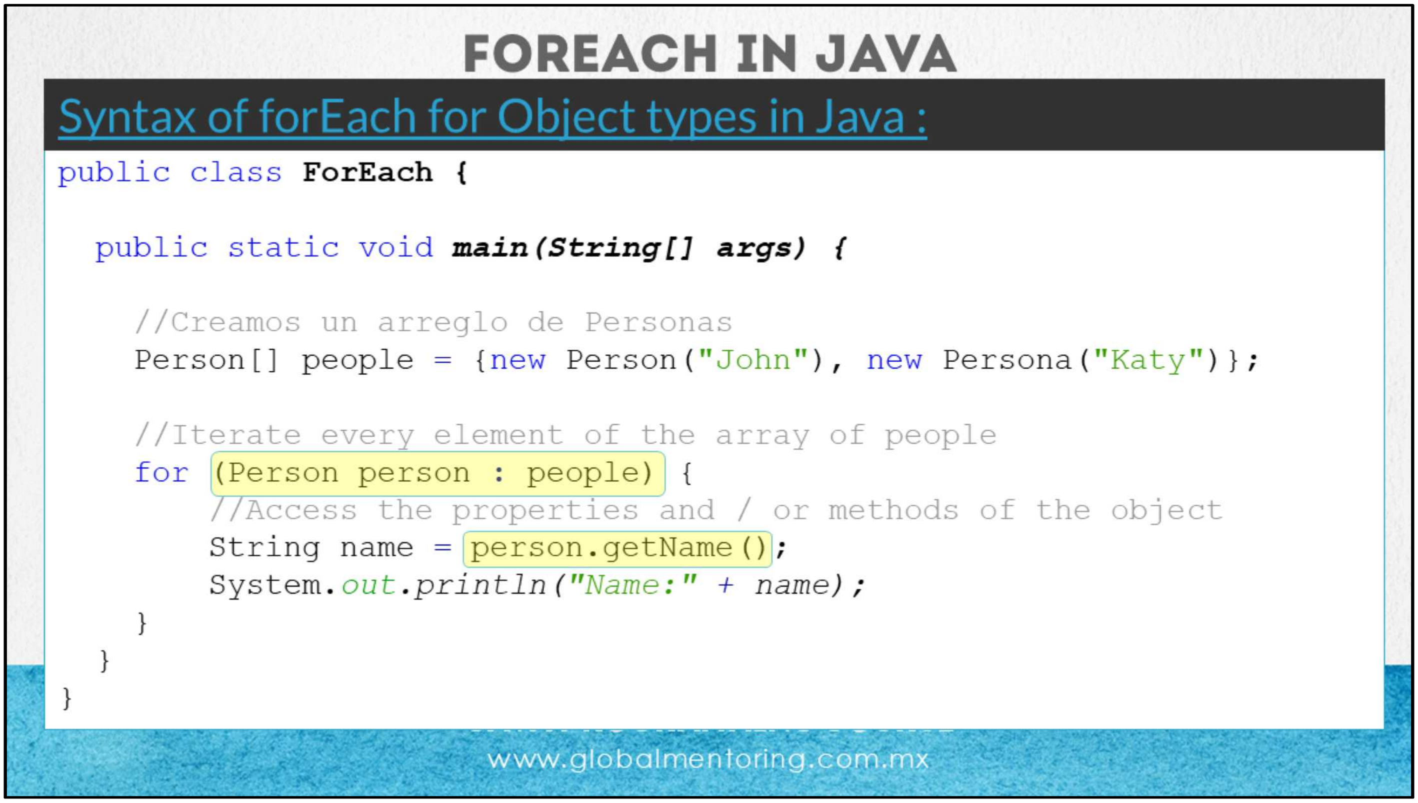
It basically uses two elements, the variable and the array or collection, separated by a colon

JAVA PROGRAMMING COURSE
www.globalmentoring.com.mx

In addition to the for loop which we already know, in which the number of elements to be iterated must be indicated, as well as having control of a counter, it is possible to use a for loop known as extended for or for each loop. This is a summary syntax for iterating the elements of an array or data collection.

This for each loop allows you to iterate easier the elements of an array or collection as it avoids errors by avoiding the declaration of counters to traverse the array. With this loop we can iterate arrays of any type of data, whether primitive types or Object type.

The syntax used is very simple, we only need to specify a variable of the data type of the array, which is used to store the elements of the array or collection one by one, and on the other hand we provide the name of the array or collection that we want to iterate. With this we can directly access the elements of the arrangement or collection one by one.



FOREACH IN JAVA

Syntax of forEach for Object types in Java :

```
public class ForEach {

  public static void main(String[] args) {

    //Creamos un arreglo de Personas
    Person[] people = {new Person("John"), new Persona("Katy")};

    //Iterate every element of the array of people
    for (Person person : people) {
        //Access the properties and / or methods of the object
        String name = person.getName();
        System.out.println("Name:" + name);
    }
  }
}
```

www.globalmentoring.com.mx

In this code we can see that it is also possible to work with object types, such as, for example, the Person type with which we have previously worked.

The syntax is the same, and we should only indicate again, the variable that will store the elements of the array one by one according to their data type, and once we have access to this type, then we can access the attributes or methods of the type that we have declared, for example, person.getName ();

The syntax of for each loop is recommended when we want to go through the elements of an array or collection in its entirety, but if we want to have more precise control and stop the execution of the loop once an element has been found, or have reached a certain index, then it is advisable to use a normal for loop, or even while or do-while depending on the case, since with all these loops we have access to the index and we know at all times the index value that is being traversed. This allows us to make decisions when going through each of the elements, and this does not happen in the for each loop since we have no control of any counter or index when traversing the elements.

Next we will see an example of this syntax.

ONLINE COURSE

JAVA PROGRAMMING

Por: Eng. Ubaldo Acosta

JAVA PROGRAMMING COURSE

www.globalmentoring.com.mx