

Python (I)

Programación (tics 100)
Semestre 01/2021

¿Qué veremos hoy?

Python básico, básico, básico



Programar

Lógica

- Definir los pasos a seguir (algoritmo)

Codificación

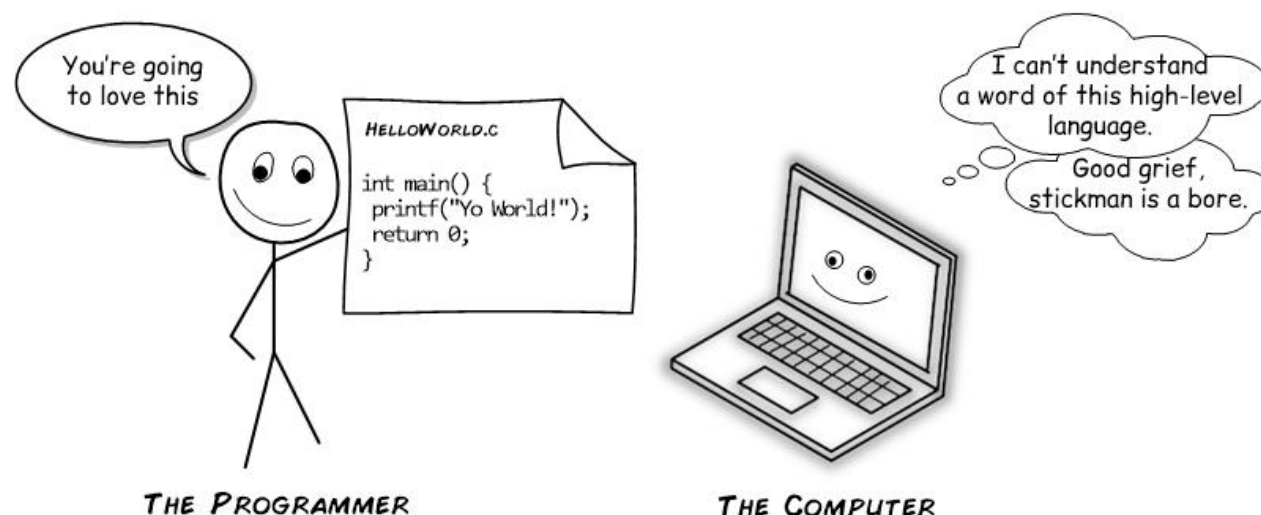
- Escribir el algoritmo como código

- Introducción a Python
 - Historia
 - Probando Python



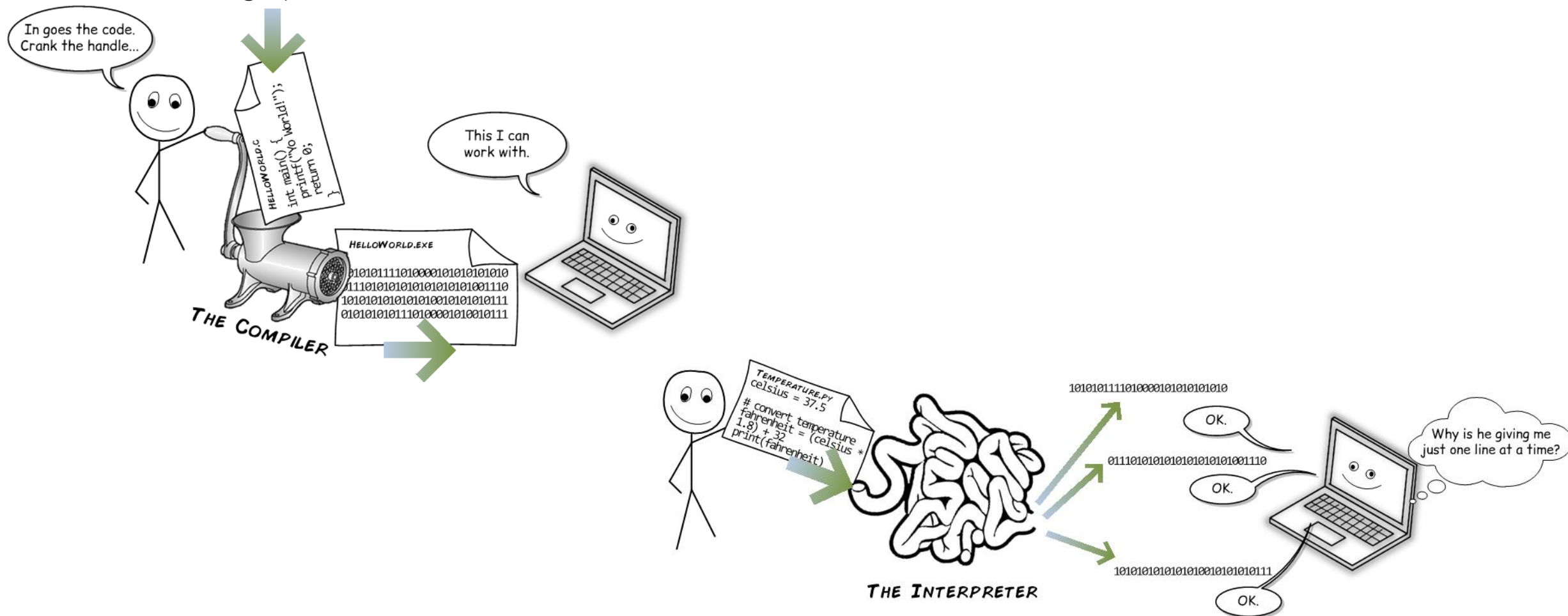
Un ordenador puede ejecutar una lista muy limitada de instrucciones:

- 8086: 127 instrucciones
- x86 (PentiumIV 32bits): 1500 instrucciones aproximadamente
- x64(i7): 3600+ instrucciones



Un Lenguaje de Programación es un **lenguaje entendible por un humano** que la máquina pueda transformar **en una secuencia de instrucciones binarias** que ella entienda!

Existen lenguajes COMPILADOS e INTERPRETADOS



Assembler x86

```

; _memcpy(dst, src, len)
; Copy a block of memory from one location to another.
;
; Entry stack parameters
;     [BP+6] = len, Number of bytes to copy
;     [BP+4] = src, Address of source data block
;     [BP+2] = dst, Address of target data block
;
; Return registers
;     AX = Zero

0000:1000                org      1000h      ; Start at 0000:1000h

0000:1000      _memcpy  proc
0000:1000 55          push    bp           ; Set up the call frame
0000:1001 89 E5       mov     bp,sp
0000:1003 06          push    es           ; Save ES
0000:1004 8B 4E 06    mov     cx,[bp+6]      ; Set CX = len
0000:1007 E3 11       jcxz    done         ; If len = 0, return
0000:1009 8B 76 04    mov     si,[bp+4]      ; Set SI = src
0000:100C 8B 7E 02    mov     di,[bp+2]      ; Set DI = dst
0000:100F 1E         push    ds           ; Set ES = DS
0000:1010 07         pop     es

0000:1011 8A 04      loop    mov     al,[si]      ; Load AL from [src]
0000:1013 88 05       mov     [di],al      ; Store AL to [dst]
0000:1015 46         inc     si           ; Increment src
0000:1016 47         inc     di           ; Increment dst
0000:1017 49         dec     cx           ; Decrement len
0000:1018 75 F7       jnz     loop         ; Repeat the loop

0000:101A 07      done    pop     es           ; Restore ES
0000:101B 5D         pop     bp           ; Restore previous call frame
0000:101C 29 C0       sub     ax,ax         ; Set AX = 0
0000:101E C3         ret              ; Return
0000:101F          end proc

```

c

```

char dst;

char src;

int len;

dst=0x0f;

src=0x1c;

len =2;

memcpy(dst,src,len);

```

Python

```

a=b

```

Cronología

Guido Van Rossum

(creador de Python)

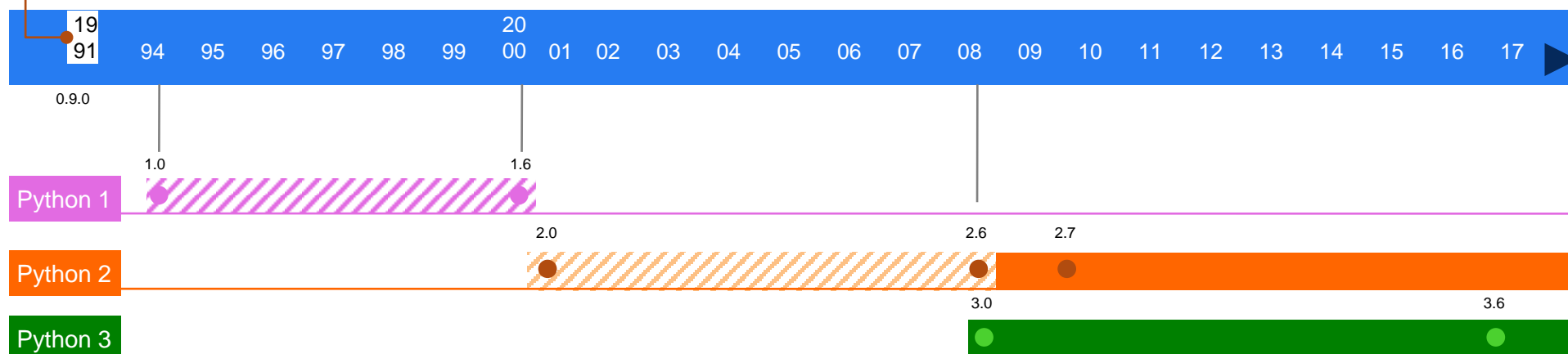
“es un científico de la computación conocido por ser el autor del lenguaje de programación Python en 1991” (fuente: Wikipedia)



Python es un lenguaje de propósito general y de tipo script. Es empleado para aplicación de interfaces, bases de datos, computación científica, aplicaciones web (client-server web programming) entre otras.





















Principales características:

- Facilidad de lectura (Código Interpretado)
- Sistema de tipo dinámico
- Manejo automático de administración de memoria



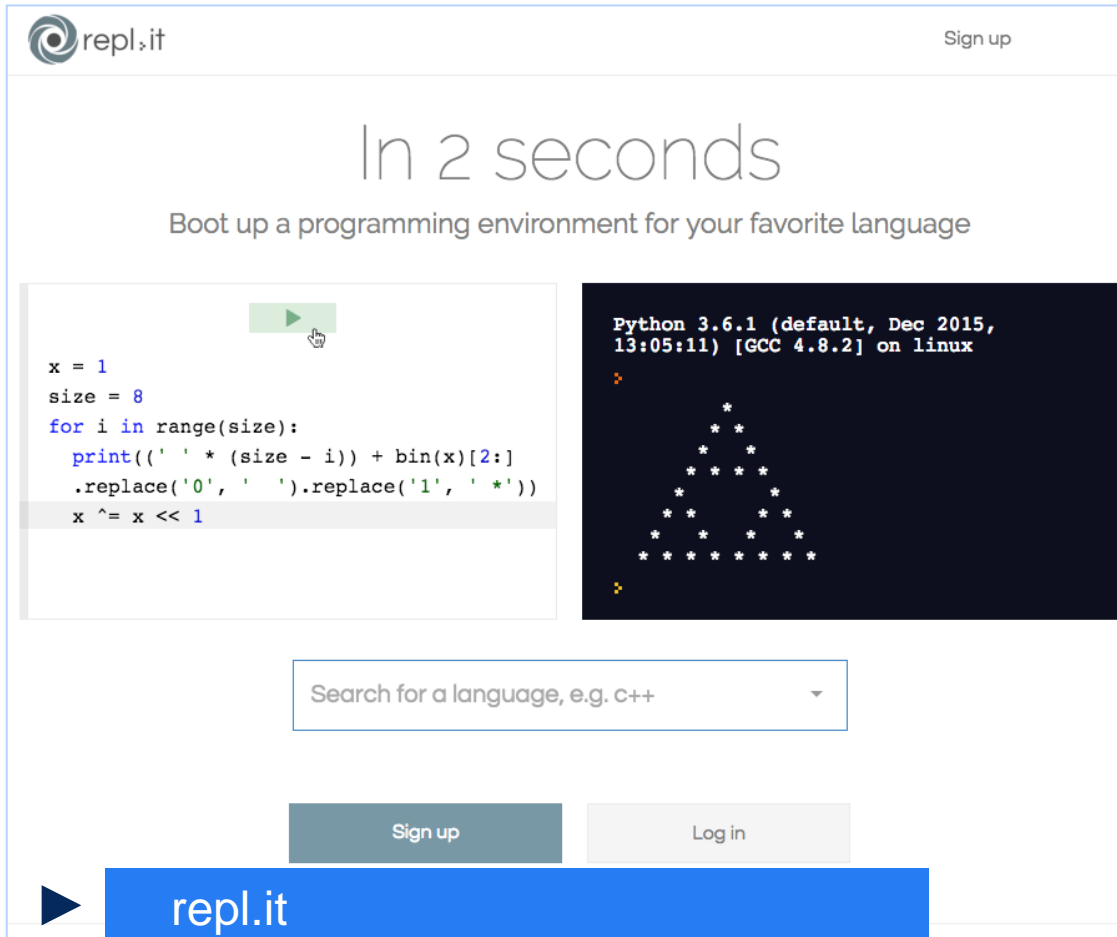
Popularidad

2021

Rank	Language	Type	Score
1	Python ▼	  	100.0
2	Java ▼	  	95.3
3	C ▼	  	94.6
4	C++ ▼	  	87.0
5	JavaScript ▼		79.5
6	R ▼		78.6
7	Arduino ▼		73.2
8	Go ▼	 	73.1
9	Swift ▼	 	70.5
10	Matlab ▼		68.4

online

- Ambientes de trabajo



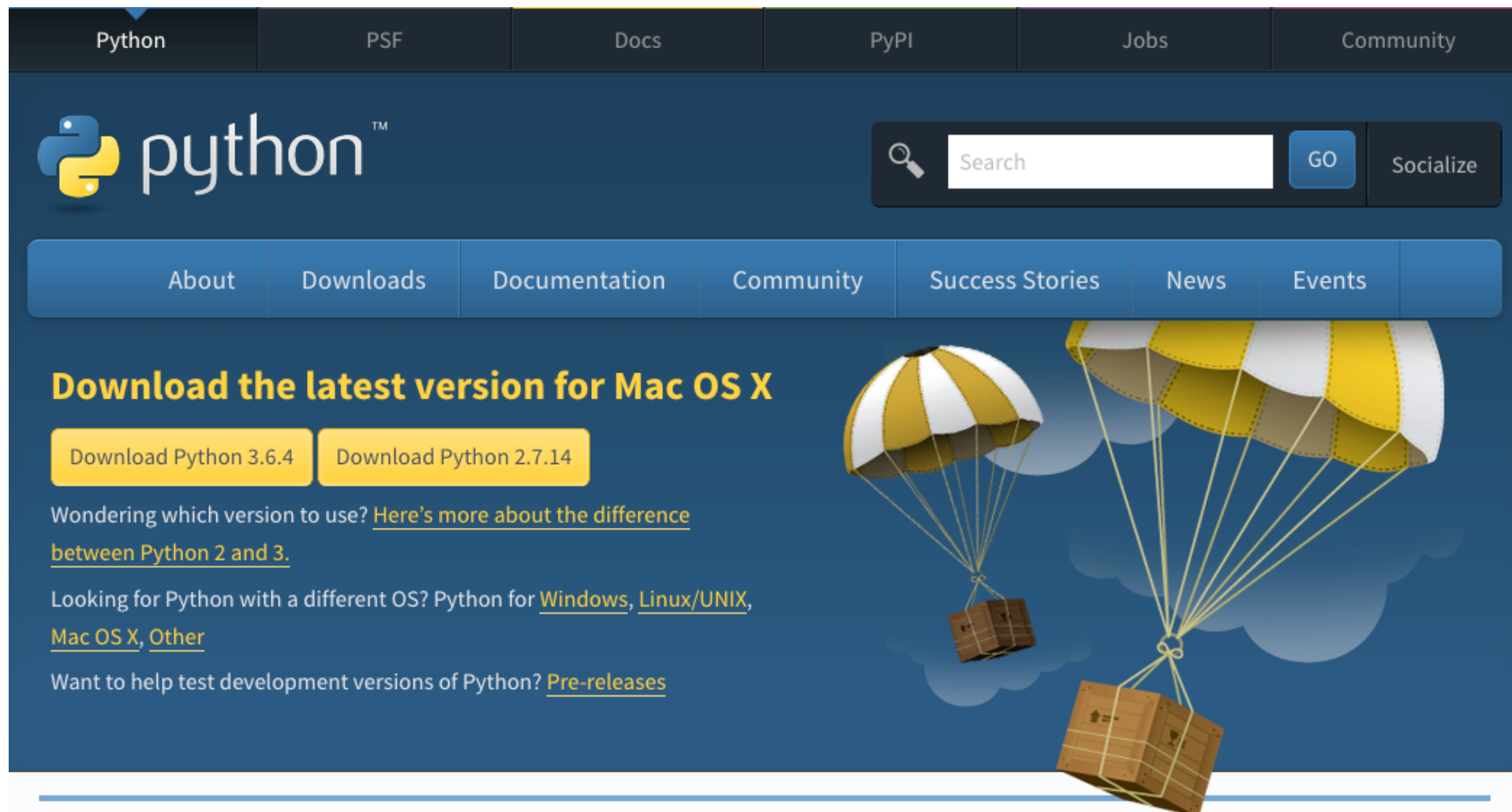
The screenshot shows the repl.it website. At the top left is the repl.it logo, and at the top right is a "Sign up" link. The main heading says "In 2 seconds" followed by the subtext "Boot up a programming environment for your favorite language". Below this, there are two panels. The left panel shows a code editor with Python code:

```
x = 1
size = 8
for i in range(size):
    print(' ' * (size - i) + bin(x)[2:]
        .replace('0', ' ').replace('1', ' *'))
x ^= x << 1
```

 A green play button is above the code. The right panel shows a terminal window with the text "Python 3.6.1 (default, Dec 2015, 13:05:11) [GCC 4.8.2] on linux" and a pattern of asterisks forming a triangle. Below these panels is a search bar with the placeholder text "Search for a language, e.g. c++". At the bottom are "Sign up" and "Log in" buttons. A blue banner at the very bottom contains the repl.it logo and the text "repl.it".

Instalación

offline



The screenshot shows the Python.org website with a dark blue header and navigation bar. The main content area features the Python logo and a search bar. Below the navigation bar, there is a section titled "Download the latest version for Mac OS X" with two yellow buttons: "Download Python 3.6.4" and "Download Python 2.7.14". To the right of the text, there is an illustration of two parachutes carrying boxes. The text below the buttons provides links for more information and for downloading Python for other operating systems.

Python PSF Docs PyPI Jobs Community

python™

Search GO Socialize

About Downloads Documentation Community Success Stories News Events

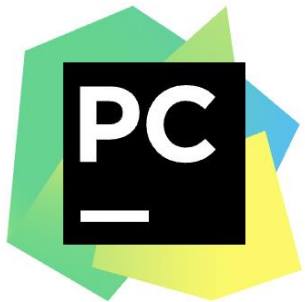
Download the latest version for Mac OS X

Download Python 3.6.4 Download Python 2.7.14

Wondering which version to use? [Here's more about the difference between Python 2 and 3.](#)

Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [Mac OS X](#), [Other](#)

Want to help test development versions of Python? [Pre-releases](#)



Versión: 2021.2
Build: 212.4746.96
27 de julio de 2021

[Requisitos del sistema](#)[Instrucciones de instalación](#)[Otras versiones](#)

Descargar PyCharm

[Windows](#)[macOS](#)[Linux](#)

Professional

Para desarrollo de Python tanto científico como de web. Compatible con HTML, JS y SQL.

Descargar

.dmg (Intel)



Prueba gratis

Disponible para Intel y Apple Silicon

Community

Para un desarrollo Python puro

Descargar

.dmg (Intel)



Gratis, código abierto

Disponible para Intel y Apple Silicon

- 
- The collage image is a composite of three elements. On the left, the Python logo is positioned above a candlestick chart with a white moving average line. On the right, a snippet of Python code is displayed on a dark blue background. The code defines a function `getFidAndValue` that iterates over a dictionary `self.FidValue`, filtering for 'DATE' types and calculating dates and values.
- ```

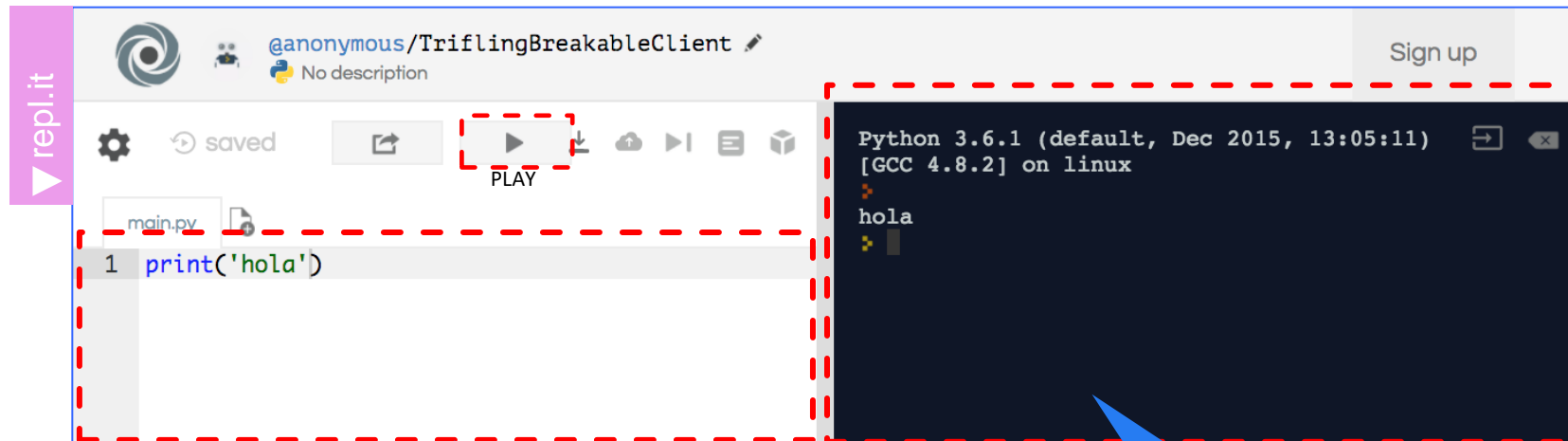
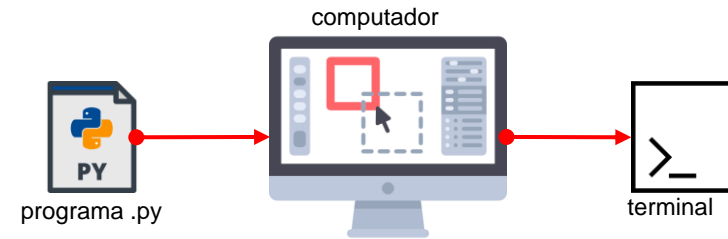
def getFidAndValue(self):
 #Read item in dictionary
 for key, value in item.FidValue.items():
 typeOfFID = mapFidType[key]
 if (typeOfFID == "DATE"):
 d = datetime.datetime.strptime(str(value), "%Y-%m-%d")
 dataCal = datetime.date.strptime(str(value), "%Y-%m-%d")
 FidAndValue = FidAndValue + str(d) + "\n"
 else:
 FidAndValue = FidAndValue + str(value) + "\n"
 return FidAndValue

```

# Lección 01



Un programa en Python está conformado por **líneas de texto** que contienen **una o más instrucciones y sentencias del lenguaje**



Esta zona podemos escribir las instrucciones y sentencias, y luego ejecutarlas haciendo clic en el botón PLAY

Esta zona se conoce como **terminal**. En ella se muestran los resultados de nuestro programa

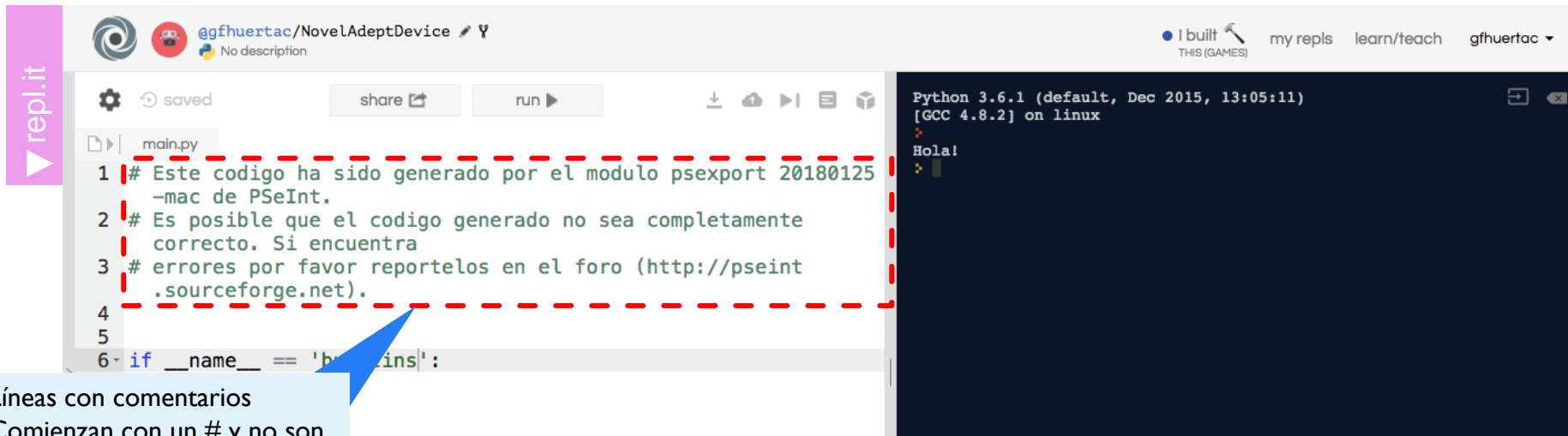
# Lección 01

▼ # (comentarios)

Cuando deseamos escribir alguna nota para recordar o para otra persona podemos utilizar comentarios.



Una línea de comentarios en Python comienza con el símbolo # y todo lo que está después de este no será tomado en cuenta por el intérprete e Python al momento de ejecutar el programa.



The screenshot shows a Python REPL interface. On the left, a code editor displays a file named `main.py` with the following content:

```
1 # Este código ha sido generado por el modulo psexport 20180125
 # -mac de PSeInt.
2 # Es posible que el código generado no sea completamente
 # correcto. Si encuentra
3 # errores por favor reportelos en el foro (http://pseint
 # .sourceforge.net).
4
5
6 if __name__ == '__main__':
```

A red dashed box highlights lines 1 through 4, which are comments. A blue arrow points from a text box below to this box. The terminal window on the right shows the output of the program:

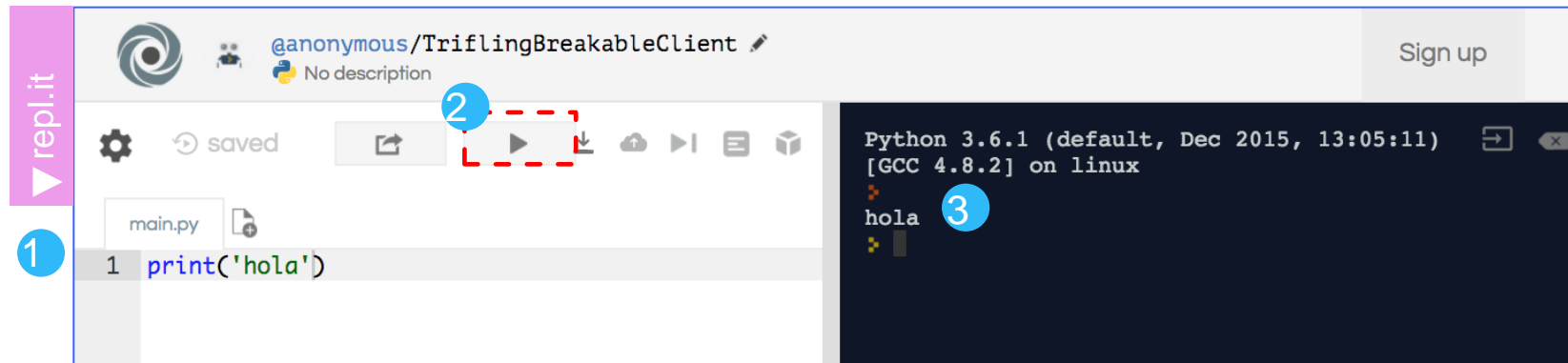
```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>
> Hola!
>
```

Líneas con comentarios  
Comienzan con un # y no son  
tomados en cuenta al  
momento de ejecutar el  
programa

# Lección 01

▼ print()

El comando `print()` permite desplegar el resultado de un variable o un texto por la terminal. Si escribimos texto, éste debe ir escrito entre comillas. Si escribimos una variable, éste despliega el valor almacenado en dicha variable.



El primer paso consiste en escribir uno o más instrucciones. El segundo paso es ejecutar el código, y el último paso es revisar los resultados en la terminal

Secuencia de pasos:

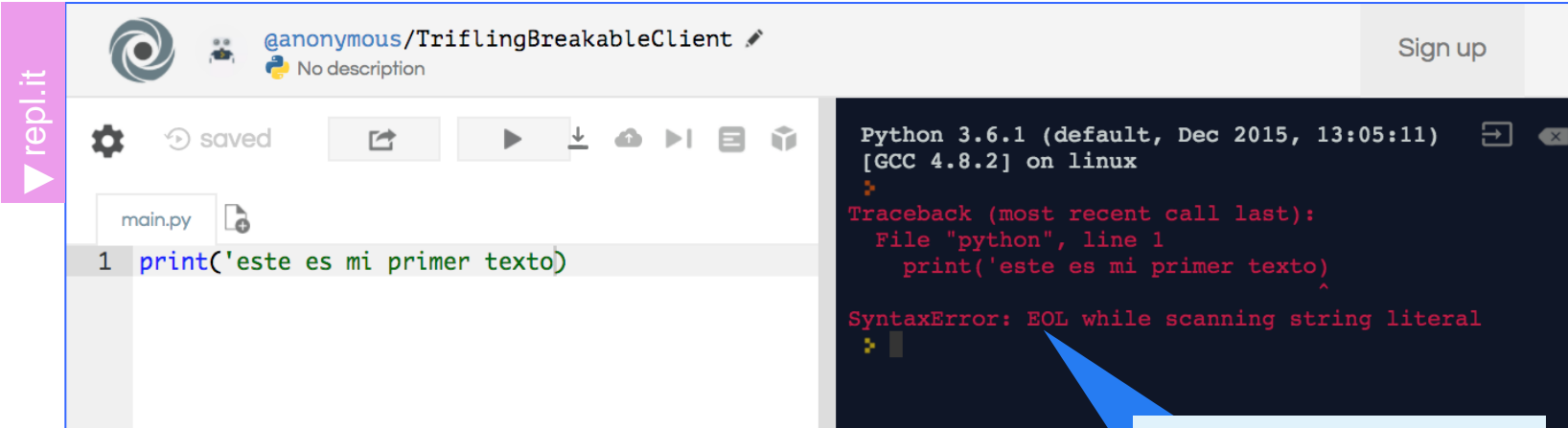




# Lección 01

▼ print()

Si escribimos texto, **éste debe ir escrito entre comillas**. En caso contrario, el computador arrojará un error ya que no sabrá donde termina la cadena de texto.



The screenshot shows a web-based Python REPL interface. The left sidebar has a 'repl.it' logo. The main area shows a file named 'main.py' with the following code: `1 print('este es mi primer texto)`. The right panel shows the output, which is a traceback error: `Python 3.6.1 (default, Dec 2015, 13:05:11) [GCC 4.8.2] on linux`, followed by `Traceback (most recent call last):`, `File "python", line 1`, `print('este es mi primer texto)`, and finally `SyntaxError: EOL while scanning string literal`. A blue arrow points from the error message to a text box on the right.

✗ `print('este es mi primer texto)`

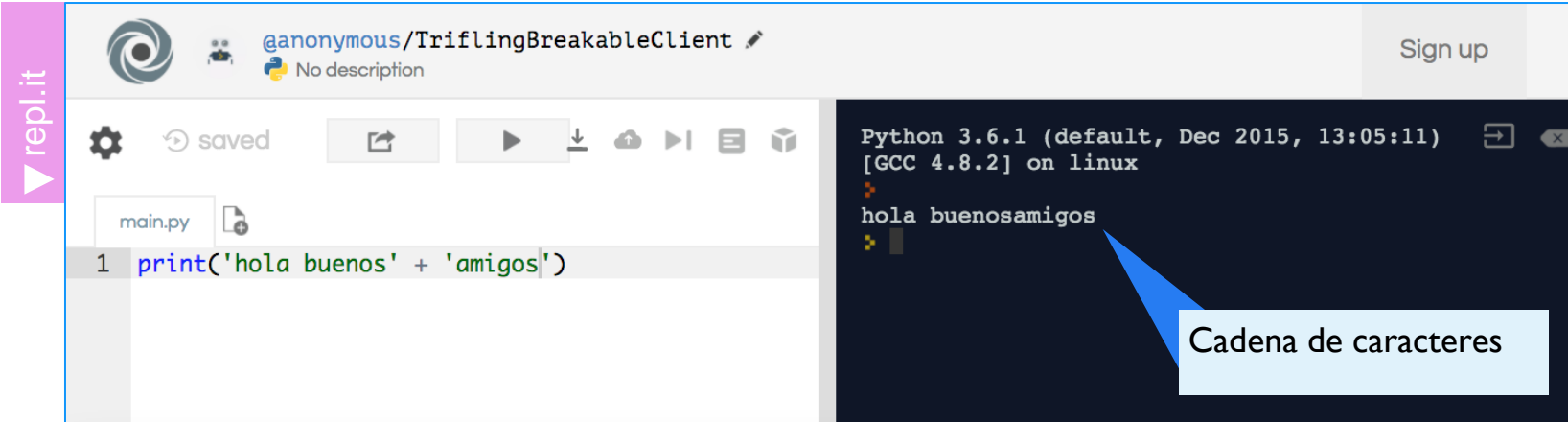
👍 `print('este es mi primer texto')`

Como se observa en la terminal, no se muestra el texto. Esto ocurre por que hemos omitido el símbolo ' al terminal el de texto.

# Lección 01

▼ print()

Para unir dos cadenas de texto, podemos emplear el símbolo + Es importante notar que une directamente las cadenas sin dejar un espacio en blanco entre éstas



The screenshot shows a web-based Python REPL interface. On the left, a file named 'main.py' contains the code `print('hola buenos' + 'amigos')`. On the right, the terminal output shows the Python environment (Python 3.6.1, GCC 4.8.2 on linux) and the execution result: `hola buenosamigos`. A blue callout box points to the output string with the text 'Cadena de caracteres'.

|        |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |
|--------|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|---|
| INPUT  | h | o | l | a |  | b | u | e | n | o | s | + | a | m | i | g | o | s |
| OUTPUT | h | o | l | a |  | b | u | e | n | o | s | a | m | i | g | o | s |   |

# Lección 01

▼ print()

Print es una **función** que recibe 1 o más parámetros y devuelve un resultado por pantalla, cada parámetro es separado por una coma, y tiene que ser de un solo tipo de valor, es decir texto o número, nunca ambos.



The screenshot shows a Repl.it interface with a Python file named 'main.py'. The code is:

```
1 print('hola la suma de 2','y 7 es',2+7)
2
```

Blue arrows point from the text labels below to the code: 'Cadena de caracteres' points to the first string argument, and 'Números enteros' points to the expression `2+7`.

The output terminal shows:

```
Python 3.7.4 (default, Jul 9 2019, 00:06:43)
[GCC 6.3.0 20170516] on linux
hola la suma de 2 y 7 es 9
>
```

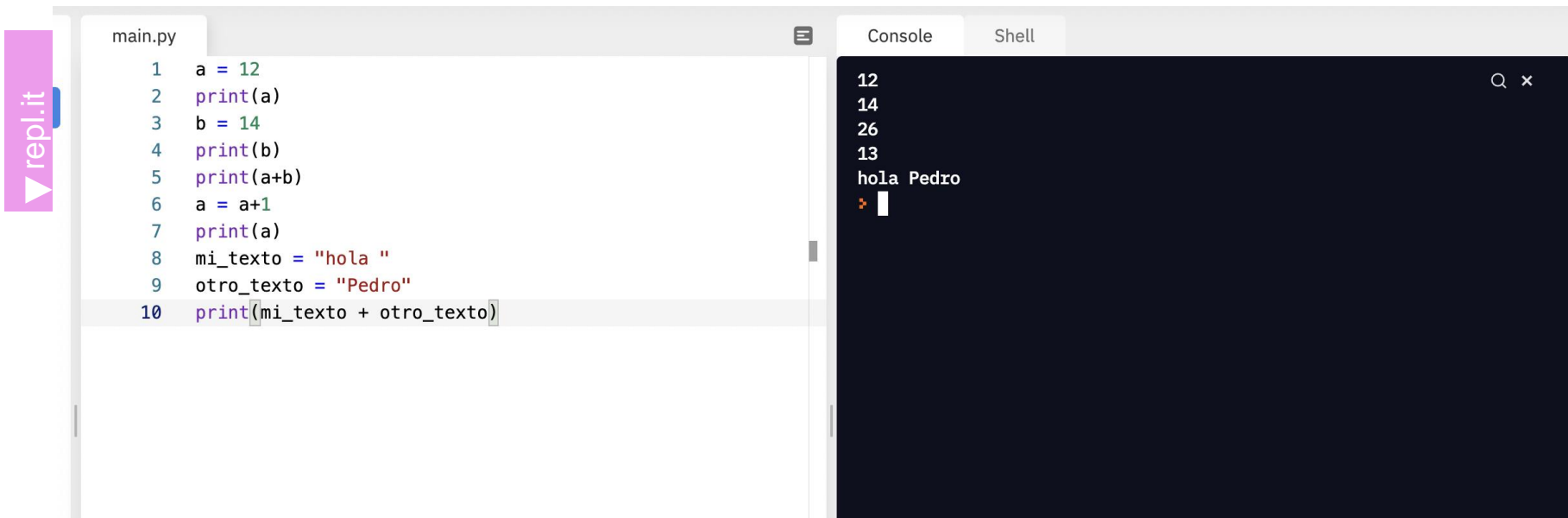
Labels at the bottom of the image:

- Cadena de caracteres
- Números enteros

# Lección 01

## ▼ Variables

Una variable es un nombre que le damos a una dirección de memoria y que se puede ocupar en el programa para ser manejada de manera abstracta ( sin saber cual es el valor que está guardado adentro)



The screenshot shows a code editor with a file named `main.py`. The code in the editor is as follows:

```
1 a = 12
2 print(a)
3 b = 14
4 print(b)
5 print(a+b)
6 a = a+1
7 print(a)
8 mi_texto = "hola "
9 otro_texto = "Pedro"
10 print(mi_texto + otro_texto)
```

To the left of the code editor is a vertical pink bar with the text `repl.it` and a downward arrow. To the right of the code editor is a console window with the following output:

```
12
14
26
13
hola Pedro
>
```

# Ejemplo de algoritmo

# Nota final del ramo

Si recordamos el syllabus, la nota final del ramo está dada por la siguiente fórmula:

$$\mathbf{NF=0.2*(P1+P2)+0.25*P3+0.1*T+0.1*P+0.15*C}$$

Donde:

- P1 es la nota de la prueba 1
- P2 es la nota de la prueba 2
- P3 es la nota de la prueba 3
- T es el promedio de las tareas
- P es el proyecto
- C es el promedio de los controles

Escriba un algoritmo para calcular la nota final de un alumno

# Proceso

- Preguntar la nota de la prueba 1
- Guardar la nota de la prueba 1
- Preguntar la nota de la prueba 2
- Guardar la nota de la prueba 2
- Preguntar la nota de la prueba 3
- Guardar la nota de la prueba 3
- Preguntar la nota promedio de las tareas
- Guardar la nota promedio de las tareas
- Preguntar la nota del proyecto
- Guardar la nota del proyecto
- Preguntar la nota promedio de los controles
- Guardar la nota promedio de los controles

# Proceso

$$NF=0.2*(P1+P2)+0.25*P3+0.1*T+0.1*P+0.15*C$$

- Asignar la nota final como la suma de la nota de la prueba 1 y la prueba 2
- Multiplicar por 0.2 la nota final
- Multiplicar la nota de la prueba 3 por 0.25 y sumar el resultado a la nota final
- Multiplicar la nota promedio de las tareas por 0.1 y sumar el resultado a la nota final
- Multiplicar la nota del proyecto por 0.1 y sumar el resultado a la nota final
- Multiplicar la nota promedio de controles por 0.15 y sumar el resultado a la nota final



# Proceso

- Mostrar la nota final al estudiante

**¿Cómo convertimos nuestro  
algoritmo a un programa?**

# Traducción de instrucciones

## **Algoritmo**

(Pseudo-código)

Preguntar la nota de la prueba 1

## **Código**

(Python)

```
input("Dime la nota de la prueba 1: ")
```

Input es una función existente en Python que lee una línea desde el teclado, la convierte en una cadena de texto (sin el salto de línea final) y la **devuelve**.

# ¿Qué significa que la función devuelve algo?

Significa que al usar la función el programa nos entrega algo que **podemos usar**.

## ¿Cómo?

Podemos ignorarlo, guardarlo en una variable o imprimirlo por pantalla.

# Traducción de instrucciones

## Algoritmo

(Pseudo-código)

Preguntar la nota de la prueba 1

Guardar la nota de la prueba 1

## Código

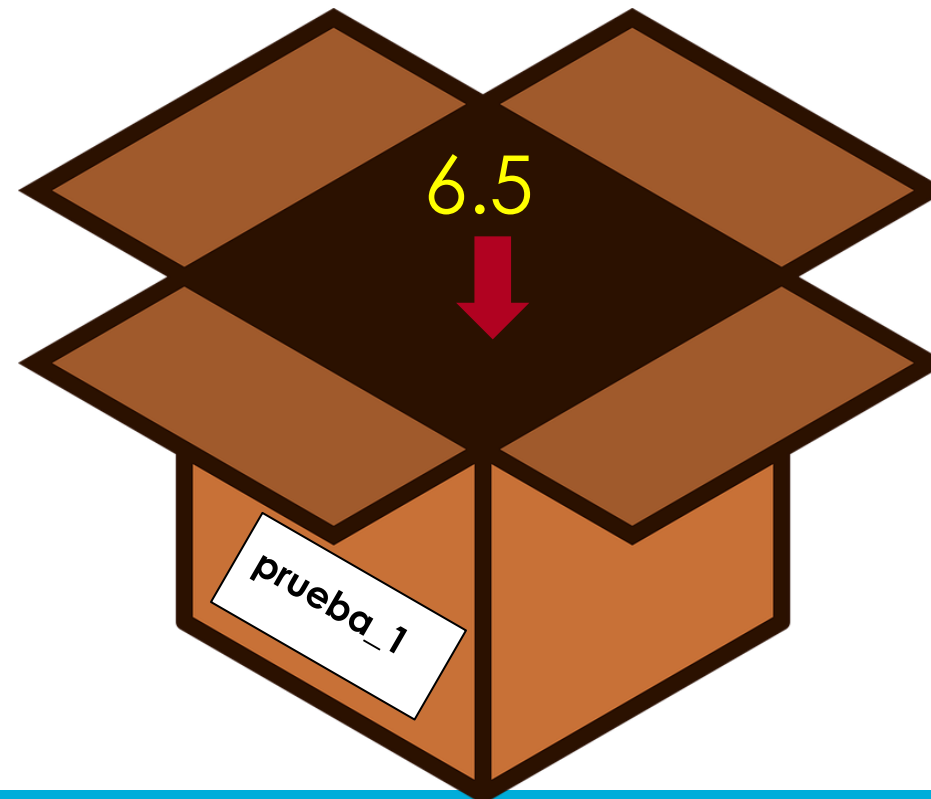
(Python)

```
prueba_1 = input("Dime
la nota de la prueba 1: ")
```

Una variable es una caja de memoria, donde podemos guardar valores.

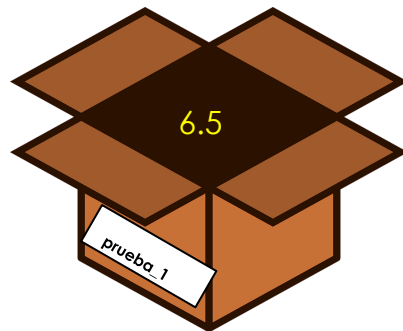
# Variables

Almacenar información en la memoria del computador



# Variables

Almacenar  
información en la  
memoria del  
computador



(Python)

```
prueba_1 = 6.5
```

El proceso de asignarle un valor a una variable se llama **asignación**. El valor puede ser algo fijo (como 6.5) o venir de una función (como `input()`) o un cálculo (lo veremos luego)

# Variables

Podemos realizar lo mismo con las pruebas 2, 3, tareas, controles y proyectos (input y guardar en variables).

Supongamos que guardamos todos los datos en variables llamadas prueba\_1, prueba\_2, prueba\_3, tareas, proyecto, controles. (¡Hazlo!)



## Continuemos

Cuando tenemos datos y queremos usarlos en una fórmula, no es necesario hacer el paso a paso.

Los lenguajes de programación como Python entienden las fórmulas y podemos usarlas directamente.

# Traducción de instrucciones

## Algoritmo

$$NF = 0.2 * (P1 + P2) + 0.25 * P3 + 0.1 * T + 0.1 * P + 0.15 * C$$

## Código

```
nota_final = 0.2*(prueba_1+prueba_2) + 0.25*prueba_3 + 0.1*tareas +
0.1*proyecto + 0.15*controles
```

En este caso hemos ASIGNADO un valor a una variable llamada nota\_final como el resultado de la fórmula (todo en una línea)

## Nota importante

input nos devuelve un TEXTO → no nos sirve para ocuparlo en una fórmula

Por lo tanto debemos **transformar** ese texto a lo que deseamos

# Funciones de transformación

Python provee de funciones de transformación incorporadas, para resolver este problema

Ejemplo:

```
prueba_1 = input("Dime la nota de la prueba 1: ")
prueba_1 = float(prueba_1)
```

Primero, pregunto por la nota de la prueba y la guardo... pero queda como texto. Por lo tanto luego la convierto a número decimal y la guardo nuevamente

# Transformando textos

## **Queremos**

- Un número entero
- Un número decimal
- Un valor de verdad

## **Usamos**

- `int()`
- `float()`
- `bool()`

# Traducción de instrucciones

## **Algoritmo**

(Pseudo-código)

Mostrar la nota final al  
estudiante

## **Código**

(Python)

**print**(nota\_final)

print es una función existente en Python que escribe algo en la pantalla para que la persona lo puede ver.

# ¿Cómo usaremos print?

Print se puede usar de muchas formas (vean ejemplos en <https://realpython.com/python-print/>)

Sin embargo, la forma más útil es usando texto con formato.

**¿Qué es eso?**



# Texto con formato

print imprime lo que recibe:

- Texto fijo: `print("Hola")`
- Variable: `print(nota_final)`

Pero, ¿qué pasa cuando queremos mezclar ambas cosas? Es aquí donde el texto con formato es útil.



## Ejemplo

```
nombre = "Claudia"
print(f"Hola {nombre}")
```

→ esto imprime **Hola Claudia** en pantalla

Lo que va entre comillas es el texto a imprimir en pantalla

f"Hola {nombre}"

La f denota que esto es  
un texto con formato

Si deseo imprimir una variable  
entonces coloco el nombre de  
la variable entre {}

¿Preguntas?

# Actividad 1

Crea un programa que pida todas las notas al alumno que tuvo en el semestre y calcule su nota final.

Las notas de los controles y tareas se deben ingresar por separado

# Actividad 2

Crea un programa que pida al usuario:

- un valor  $a$
- la raíz del cuadrado exacto más cercano a este valor

El programa debe calcular un valor aproximado a la raíz cuadrada de  $a$  ocupando solamente operadores de sumas , multiplicaciones y divisiones.

El programa debe escribir el resultado en pantalla en una frase

# Actividad 3

La estación meteorológica del observatorio es un producto fabricado en EEUU que lamentablemente ocupa el sistema de medida imperial. El display muestra la temperatura en grados Fahrenheit ( $^{\circ}\text{F}$ ) y la velocidad de viento en Millas por hora (mph)

Genere un programa que pida al usuario cual es la temperatura ( $^{\circ}\text{F}$ ) y la velocidad de viento en mph que se leen en el display.

El programa debe calcular la temperatura en grados Celcius ( $^{\circ}\text{C}$ ) y la velocidad de viento en metros por segundo (m/s) y mostrarlos en pantalla insertados en una frase.