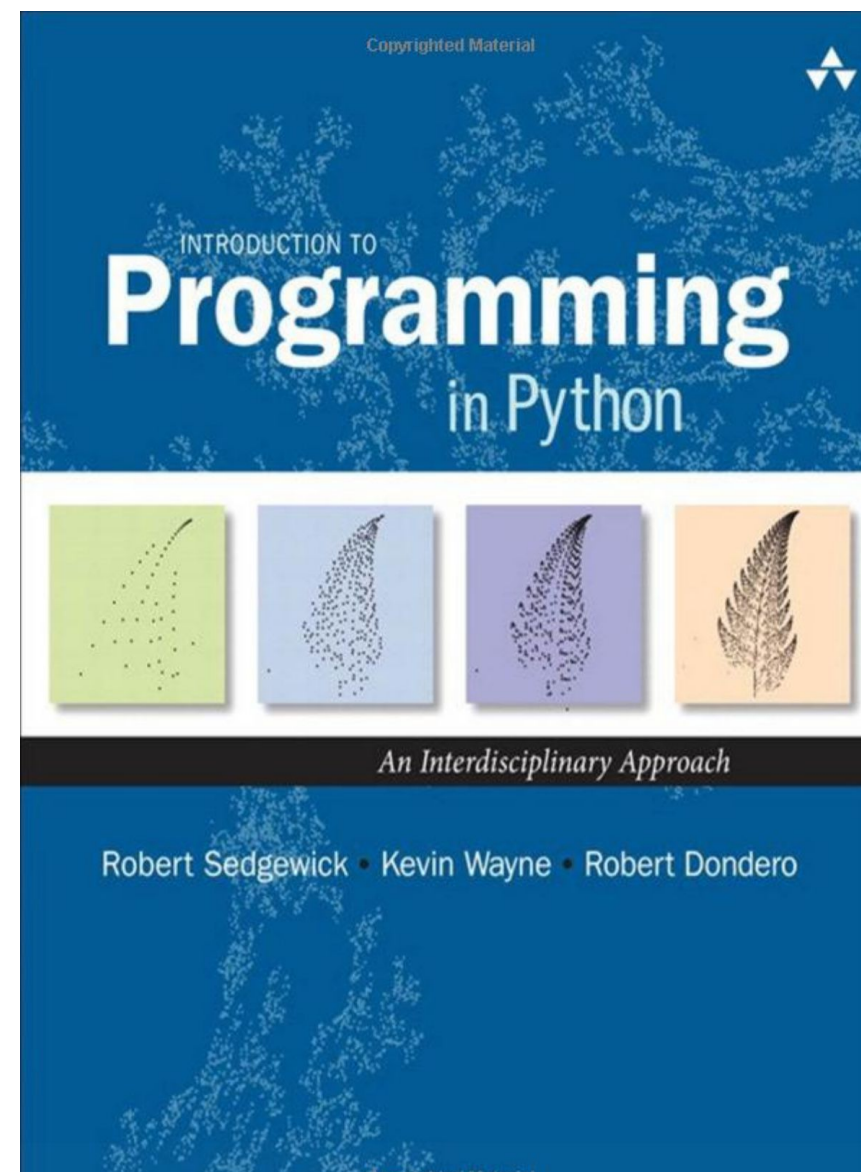


Taller de Programación

Tipos de Datos

Leonardo Causa
l.causa@udd.cl

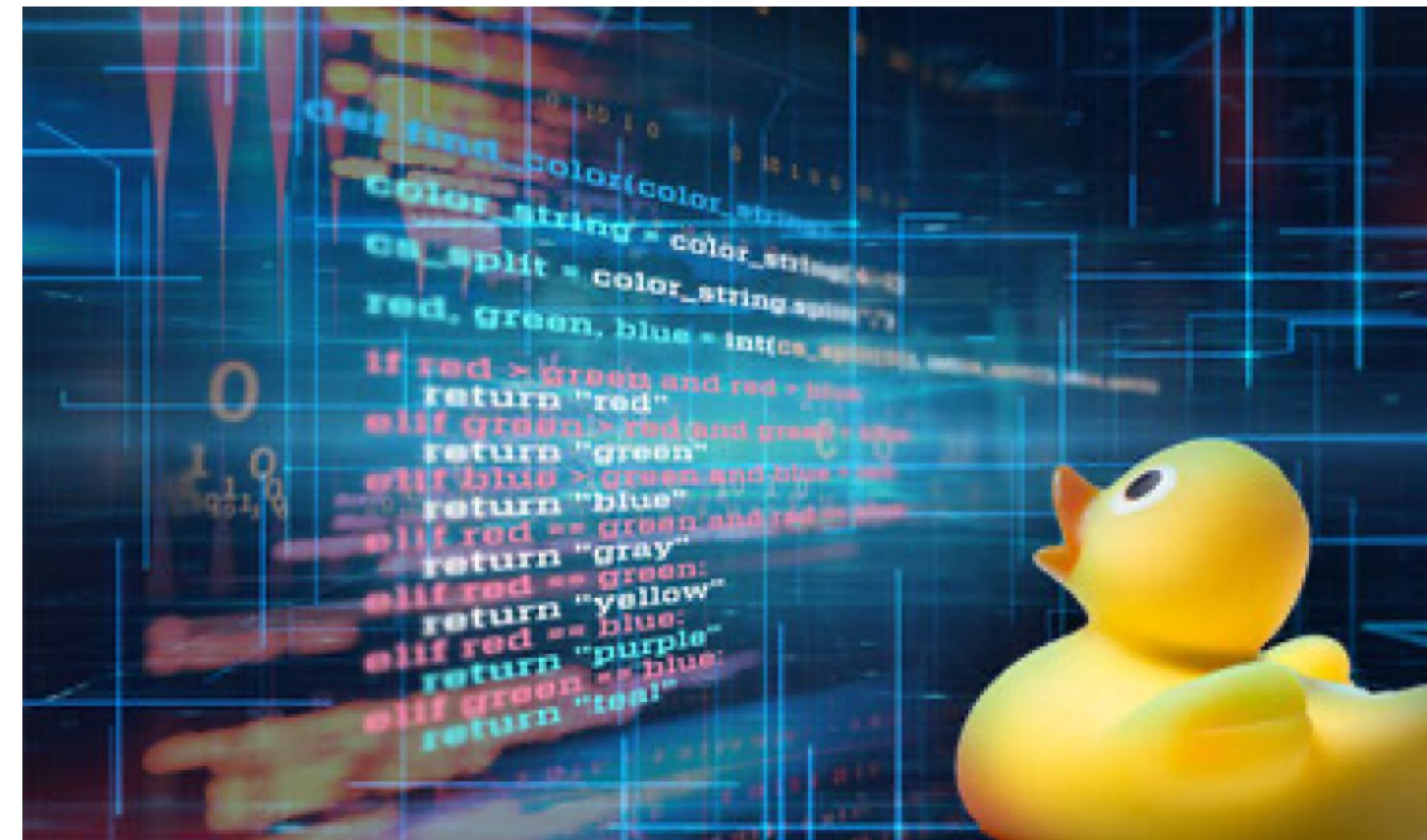


Basada en presentaciones oficiales de libro Introduction to Programming in Python (Sedgewick, Wayne, Dondero).

Disponible en <https://introcs.cs.princeton.edu/python>

Temario

- Anaconda (Spyder & Jupyter), Repl
- Shells
- Primer programa ('Hola Mundo')
- Sintaxis
- Variable y tipo de datos
- Operadores de aritmética básicos
- Entrada/salida de datos



Anaconda



Windows 

Python 3.8

64-Bit Graphical Installer (466 MB)


32-Bit Graphical Installer (397 MB)

MacOS 

Python 3.8

64-Bit Graphical Installer (462 MB)

64-Bit Command Line Installer
(454 MB)

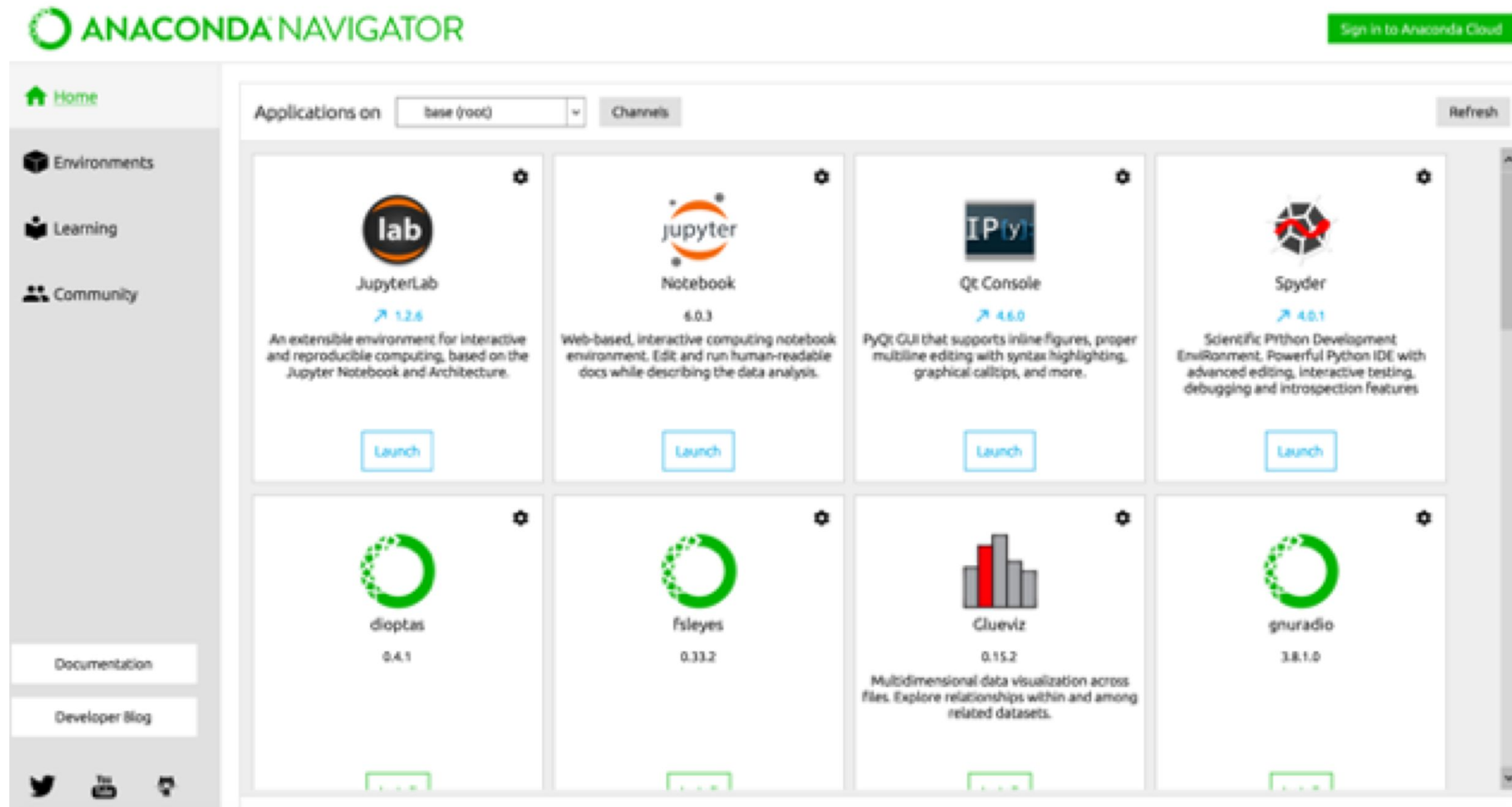
Linux 

Python 3.8

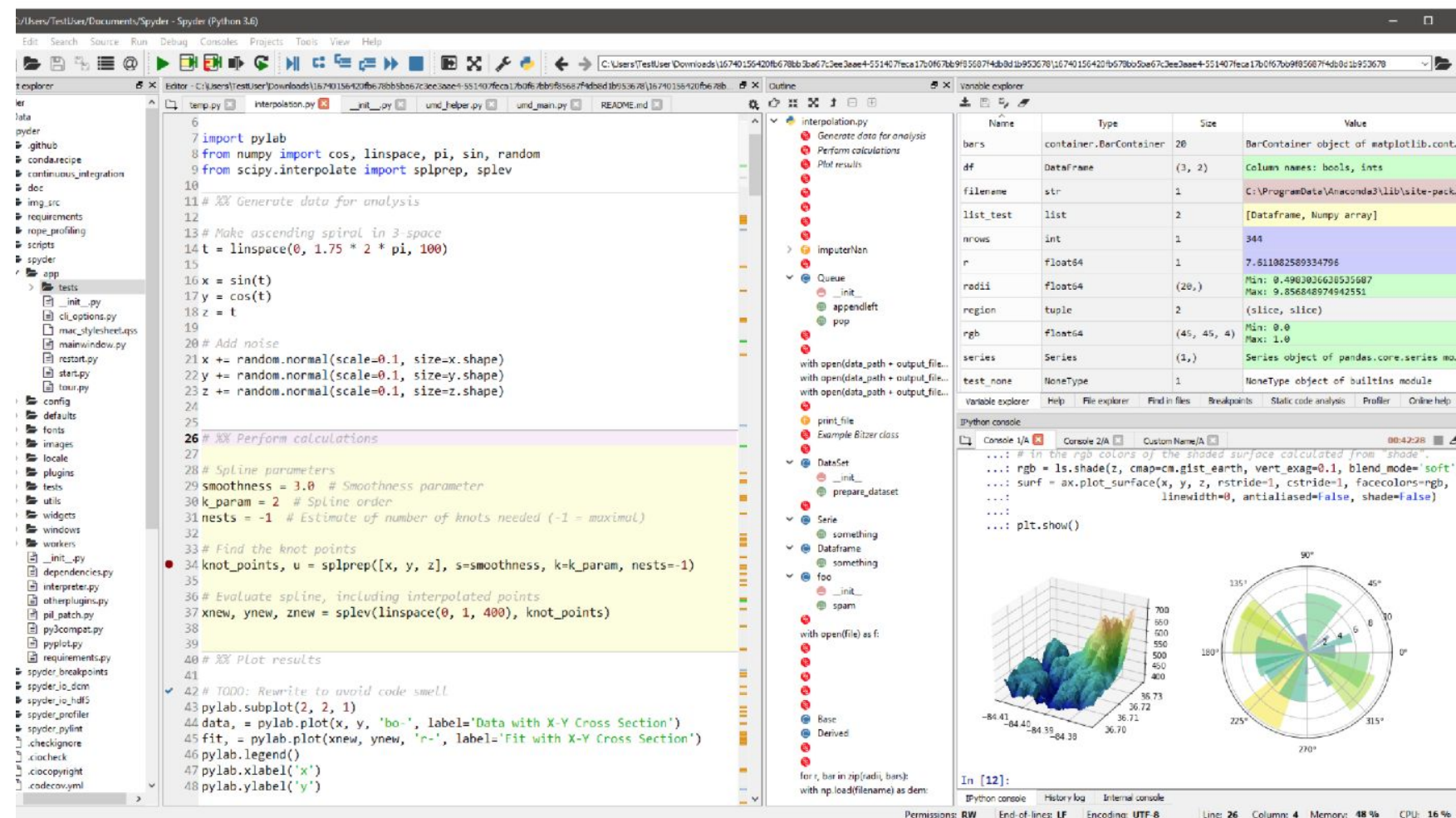
64-Bit (x86) Installer (550 MB)

64-Bit (Power8 and Power9)
Installer (290 MB)

Anaconda



Spyder & Python



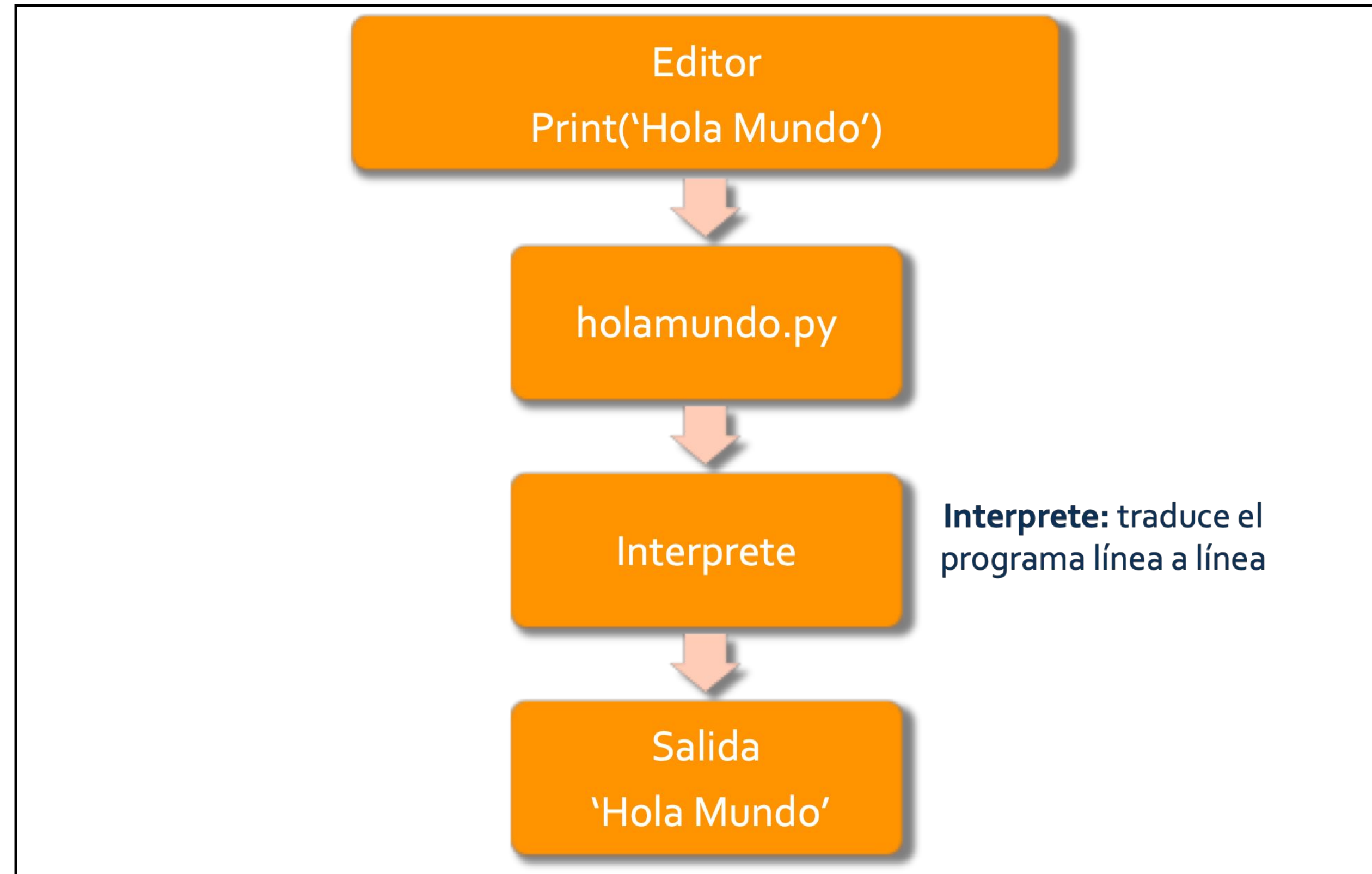
- Scientific Python Development Environment
- Entorno de desarrollo interactivo para el lenguaje Python



- Jupyter Notebook es un entorno informático interactivo para programar.

Hola mundo

```
# holamundo.py  
print( 'Hola  
mundo! ' )
```



```
2. bash  
$ python3 holamundo.py  
Hola mundo!  
$
```

Sintaxis

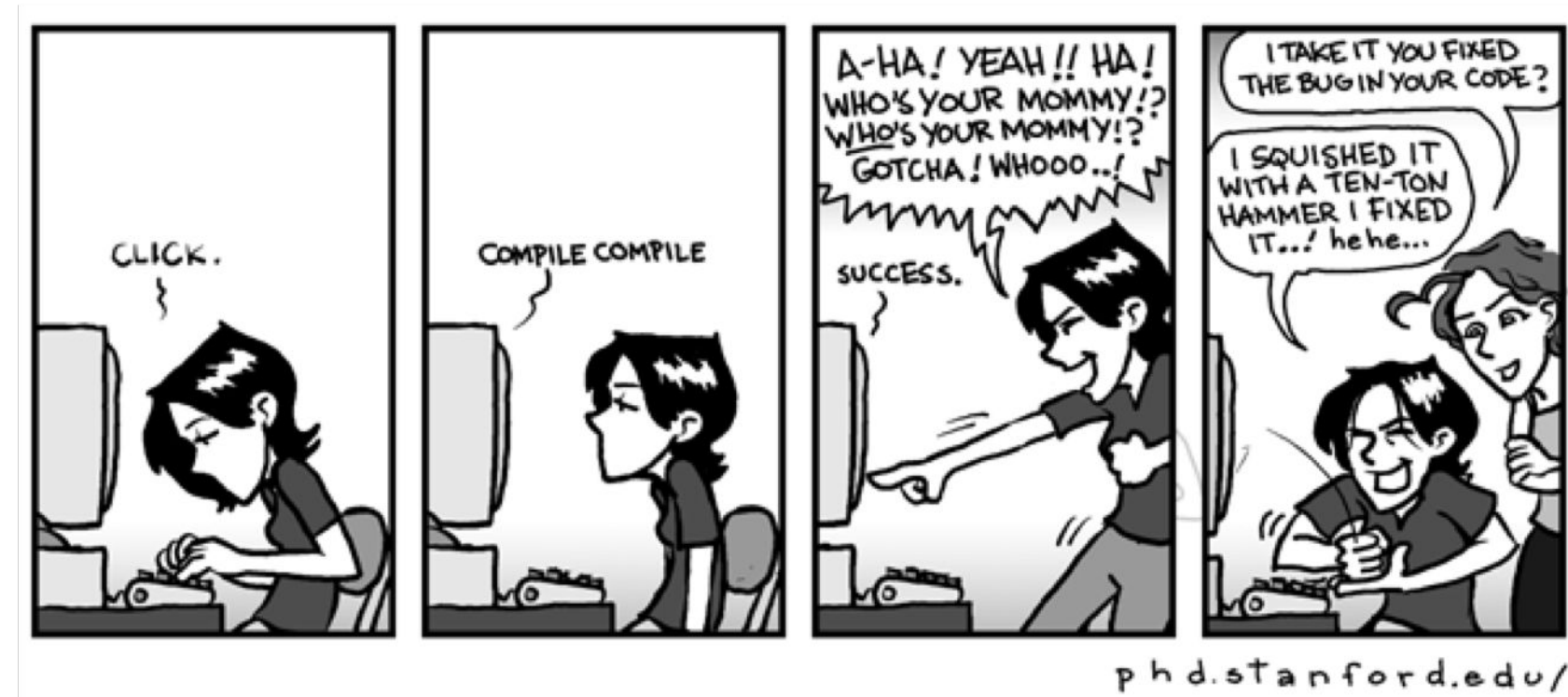
Conjunto de reglas que regulan el lenguaje

Español: "Juan perro gato" → **No es sintácticamente válido**

"Juan abraza al gato" → Sintácticamente válido

Leng. de programación: "3.2"5 → **No es sintácticamente válido**

3.2*5 → Sintácticamente válido



Variables

¿Qué es una variable?

Memoria que permite almacenar datos

Ejemplo

```
nombre = 'Cameron  
Howe'  
print(nombre)  
'Cameron Howe'
```

02_variables.py

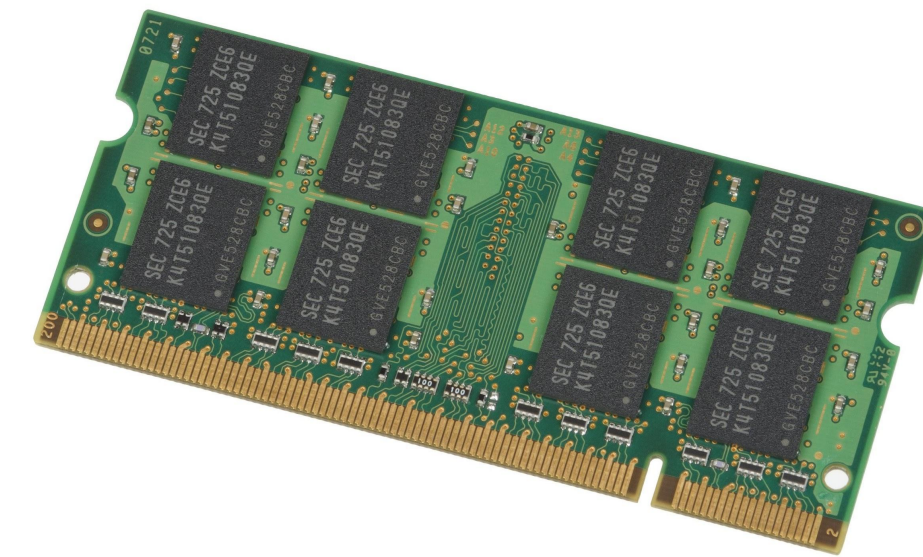


Cameron Howe, programadora protagonista de la serie Halt and Catch Fire.

Memoria

¿Qué entendemos por memoria?

Dispositivo electrónico que permite almacenar datos durante un intervalo de tiempo



1	3	4	4	6	7	0	1
---	---	---	---	---	---	---	---

Tipos de Datos

- `int`: representa números **enteros** (ej. 3)
- `float`: representa números **reales** (ej. 3.27)
- `bool`: representa valores **boolean** (**True** and **False**)
- `str`: representa valores **strings** (texto)
- `NoneType` – representa el valor **None**

Puedes usar `type()` para ver el tipo

Conversión de Tipos de Datos

Podemos convertir str en números usando:

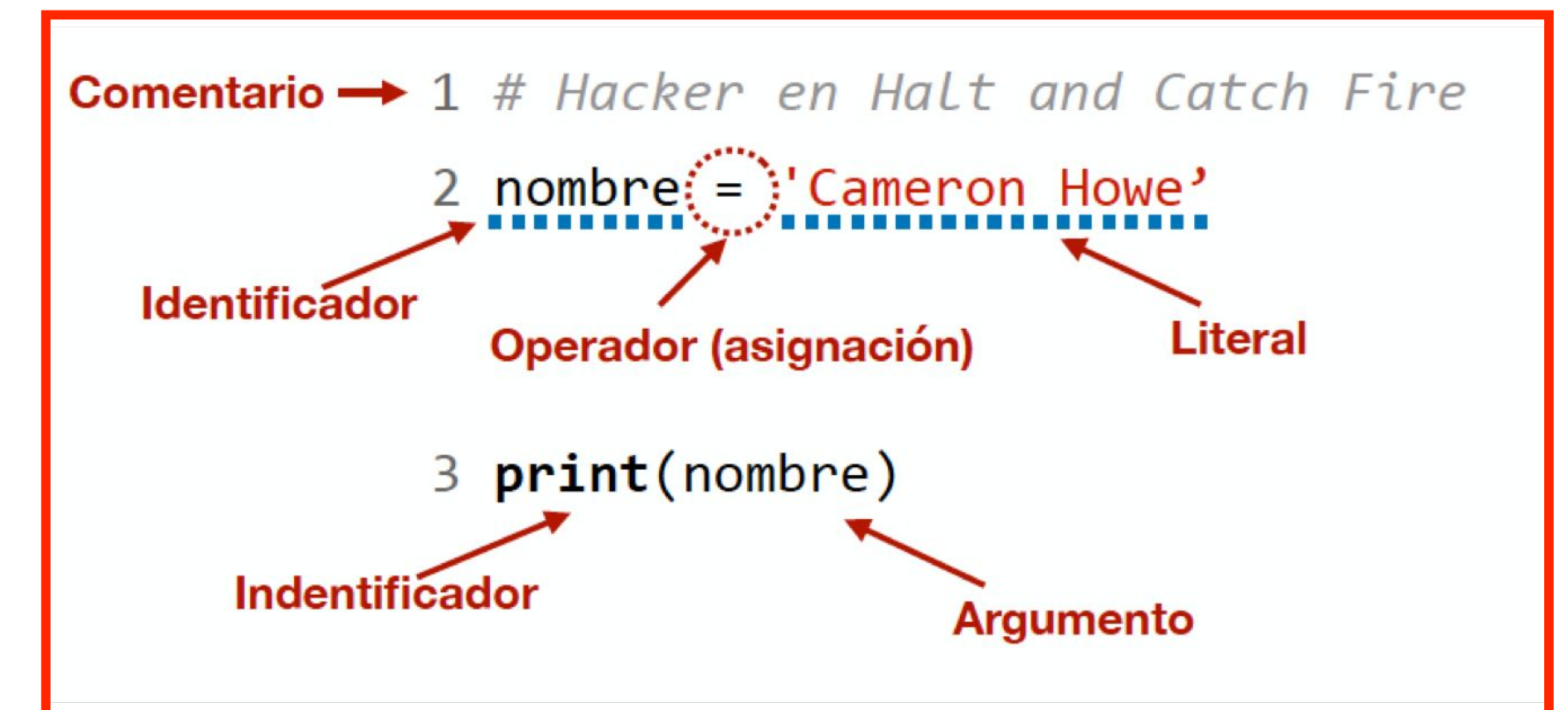
- **int(...)** para números enteros
- **float(...)** para números decimales

Y también podemos convertir números en texto usando **str(...)**

```
nombre = 'Olivia'
edad1 = '35'
edad2 = int(edad1)
print('Hola', nombre, 'veo que tienes',
      str(edad2), 'años')
decenas = edad2 // 10
print('Y al menos tienes', decenas,
      'décadas')
```


Sintaxis

- **Literal:** es una forma sencilla de escribir un valor.
 - **"hola"** es un texto (tipo de dato **str**)
 - **3.14** es un número **float**
- **Identificador:** nombre que se le puede dar una variable o función.
 - **nombre**, **print**
- **Variables:** es una referencia a una parte de la memoria que representa algún dato. La usamos para guardar resultados parciales a medida que la computación se realiza.
- **Expresiones:** combinación de variables, literales e identificadores.
- **Traza:** evolución de variables a medida que se ejecuta un programa.



Identificadores

- Deben comenzar con una letra o con un guión bajo
- Pueden contener dígitos, y guión bajo
 - x, nombre_apellido, y55, _holahola
- Pero no pueden comenzar con números o contener caracteres de operacion:
 - 12, 1x, -55, 33mineros, micro\$oft
- Tampoco pueden ser palabras reservadas:
 - Ejemplo: `int`, `if`, `return`, `float`...

Identificadores (Palabras Reservadas)

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	while
and	del	global	not	with
as	elif	if	or	yield
assert	else	import	pass	
break	except	in	raise	

Identificadores

- Deben ser informativos!
- Siglas son confusas: `LAX`, `CCP`, `myvar`
- No use nombres largos: `mi_variable_de_tipo_string`
- Recuerde que su código será leído por otra persona... incluido su “futuro yo”
- Buenos ejemplos:
 - `edad`, `suma_parcial`, `temperatura`, `process_type`, `current_time`

Operaciones Sobre Datos

Operador	Significado
+	Suma (o adición)
-	Resta (o sustracción)
*	Multiplicación
**	Exponente
%	Módulo (resto)
/	División
//	División entera

Texto	Números
+	+, -, /, *, %, //
(concatenar)	
+=	+=, -=, *=, etc ...
(agregar al final)	

Fuente: [https://es.wikibooks.org/wiki/Python/Generalidades/
Palabras reservadas, operadores y s%C3%ADmbolos del lenguaje](https://es.wikibooks.org/wiki/Python/Generalidades/Palabras_reservadas_operadores_y_s%C3%ADmbolos_del_lenguaje)

Operaciones Datos

Operaciones sobre
strings



```
1 primero = 'Leonardo'  
2 segundo = 'Causa'  
3 nombre = primero + ' ' + segundo  
4 print('Tu nombre es', nombre)
```



Leonardo Causa

Entrada y Salida de Datos

Funciones	de entrada/salida:
<code>input(...)</code>	recibe datos por teclado
<code>print(...)</code>	imprime datos en la pantalla

```
nombre = input('Indica tu  
nombre:')  
print('Tu nombre es', nombre)
```

La función input por defecto importa variables de tipo **str** (texto)

Entrada y Salida de Datos

Entrada de números

```
nombre = input('Indica tu nombre: ')\nedad = int(input('Indica tu edad: '))\nprint('Hola', nombre, 'veo que tienes',\n      edad, 'años')\ndecenas = edad // 10\nprint('Y al menos tienes', decenas,\n      'decadas')
```

P: ¿Qué sucede si eliminamos el comando `int` en la línea 2 de la entrada de números?

R: Error en línea 4, no puedes hacer una división con un texto!



Diana Carrier
@artemis_134

Programming is like magic. You write very specific instructions in arcane languages to invoke commands, and if you get it even a little bit wrong you risk unleashing demons and destroying everything.

07_enteros_in_out.py

Entrada y Salida de Datos

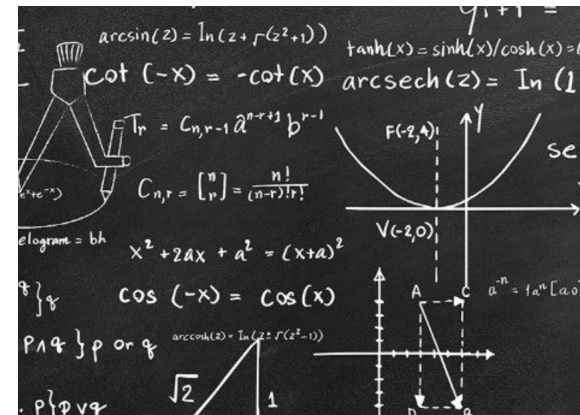
```
amigos = input('¿Cuántos amigos tienes en facebook?')
comun  = input('¿Cuántos amigos tienes en comun con tu mejor amigo?')

num_amigos = int(amigos)
num_comun  = int(comun)
porcentaje = num_comun/num_amigos*100

print('Wow, tienes', porcentaje, '% de amigos en común con tu mejor amigo!')

print('tipo variable amigos:', type(amigos))
print('tipo variable num_amigos:', type(num_amigos))
```


Entrada y Salida de Datos



math: modulo con metodos matemáticos que importamos con el comando **import**



```
1 # En la siguiente linea importamos la funcion raiz
cuadrada.
2 # Revisa otras funciones matemáticas
3 # en https://docs.python.org/3/library/math.html
4 from math import sqrt
5
6 n = float(input('Ingrese un numero decimal:
'))
7 print('n = ', n)
8 print('5 por n = ', 5*n)
9 print('n dividido dos = ', n/2)
10 print('raiz cuadrada de n = ', sqrt(n))
```

Revisar

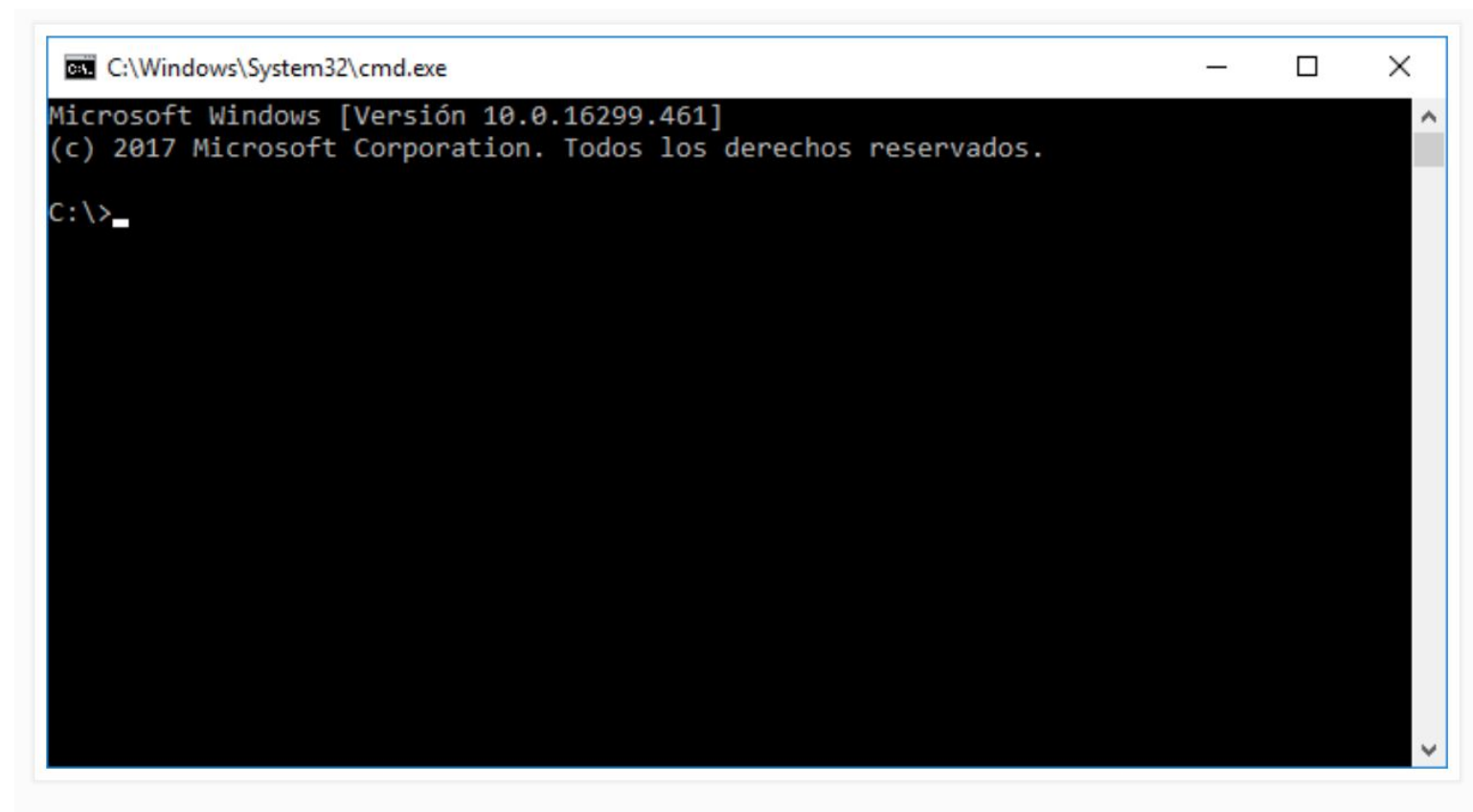
<https://docs.python.org/2/library/math.html>

Uso de la Consola

- **Windows**

>>**cd**: Cambia a otro directorio

>>**dir**: Muestra una lista de archivos y subdirectorios en un directorio

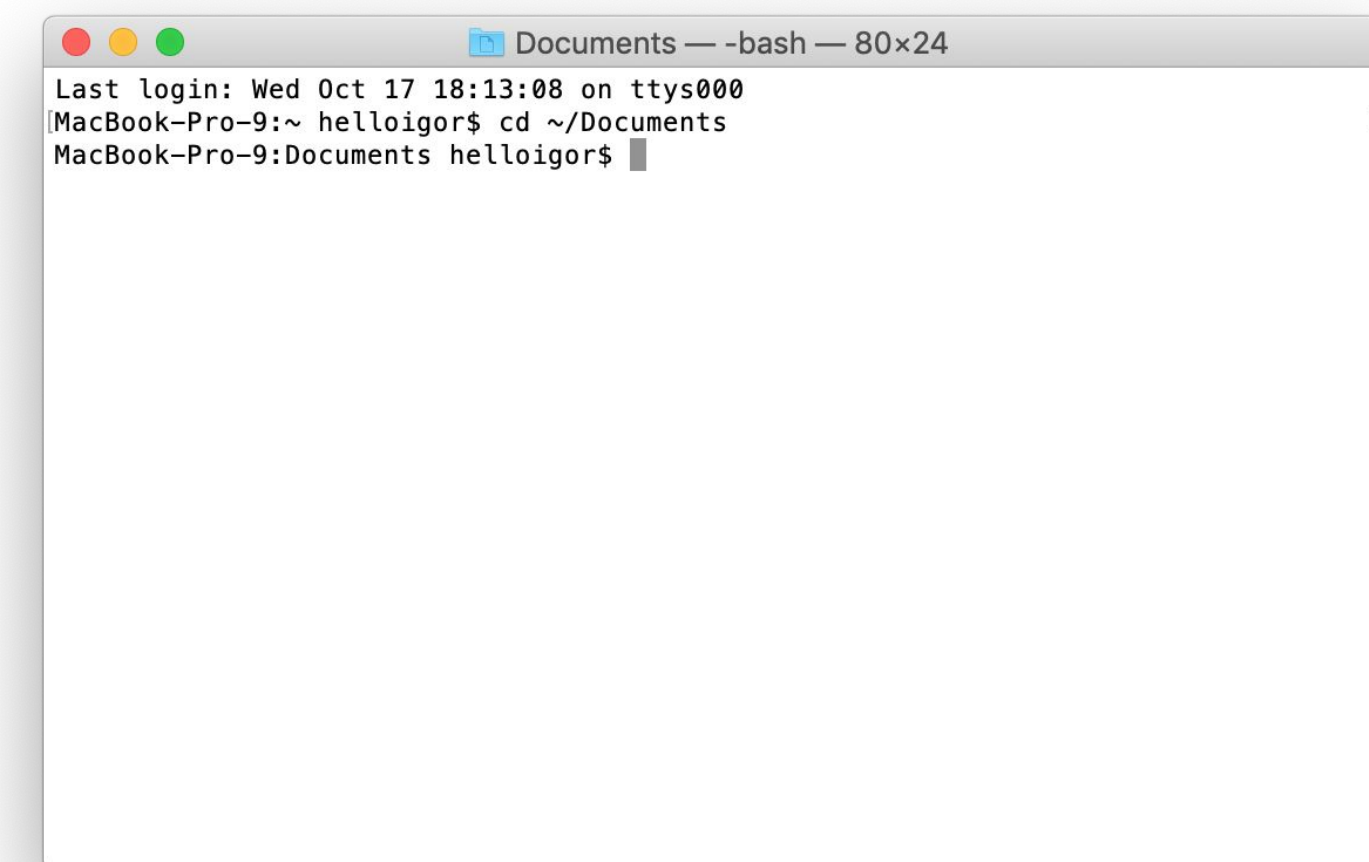


cmd (Windows)

- **Mac**

>>**cd**: Cambia a otro directorio

>>**ls**: Muestra una lista de archivos y subdirectorios en un directorio



Terminal (Mac)

Actividades

1. Crea un programa que calcule la solución de una ecuación de segundo grado de la forma:

- Asume que el usuario entrega el valor de a, b y c.
- Recuerda que la raíz se puede calcular con:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

2. Utilizando el archivo 10_argumento.py

- Explóralo y ejecútalo desde la consola
- Modifícalo de modo que imprima tres nombres

Resumen

Tipos de datos

- **int**: números enteros. Ej: -1,10, 12121
- **float**: números decimales. Ej: 1.1, 23.99, 2.0
- **str**: texto. Ej: 'Ingeniería Civil', '😊'
- **bool**: binario: Ej: True, False

← La mayúscula es obligatoria!

Funciones

- **input(. . .)**: captura de datos desde el teclado
- **print(. . .)**: imprimir en pantalla
- **type(. . .)**: verificar el tipo de dato de una variable o literal

Funciones matemáticas

```
>>> import math
>>> dir(math)
['__doc__', '__file__', '__loader__', '__name__', '__package__', '__spec__', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'copysign', 'cos', 'cosh', 'degrees', 'e', 'erf', 'erfc', 'exp', 'expm1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma', 'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'log2', 'modf', 'nan', 'pi', 'pow', 'radians', 'remainder', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'tau', 'trunc']
```

Conceptos

- **concatenar**: unir dos variables de tipo str
- **entrada/salida**: ingresar y extraer datos
- **traza**: seguimiento de variables según se ejecuta el programa

Sintáxis

```
var_texto = 'literal con text'
var_entero = -112
var_float = 3.1416

#concatenar
t1 = 'mi casa'
t2 = 'es bonita'
texto = t1 + ' ' + t2 # 'mi casa es bonita'

#operaciones aritméticas
suma = 1+2 # -, *, /
div = 1 // 2 # division entera
mod = 1 % 2 # resto
```

Más info en <https://docs.python.org/3/library/math.html>