ENTERPRISE APPLICATION DEVELOPMENT-1

Generics in Java

- Generics in java were introduced as one of features in JDK 5
- The angular brackets “<>” used in generics

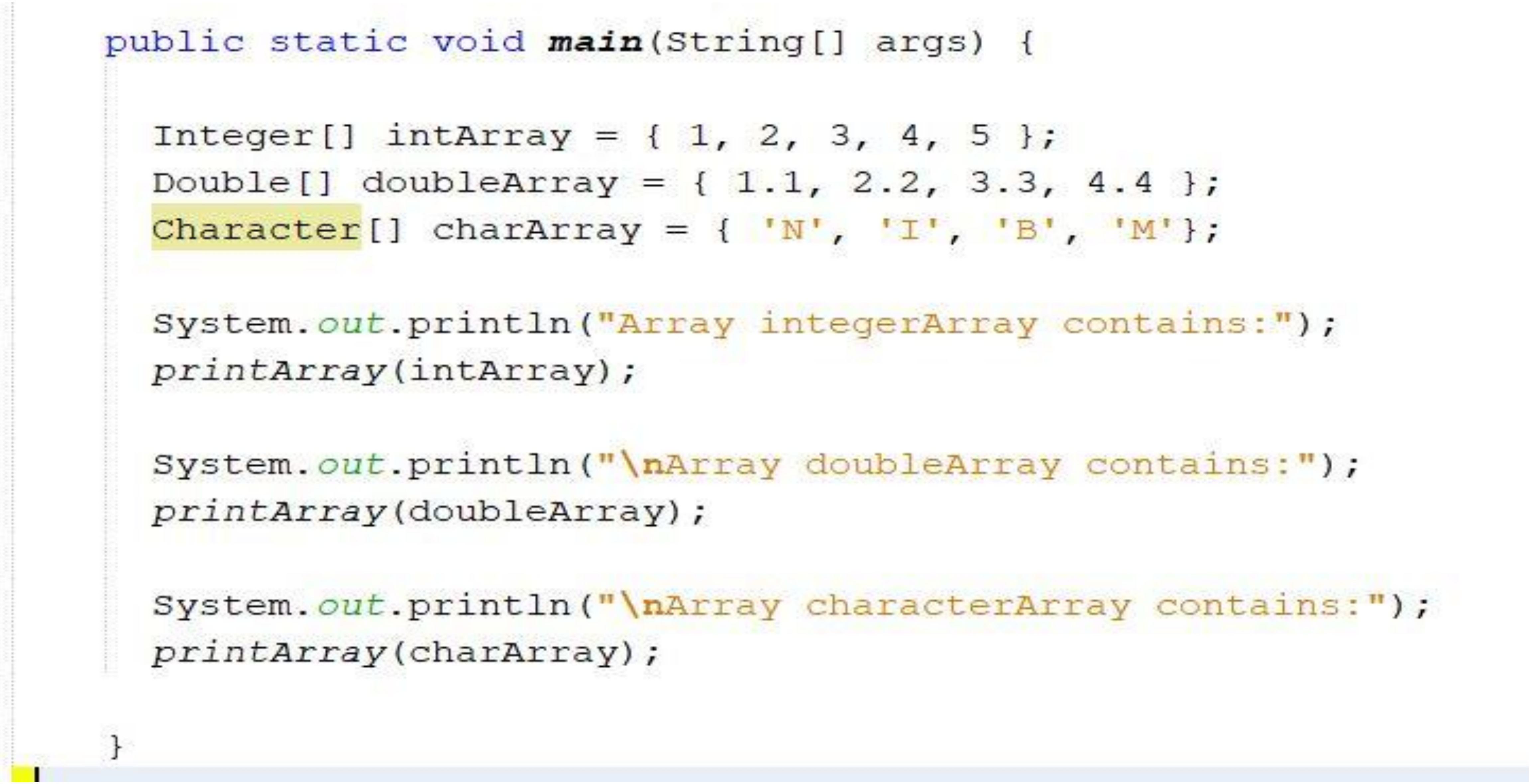
Java Generics Example

*The List interface represents a list of Object instances. This means that we could put any object into a List. Here is an example:

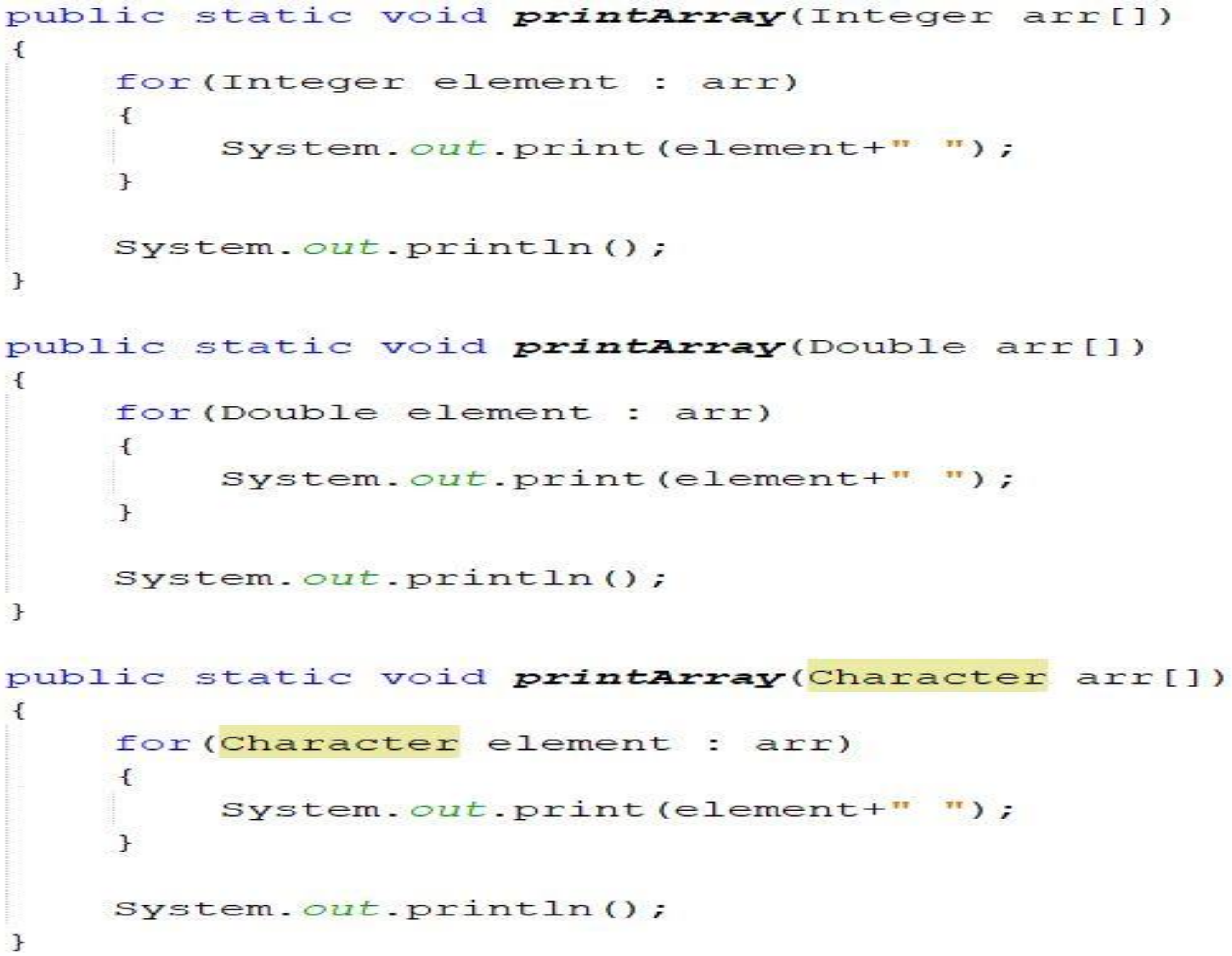
```
List list = new ArrayList();

list.add(new Integer(2));
list.add("a String");
```

*Because any object could be added, you would also have to cast any objects obtained from these objects. For instance:

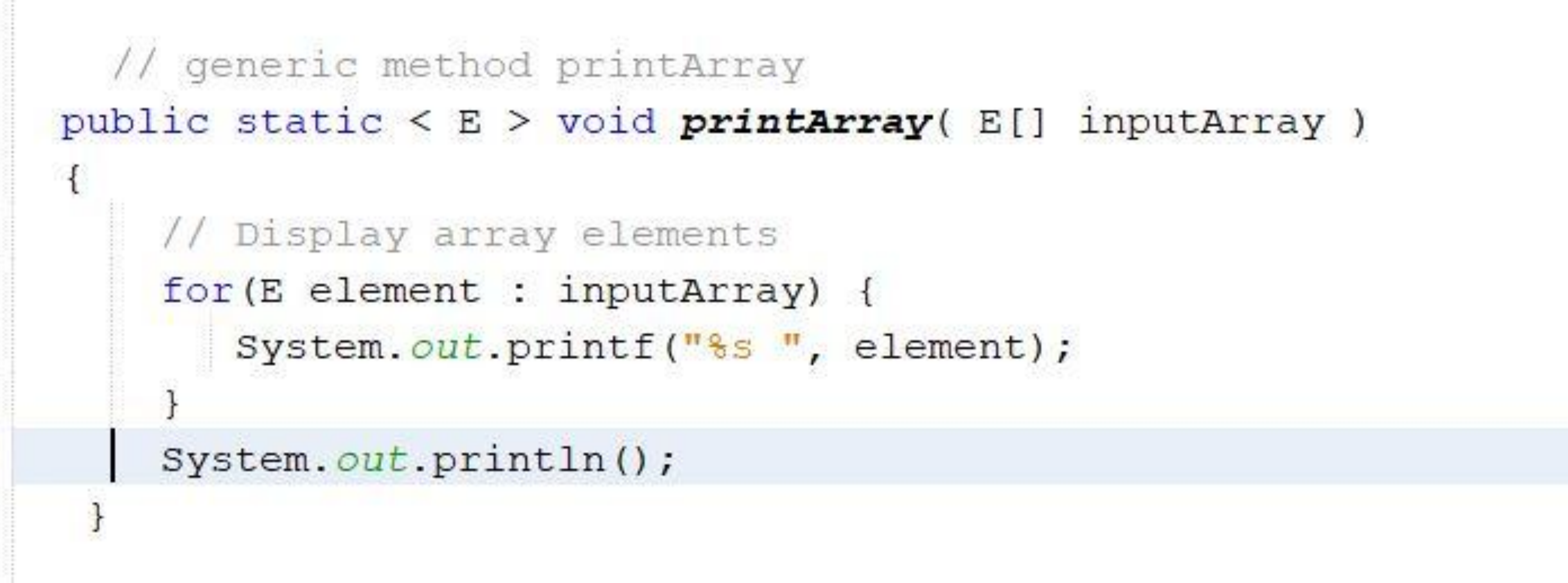
```
Integer integer = (Integer) list.get(0);

String string   = (String) list.get(1);
```



```
public static void main(String[] args) {

    Integer[] intArray = { 1, 2, 3, 4, 5 };
    Double[] doubleArray = { 1.1, 2.2, 3.3, 4.4 };
    Character[] charArray = { 'N', 'I', 'B', 'M'};

    System.out.println("Array integerArray contains:");
    printArray(intArray);

    System.out.println("\nArray doubleArray contains:");
    printArray(doubleArray);

    System.out.println("\nArray characterArray contains:");
    printArray(charArray);

}
```



```
public static void printArray(Integer arr[])
{
    for(Integer element : arr)
    {
        System.out.print(element+" ");
    }

    System.out.println();
}

public static void printArray(Double arr[])
{
    for(Double element : arr)
    {
        System.out.print(element+" ");
    }

    System.out.println();
}

public static void printArray(Character arr[])
{
    for(Character element : arr)
    {
        System.out.print(element+" ");
    }

    System.out.println();
}
```



```
  // generic method printArray
public static < E > void printArray( E[] inputArray )
{
    // Display array elements
    for(E element : inputArray) {
       System.out.printf("%s ", element);
    }
    System.out.println();
 }
```