

Fundamentos de programación

Grado de Ingeniería del Software

Jorge García Gutiérrez jorgarcia@us.es F1.56

Hasta ahora...

- Encapsulación y 00
- Uso de Eclipse básico
- Estructura básica de un programa en Java







- Tipos básicos y wrappers.
- Variables y constantes
- If-else y switch
- While, for y for-extendido.
- Listas, conjuntos y Maps

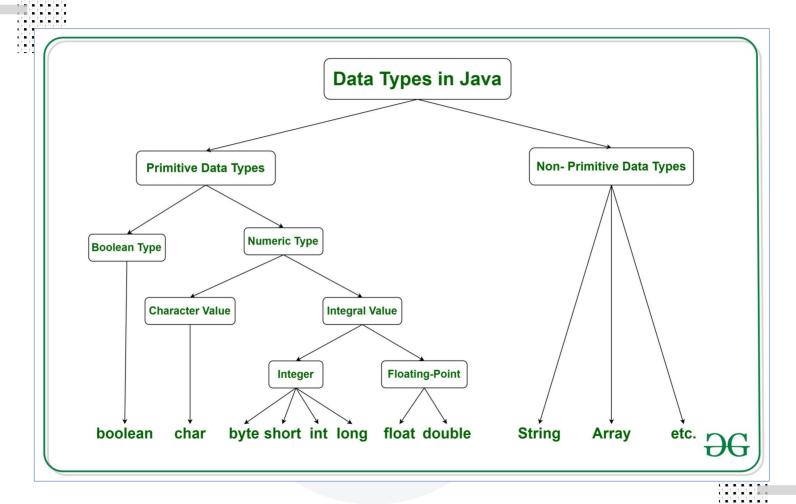




- Tipos básicos y wrappers.
- Variables y constantes
- If-else y switch
- While, for y for-extendido.
- Listas, conjuntos y Maps







Primitive Data Types	Wrapper Classes
int	Integer
float	Float
double	Double
boolean	Boolean (Added from 1.5)
short	Short
byte	Byte
char	Character
long	Long

.

.

https://dotnettutorials.net/lesson/wrapper-classes-in-java/

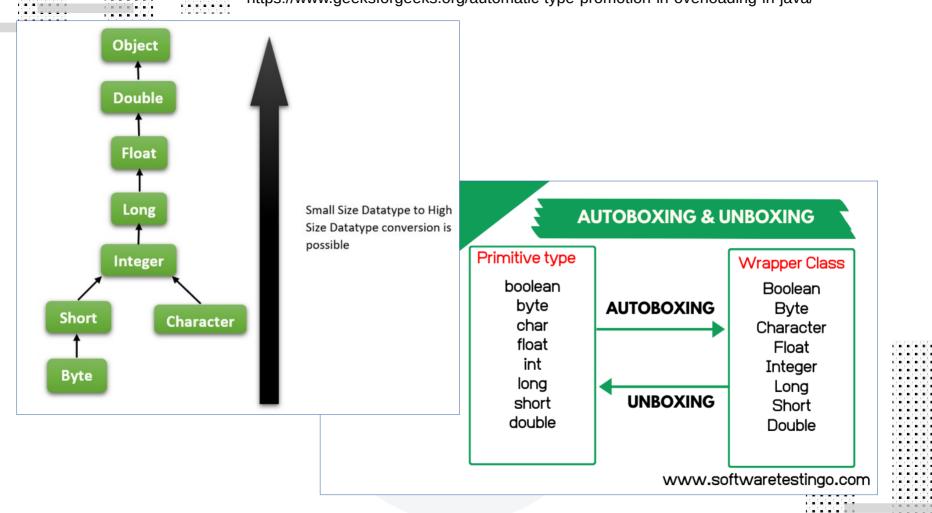
0.4040404040404

.

.

.

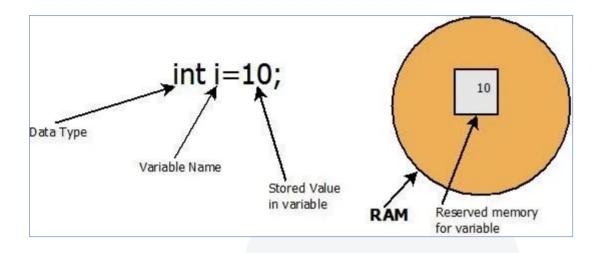
.



- Tipos básicos y wrappers.
- Variables y constantes
- If-else y switch
- While, for y for-extendido.
- Listas, conjuntos y Maps







https://www.stechies.com/variable-types-data-types-java/


```
public class Season2
{
    public static final int SPRING = 0;
    public static final int SUMMER = 1;
    public static final int FALL = 2;
```

.

0.4040404040404

C 4 (4)

The section of the se

.

.

.

.

```
class EnumExample1{
//defining enum within class
public enum Season { WINTER, SPRING, SUMMER, FALL }
//creating the main method
public static void main(String[] args) {
//printing all enum
for (Season s: Season.values()){
System.out.println(s);
System.out.println("Value of WINTER is: "+Season.valueOf("WINTER"));
System.out.println("Index of WINTER is: "+Season.valueOf("WINTER").ordinal());
System.out.println("Index of SUMMER is: "+Season.valueOf("SUMMER").ordinal());
```

.

.

103030303030303

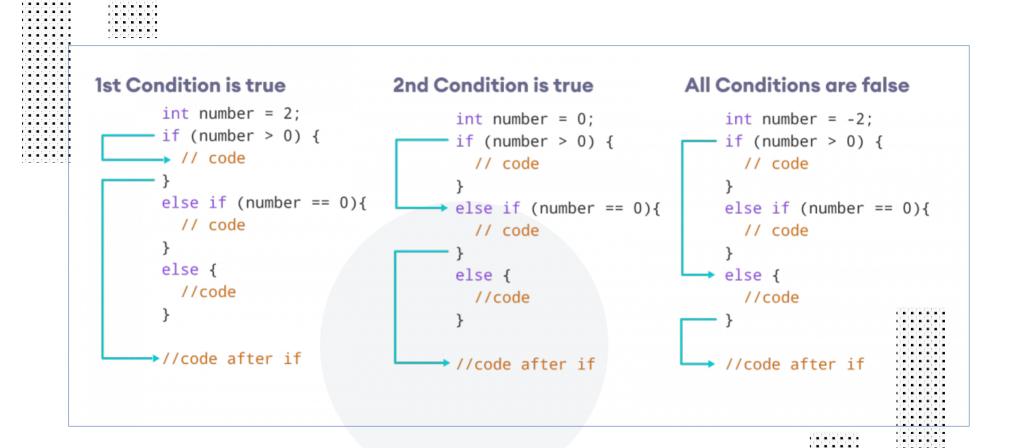
0.000

.

- Tipos básicos y wrappers.
- Variables y constantes
- If-else y switch
- While, for y for-extendido.
- Listas, conjuntos y Maps







```
int n = 10;
                                         = 10
switch (n) {
                                         = 90
    case 10:
        System.out.println("= 10");
                                         non of the above
    case 90:
        System.out.println("= 90");
    case -3:
        System.out.println("= -3");
    default:
        System.out.println("non of the above");
```

.

- Tipos básicos y wrappers.
- Variables y constantes
- If-else y switch
- While, for y for-extendido.
- Listas, conjuntos y Maps





```
package aulafacil;
     import java.util.Scanner;
 3
     public class AulaFacil {
         public static void main(String[] args) {
            int n = 0;
            while (n \le 10) // si n es menor o igual a 10, entra en el While
10
11
                System.out.println(n);
12
                n++; // Es lo mismo que n = n + 1
13
14
15
16
17
18
                                                                               AulaFacil.com
```

https://www.aulafacil.com/cursos/programacion/java-basico/bucles-while-l13525 :::

103030303030303

.

.

```
package aulafacil;

public class AulaFacil {

public static void main(String[] args) {

int i;

for (i = 0; i <= 10; i++) //Sentencia For
 {
    System.out.println(i); //instrucciones del bucle
    }
}

AulaFacil.com</pre>
```

.

.

.

.

.

.

.

.

.

.

.

.

.

.

```
public class WriteforEachLoops {
   public static void main (String[] args) {
       String[] names={"Regina","Stephen","Dave","Marsha"};
       System.out.println("For each loop output:");
       for (String name : names) {
            System.out.println(name);
       }
    }
}
```

.

.

.

.

.

.

.

- Tipos básicos y wrappers.
- Variables y constantes
- If-else y switch
- While, for y for-extendido.
- Listas, conjuntos y Maps





```
import java.util.List;
import java.util.ArrayList;
public class SimpleBuiltInListExample {
                                                          If we import, we can use
                                                          the "simple name"
  public static void main(String[] args) {
                                                          (ArrayList) as opposed to
    List<Integer> L = new ArrayList<>(); ←
                                                          the longer "canonical
    L.add(5);
                                                          name" (java.util.ArrayList).
    L.add(10);
    L.add(15);
    System.out.println(L);
```

```
Set<String> S = new HashSet<>();
S.add("Tokyo");
S.add("Beijing");
S.add("Lagos");
S.add("São Paulo");
System.out.println(S.contains("Tokyo"));
```

```
s = set()
s.add("Tokyo")
s.add("Beijing")
s.add("Lagos")
s.add("São Paulo")
print("Tokyo" in s)
```

```
$ java SetExample
true
$ python set_example.py
True
```

```
import java.util.Map;
import java.util.HashMap;
class map {
   public static void main(String[] args) {
        Map<String, Integer> ID = new HashMap<>();
       ID.put("Ten", 10):
       ID.put("Five", 5);
        System.out.println("Map: " + ID);
        System.out.println("Map Keys: " + ID.keySet());
        System.out.println("Map Values: " +ID.values());
        System.out.println("Map Entries: " + ID.entrySet());
        // Remove Elements from the map
        int val = ID.remove("Five");
        System.out.println("Removed Map Value: " + value);
```

103030303030303

0.000

.

.

.

.

.

TAREA:

.

- 1. Leer comprensivamente el tema 1 y 2
- 2. Intentar replicar los ejemplos directamente en Eclipse con lo visto en esta clase.