



## TP 3 : Intro a Java

Programación Orientada a Objetos II  
Comisión 2  
2º Cuatrimestre de 2024

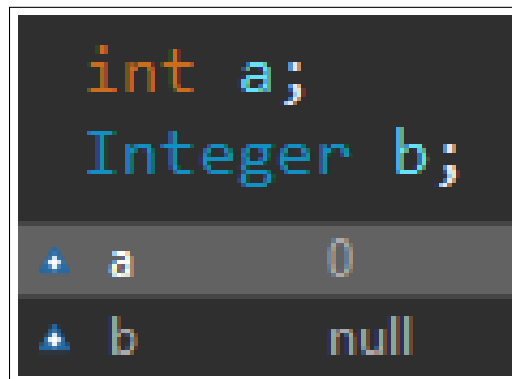
Nicolás Salvaneski

4. Las salidas que producen dichas líneas de código son las siguientes:

- `s.length()`; devuelve el número 3.
- `t.length()`; lanza un error, ya que `t` no está inicializado, no tiene valor, por lo tanto tampoco tiene longitud.
- `1 + a`; devuelve el String "1abc" ya que está casteando el 1 a '1' y concatenándolo con "abc".
- `a.toUpperCase()`; devuelve "ABC".
- `"Libertad".indexOf("r")`; devuelve 4, que es la posición del carácter 'r' en el String "Libertad". Los índices, al igual que en C, empiezan por el 0.
- `"Universidad".lastIndexOf('i')`; devuelve 7, que es el último índice en el que encontró un carácter 'i' en el String "Universidad".
- `"Quilmes".substring(2,4)`; devuelve el String "il" ya que es el substring que va desde el índice 2 hasta la 4 sin incluir.
- `(a.length() + a).startsWith("a")`; devuelve false, ya que al hacer `a.length()` esto da 3 y al concatenarlo con `a`, no empieza con "a", empieza con '3'.
- `s == a`; devuelve true, ya que está comparando por identidad (o referencia) y ambas variables apuntan al mismo dato en memoria.
- `a.substring(1,3).equals("bc")`; devuelve true, ya que compara por igualdad que ambas cadenas son iguales. `a.substring(1,3)` devuelve "bc", cadena que es igual a "bc".

5. Según la documentación de Java

1. Los tipos primitivos son tipos de datos predefinidos por el lenguaje que usan palabras reservadas, pero a diferencia de los tipos referenciales, no son objetos creados a partir de una clase. Su valor por defecto no es **null** como en los referenciales, ya que no se trata de un puntero a la memoria heap, sino que es un literal que representa al neutro de dicho tipo, por ejemplo, el número cero.
2. Integer es un wrapper de int, es decir, es un tipo referencial que corresponde a su tipo primitivo. Se utilizan para las colecciones, ya que estas solo pueden manejar objetos.
3. Si se define una variable de instancia de tipo int pero no se inicializa, su valor por defecto es 0. En cambio, la de tipo Integer se va a inicializar con valor por defecto null.



4. En el caso de las variables de método, no tienen un valor por defecto sino que el compilador lanza un error en tiempo de compilación donde indica que no fueron inicializadas.

7.
  1. En el sistema de archivos está la carpeta **model** y dentro las carpetas **gui** y **stack**.
  2. La relación entre el sistema de archivos y los nombres de los paquetes es que dentro de **src** y de **bin**, se organizan los directorios y subdirectorios de acuerdo a los nombres de los paquetes. Por ejemplo, el paquete `ar.edu.unq.tp2` corresponde a una estructura de carpetas `ar -> edu -> unq -> tp2`.  
Aunque sigan la misma estructura de carpetas, **src** y **bin** se diferencian en que la primera guarda los archivos de código **.java**, mientras que la segunda guarda sus correspondientes archivos compilados en **.class**, los cuales luego podrá interpretar la JVM (Java Virtual Machine).
10.
  1. El objeto cualquiera que le pide la edad a `Persona`, no conoce la implementación que permite calcularla. Este mecanismo de abstracción se llama Encapsulamiento.