

Unit 4: Data Collections

Wrapper Classes

Adapted from:

- 1) Building Java Programs: A Back to Basics Approach
by Stuart Reges and Marty Stepp
- 2) Runestone CSAwesome Curriculum

This work is licensed under the
Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

<https://longbaonguyen.github.io>

Wrapper Classes

- A wrapper class takes an existing value of primitive type and “wraps” or “boxes” it in an object and provides a new set of methods for that type.
- It can be used in Java container classes that requires the item to be objects. (Arraylist)

Wrapper Classes

The wrapper class allows

1. A primitive value to be stored in an object (wrapping or boxing the primitive in a wrapper object). Java does this for you automatically, which is called autoboxing.
2. The primitive value to be retrieved again (unwrapping or unboxing from the wrapper object), which Java also does for you.

Wrapper Classes

You will need to know two wrapper classes:

1) Integer class

2) Double class

Wrapper Classes

Integer and Double are wrapper classes...not Rapper Classes.

These are Rapper Classes:

```
public class Tupac{...}
```

```
public class Biggie{...}
```

Ye

```
public class Kanye{...}
```

```
public class KendrickLamar{...}
```



Integer Class

The Integer class wraps a value of type `int` in an object.

Here is the one method you need to know:

`static int parseInt(String s)`: Returns the `int` that the string `s` represents.

The class also has two static variables: A Java integer uses 32 bits(0 or 1) of memory. One bit is used for the sign(+ or -). Thus:

`Integer.MIN_VALUE`— The minimum value represented by an `int` or `Integer`
 $= -2^{31} = -2147483648$

`Integer.MAX_VALUE`— The maximum value represented by an `int` or `Integer`
 $= 2^{31} - 1 = 2147483647$

Double Class

The Double class wraps a value of type double in an object.

Here is the one method you need to know:

`static double parseDouble(String s):` Returns the double that the string `s` represents.

Autoboxing: you never construct a wrapper object yourself. `Double d = 2.5;` is all you need.

Auto-Boxing and Unboxing

Auto-boxing is the automatic boxing of primitive types in their wrapper classes.

Auto-unboxing is the automatic conversion of a wrapper object back to its corresponding primitive type.

Java does both for you, so you never write the boxing code yourself. **The `Integer(int)` and `Double(double)` constructors, and the `intValue()` and `doubleValue()` methods, are no longer on the AP reference sheet.**

Autoboxing and Auto-unboxing

```
Integer a = 5;           // auto-boxing
int x = a;               // auto-unboxing, x = 5
int y = a + x;           // auto-unboxing, y = 10
```

```
Integer b = 7;           // auto-boxing
int n = Integer.parseInt("42"); // n = 42
Double d = 7.5;          // auto-boxing
```

```
double e = d;            // auto-unboxing
double f = d + 2.0;       // auto-unboxing
double g = Double.parseDouble("3.5");
```

References

- 1) Building Java Programs: A Back to Basics Approach by Stuart Reges and Marty Stepp
- 2) Runestone CSAwesome Curriculum:
<https://runestone.academy/runestone/books/published/csawesome/index.html>

For more tutorials/lecture notes in Java, Python, game programming, artificial intelligence with neural networks:

<https://longbaonguyen.github.io>