# Ayudantía de Programación

# Introducción a Python

Gonzalo Fernández

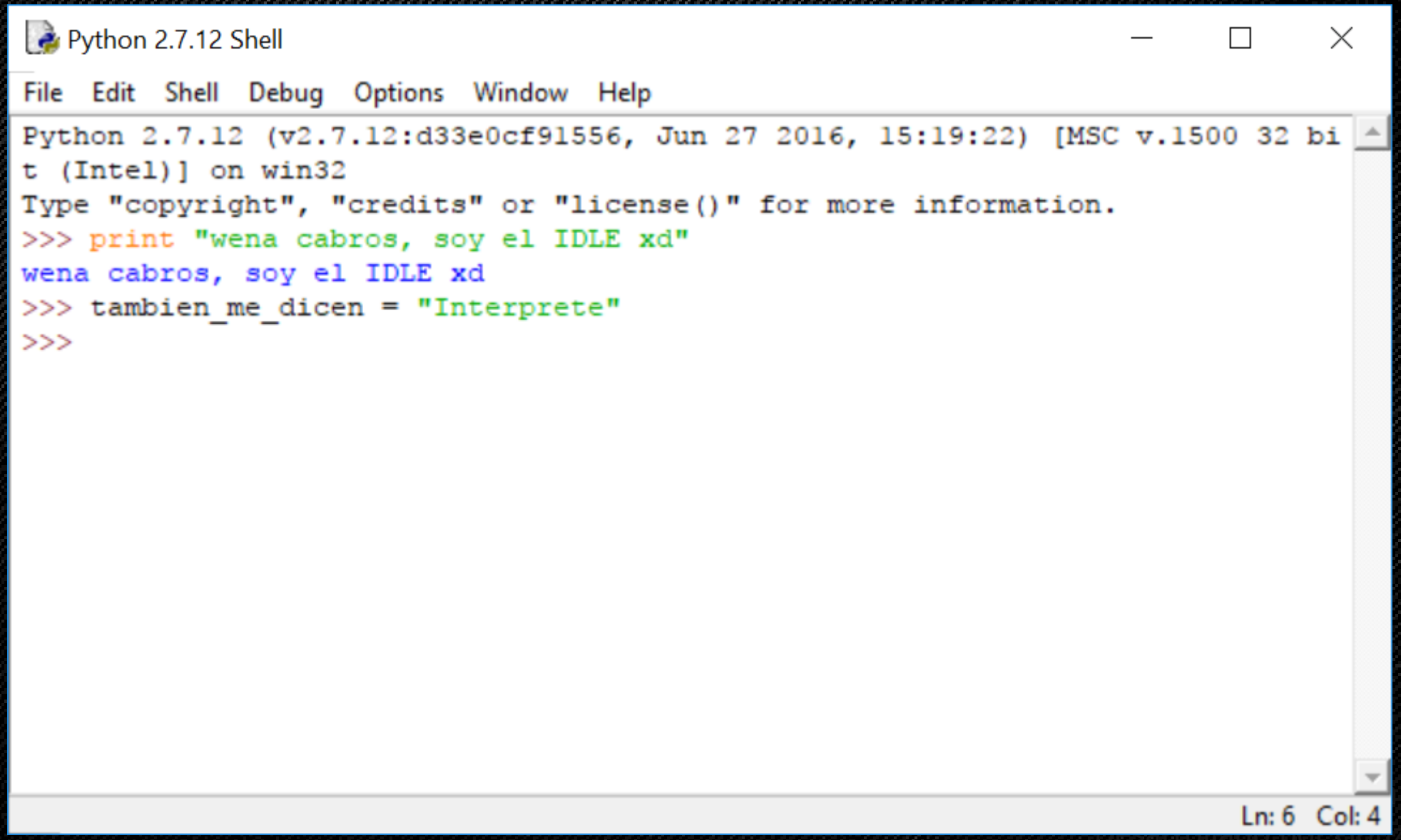
IWI-131 24-09-2018

# Temas de la Ayudantía de Hoy

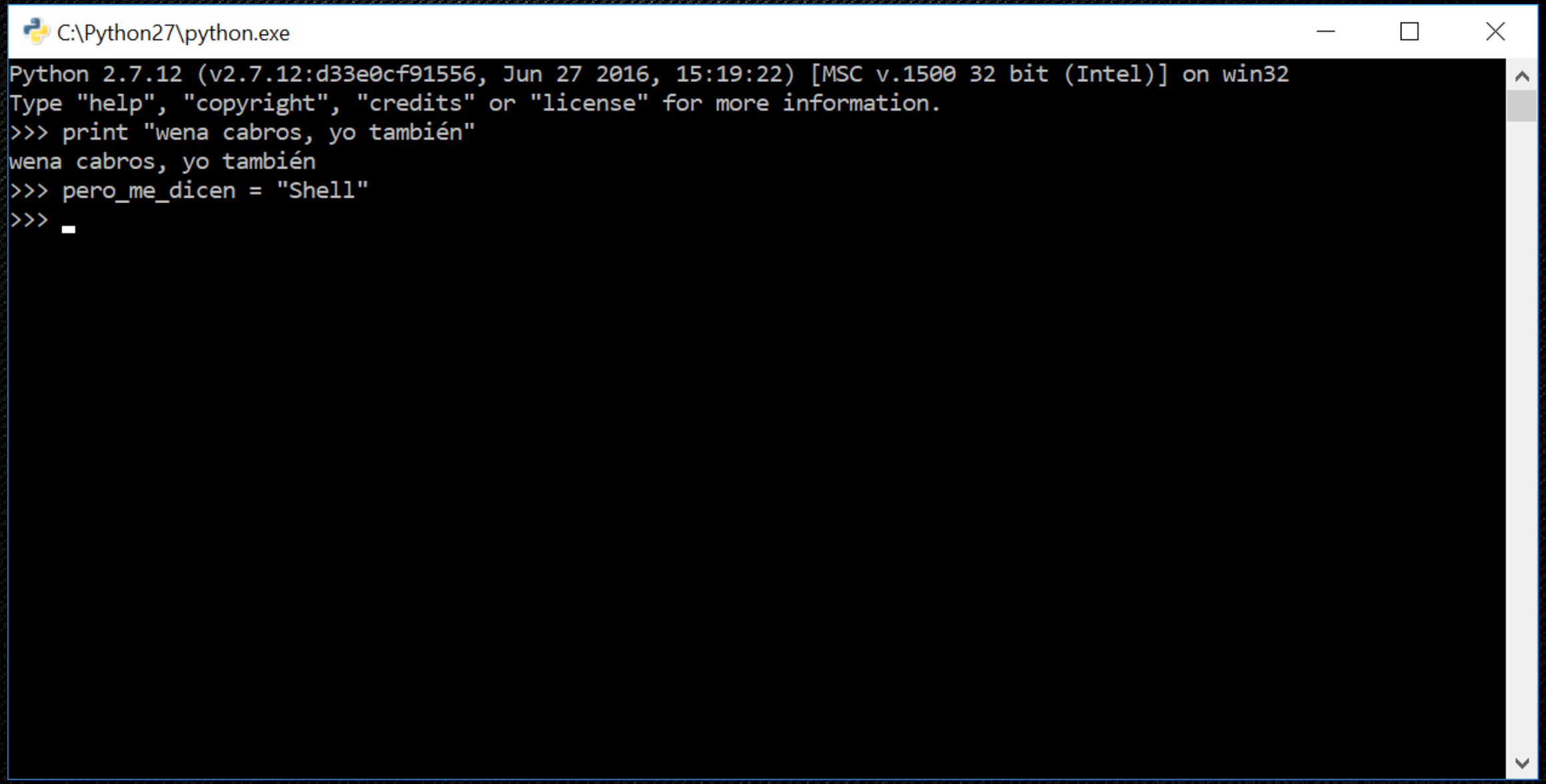
- Conociendo el intérprete de Python
- Tipos de datos y operaciones
- Ejemplos de sentencias básicas
- Entrada y salida de datos
- Ejercicios

# Dudas

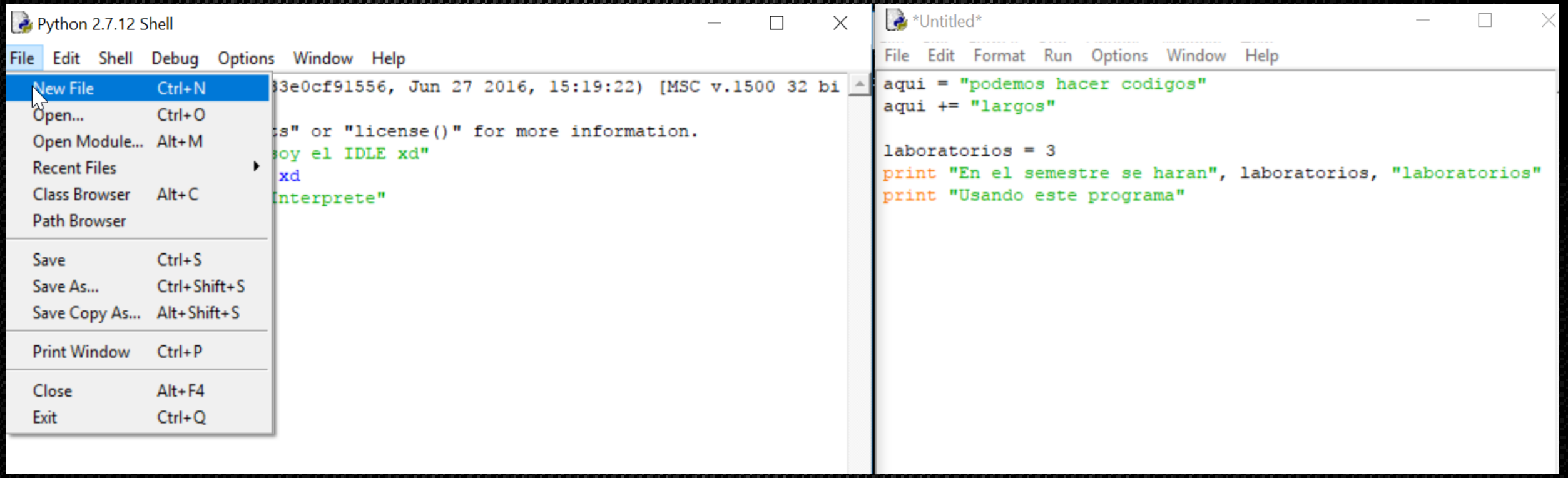
# Conociendo el Intérprete de Python



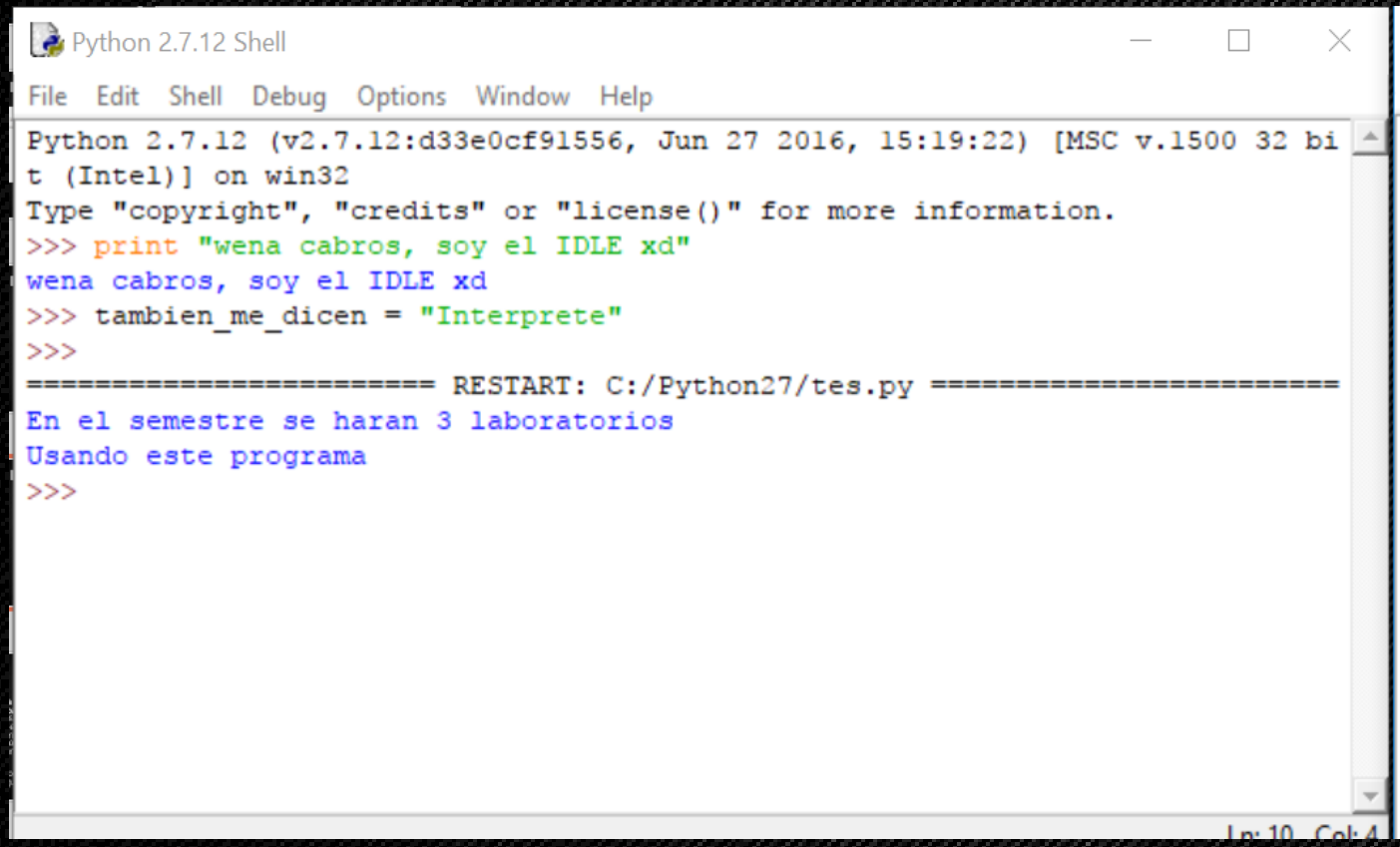
# Conociendo el Intérprete de Python

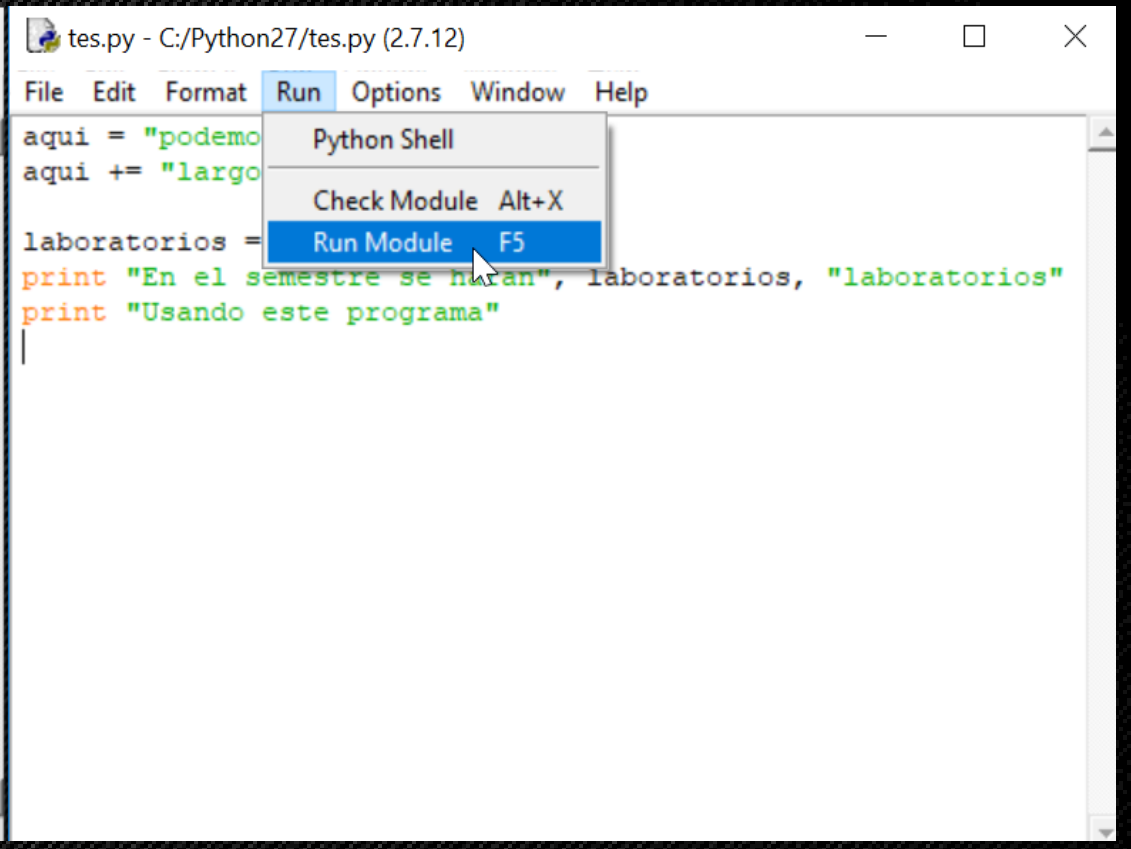


# Conociendo el Intérprete de Python



# Conociendo el Intérprete de Python



*REPL* [python] - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

asdasd.py

```
1
2  a = int(raw_input("n1: "))
3  b = int(raw_input("n2: "))
4
5  if a > b:
6      print "El numero mayor es:", a
7  else:
8      print "El numero mayor es:", b
9
10
```

*REPL* [python]

```
n1: 1
n2: 6
El numero mayor es: 6

***Repl Closed***
```

Line 6, Column 1 UTF-8 Tab Size: 4 Python

# Tipos de Datos y Operadores

En Python existen **variables**, estas tienen un nombre, un valor y un tipo determinados, donde los 2 últimos pueden variar a lo largo del programa.

```
entero = 5
flotante = 5.5555
string = “Hola Mundirijillo”
AlumnoAprueba = False
```

# Tipos de Datos y Operadores

Los tipos de datos ofrecidos por Python son:

- **Enteros**: Números naturales
- **Flotantes**: Números reales
- **Strings**: Texto en general (siempre van entre "comillas")
- **Booleanos**: True / False

# Tipos de Datos y Operadores

Python trae consigo un conjunto de operadores para poder trabajar en conjunto con sus variables.

| Suma | Asignación | Multiplicación | División |
|---|---|---|---|
| + | = | * | / |
| Resta | Módulo | Potencias | Precedencia |
| - | % | ** | ( ) |

# Ejemplos de Sentencias Básicas

tes.py - C:/Python27/tes.py (2.7.1...

File Edit Format Run Options Window Help

```
A = 40
B = 4
C = A + B
print "A + B =", C
C = A - B
print "A - B =", C
C = A * B
print "A * B =", C
C = A / B
print "A / B =", C
C = A ** B
print "A ** B =", C
C = A % B
print "A % B =", C
D = ((A*B)+(B*C))*A
print "((A*B)+(B*C))*A =", D
D = A*B+B*C*A
print "A*B+B*C*A =", D
```

Python 2.7.12 Shell

File Edit Shell Debug Options Window Help

```
Python 2.7.12 (v2.7.12:d33e0cf91556, Jun 27 2016, 15:19:22) [MSC
Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Python27/tes.py ==========
A + B = 44
A - B = 36
A * B = 160
A / B = 10
A ** B = 2560000
A % B = 0
((A*B)+(B*C))*A = 6400
A*B+B*C*A = 160
>>>
```

tes.py - C:/Python27/tes.py (2.7.1...

File Edit Format Run Options Window Help

```
A = 5
print A
A += 5
print A
A -= 20
print A
```

```
((A*B)+(B*C))*A = 6400
A*B+B*C*A = 160
>>>
========== RESTART: C:/Python27/tes.py ========
5
10
-10
>>>
```

tes.py - C:/Python27/tes.py (2.7.1...

File Edit Format Run Options Window Help

```
malditos = "ohayou"
otakus = "gozaimasu"
print malditos, otakus

primerTexto = "Parte 1"
segundoTexto = "Parte 2"
tercerTexto = primerTexto + segundoTexto
print tercerTexto

primerTexto += segundoTexto
print primerTexto
```

```
-10
>>>
========== RESTART: C:/Python27/tes.py =
ohayou gozaimasu
Parte 1Parte 2
Parte 1Parte 2
>>>
```

# Entrada y Salida de Datos

Una de las partes más importantes de la mayoría de programas es la forma en que éste recibirá y devolverá datos.

En Python, existen 2 funciones que se encargan de resolver este tema.

| Entrada | Salida |
| --- | --- |
| `raw_input` | `print` |

# Entrada y Salida de Datos

tes.py - C:/Python27/tes.py (2.7.12)

File Edit Format Run Options Window Help

```
entrada = raw_input("Ingresa algo: ")
print entrada
```

```
TypeError: cannot concatenate 'str' and 'int' objects
>>>
========================== RESTART: C:/Python27/tes.py ==
Ingresa algo: algo
algo
>>>
```

tes.py - C:/Python27/tes.py ...

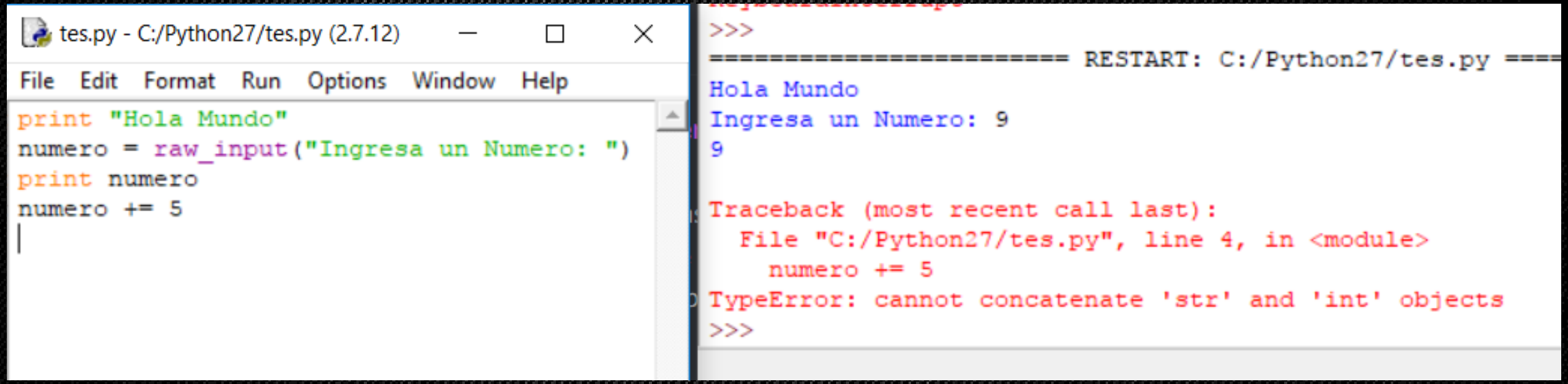
File Edit Format Run Options Window Help

```
p1 = raw_input()
print p1
p2 = raw_input("holi ")
print p2
p3 = raw_input()
print p3

print p3, p2, p1
print p3+p2+p1
```

```
>>>
========================== RESTART: C:/Python27/tes.py ==
nanda
nanda
holi kore
kore
wa
wa
wa kore nanda
wakorenanda
>>>
```

# Entrada y Salida de Datos



# Entrada y Salida de Datos

tes.py - C:/Python27/tes.py (2.7.12)

File Edit Format Run Options Window Help

```
numero = raw_input("Ingresa un Numero: ")
numero = int(numero)
numero += 5
print numero

# Tambien es valido lo siguiente:

numero = int(raw_input("Ingresa un Numero: "
numero += 5
print numero
```

```
>>>
========================== R
Ingresa un Numero: 5
10
Ingresa un Numero: 5
10
>>>
>>>


>>>

>>>
```

# Entrada y Salida de Datos

tes.py - C:/Python27/tes.py (2.7.12)

File Edit Format Run Options Window Help

```
print 6 / 8, 6.0 / 8.0

n = 5
print 3 / n, 3 / float(n)

n = 5.0
print 3 / n, 3.0 / n

numero = float(raw_input("Ingresa un Numero: "))
print 3 / numero, 3.0 / numero
```

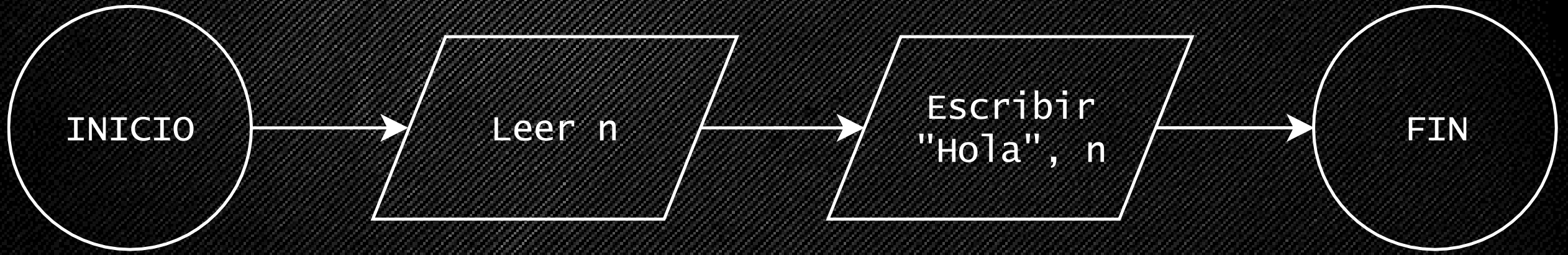
```
>>>
========== RESTART: C:/Python27/tes.py ==========
Ingresa un Numero: 123.456

Traceback (most recent call last):
  File "C:/Python27/tes.py", line 2, in <module>
    print int(numero), float(numero)
ValueError: invalid literal for int() with base 10: '123.456'
>>>
========== RESTART: C:/Python27/tes.py ==========
0 0.75
0 0.6
0.6 0.6
Ingresa un Numero: 123.456
0.024300155521 0.024300155521
>>>
```

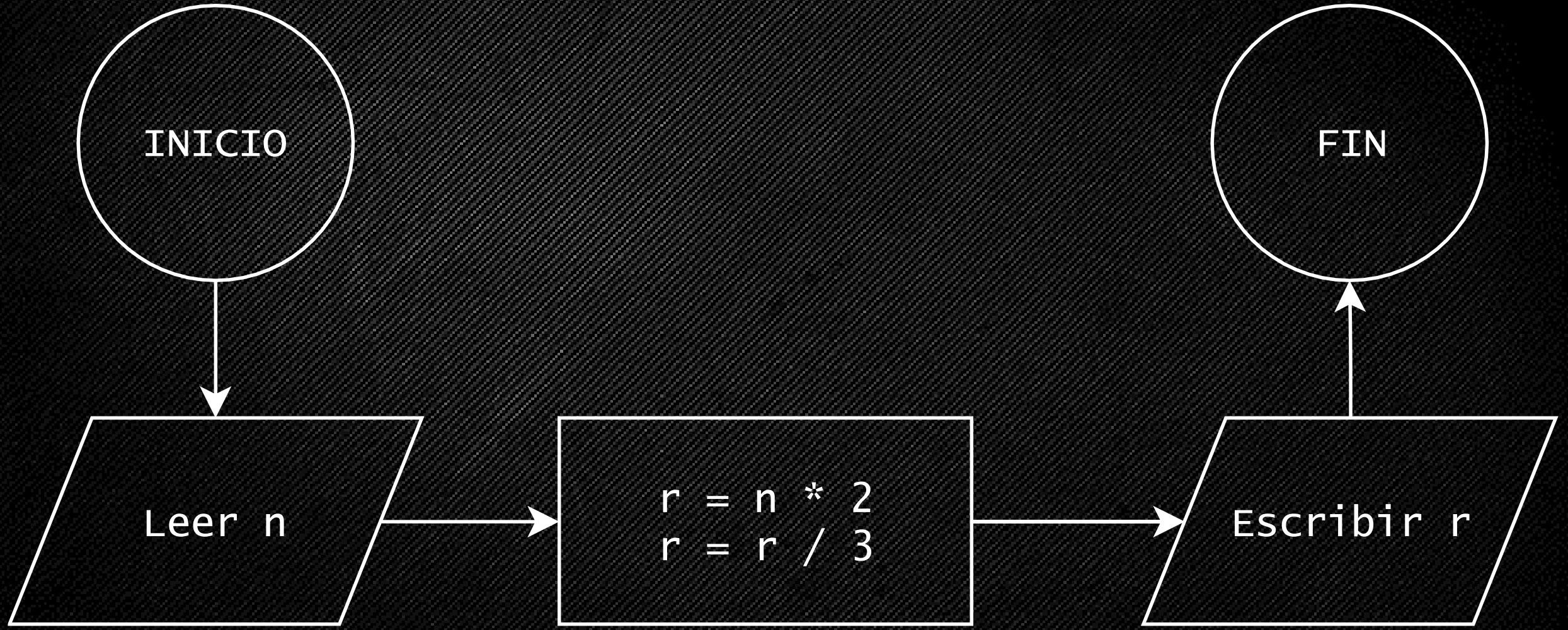
# Ejercicios

# Hola Mundo

# Saludar



# Duplicar y Dividir



# Múltiplos

Realice un programa que solicite un número al usuario y muestre en pantalla los primeros 5 múltiplos del mismo, según el siguiente ejemplo:

```
Ingrese un numero: 5
5
10
15
20
25
```

# Promedio de Notas

Realice un programa que solicite al usuario sus 3 notas de certámenes y muestre en pantalla su promedio final, de acuerdo al siguiente ejemplo:

```
Ingrese sus notas:
0
64
100
Su promedio final es: 54.333333333333336
```

# Áreas y Perímetros

Realice un programa que solicite al usuario un número **n** y muestre en pantalla el área y perímetro de un cuadrado, círculo y triángulo, asumiendo que todos son de lado/radio **n**

```
Ingrese un numero: 5
Cuadrado: A = 25, P = 20
Triángulo: A = 10.825317547305483, P = 15
Círculo: A = 78.53981633974483, P = 31.41592653589793
```

# Invertir un Número

Realice un programa que solicite al usuario un número de **3** dígitos y muestre en pantalla el número invertido según el siguiente ejemplo:

```
Ingrese un numero: 123
321
```

# Fin de la Ayudantía 2

Gonzalo Fernández

IWI-131 24-09-2018