For each loop syntax :

==========

```
for(data_type item : collection/array){
 ...
}
```

1. collection is a collection or array variable that you have to loop through.
2. item is a single item from the collection.

Example: 1

```
String[] arrData = {"Alpha", "Beta", "Gamma", "Delta", "Sigma"};
   for (String strTemp : arrData){
            System.out.println(strTemp);
   }
```

```
   for(int i = 0; i< arrData.length; i++){
            System.out.println(arrData[i]);
   }
```

output: Alpha
Beta
Gamma
Delta
Sigma

Example: 2

1.
```
   int[] numArray = {10, 20, 30, 40};
      for(int num : numArray){
        System.out.println(num);
      }
```

2.
```
 public static void main(String[] args) {
    List<Integer> numList = new ArrayList<Integer>();
      numList.add(10);
      numList.add(20);
      numList.add(30);
      numList.add(40);
     //foreach loop
       for(int num : numList) {
         System.out.println(num);
       }
  }
```

```
3.  public static void main(String[] args) {
      List<Integer> numList = new ArrayList<Integer>();
       numList.add(10);
       numList.add(20);
       numList.add(30);
       numList.add(40);

    //foreach loop with lambda
        numList.forEach( item -> System.out.println(item) );
    //Pass function reference
         numList.forEach( System.out::println );
  }

        output :  10
                  20
                  30
                  40
```

Example : 3

```
    class AssignmentOperator {
      public static void main(String[] args) {
        char[] vowels = {'a', 'e', 'i', 'o', 'u'};
        // foreach loop
         for (char item: vowels) {
            System.out.println(item);
         }
       }
      }
   -------------------------
   class ForLoop {
     public static void main(String[] args) {
        char[] vowels = {'a', 'e', 'i', 'o', 'u'};
        for (int i = 0; i < vowels.length; ++ i){
           System.out.println(vowels[i]);
        }
     }
   }

              output :   a
                         e
                         i
                         o
                         u
```