ArrayList & Wrapper Classes

▼ KEYPOINT TO LEARN THE TOPİCS PROPERLY

1. WHY => Why that topic exists in Java?
2. HOW => How you can implement it? (syntax and related rules)

These two question will basically help you about;

*"learning the topics" during the course
*"building your answer" during the interview process as well

▼ ARRAYLIST

- ArrayList is a class that is similar to an array and allows you to store "ONLY objects" NOT primitives
- ArrayList size is "adjustable". We can add or remove items from ArrayList
- ArrayList class in "java.util.package" and we have to import it

```
ArrayList<String> nameList = new ArrayList<>(); (SYNTAX)

nameList.add("Emre");
nameList.addAll( Arrays.asList( "Melek", "Ziya", "Ahmet"  )   )
```

```
ArrayList<Car> cars = new ArrayList<>(); (SYNTAX)!!

cars.add( new Car("Toyota") );

  // dataType may be String, Integer or any class that you already created
  //for example: Car class
```

- What are the differences between "Array vs ArrayList"? => "Popular Interview Questions”
- ArrayList Methods:

  -add(value)
  -add(index, value) // similar to insert
  -clear()
  -contains(value)
  -indexOf(value)

-lastIndexOf(value)
-get(index)
-size()
-set(index, value)

NOTE!!!

Do not confuse the following remove() methods, they have different purpose based on their parameters type

```
-remove(index) ==> arrayList.remove(3); //removes the object having index 3

-remove(object)==> arrayList.remove( Integer.valueOf(3) ); //removes the obje
ct
```

- Bulk Operations: They always require to be passed an Collection type (ArrayList)
  - addAll()

```
list.addAll(  Arrays.asList( array ) )

list.addAll(  Arrays.asList( "Java", "Python", "C#", "C++")   );

cars.addAll( Arrays.asList( new Car("Toyota"), new Car("Ford") )   );
```

  - removeAll()

```
list.removeAll(  Arrays.asList( "Java" ) ); // removes all Java

list.remove("Java"); // removes only the first Java object
```

  - Collections.replaceAll(list,"Java", "C#"); // replaces all the Java with C#
- Collections class: It has methods that we can use by passing ArrayList as its parameter:
  - -Collections.sort(arrayList)
    -Collections.reverse(arrayList)
    -Collections.max(arrayList)
    -Collections.min(arrayList)
    -Collections.frequency(numbers, 100)
    -Collections.swap(numbers, 3, 8) // in here 3 and 8 represents the index

of the realted objects
-Collections.replaceAll(numbers, 1, 100);

▼ WRAPPER CLASSES

Each primitive has a class (object) that is dedicated to it

▼ Why we use Wrapper Classes?

-convert to a primitive to an object
-because we can not use primitive for Collections

▼ "Auto boxing" and "un-boxing”

- Auto-boxing is converting from primitive to Wrapper Class object

```
Integer num1 = 1234;
int n = 5;
Integer num2 = n;
```

- Unboxing is converting from Wrapper Class object into primitive

```
Boolean b1 = new Boolean(true);
boolean b2 = b1;
```

Useful link:
https://beginnersbook.com/2014/09/java-autoboxing-and-unboxing-with-examples/

▼ Wrapper classes also come with "useful variables" and "methods”

Variables : MIN_VALUE, MAX_VALUE (especially finding the biggest/smallest number in an array)
Methods : valueOf, parseInt, isDigit, isAlphabetic, isLetter, isLowerCase, isLetterOrDigit

```
parseInt() --> returns a primitive value (IMPORTANT)
int count = Integer.parseInt("345356");

valueOf() --> returns a Wrapper class object (IMPORTANT)
Integer count2 = Integer.valueOf("56565");
```

▼ !!! NOTE: Casting IS NOT POSSIBLE WITH WRAPPER CLASSES

```
Integer num3 = Integer.valueOf(345);
Double d3 = num3; // gives you compile error
```

▼ SHORT REMINDERS

- JAVA is hard to digest in a few days
- No worries - Just keep on practicing as much as you can do and later you will see your improvement
- Before OOP, you should get the clear logic of Core Java topics:
  - -class-object relationship
    -methods
    -if-else and switch-case statements
    -for loop and for-each loop
- Automation starts soon (may be a few weeks later?)
  - -> more easy -> fun -> HW + Assignments -> firstly try to solve by your own,
    then discuss with your mates together in the group studies
  - !! IMPORTANT NOTICE !!
    - Please do not forget to change your OS language as "English"
    Otherwise you may face some issues while Automating your codes on IntelliJ