
SISTEMAS WEB

CURSO 2021/2022

Tutorial de Python y PyCharm
M 27-01-2023



Web Sistemak by [Oskar Casquero](#) & [María Luz Álvarez](#) is licensed under a [Creative Commons Reconocimiento 4.0 Internacional License](#).

ÍNDICE

- Python:
 - Introducción
 - **PIP**: Administrador de librerías
 - **venv**: Entorno virtual
- PyCharm:
 - Crear un proyecto
 - Instalación de librerías
- Python:
 - Estructura de un programa
- PyCharm:
 - Ejecución de un programa
- Python:
 - Estructuras de datos: tuplas, listas y diccionarios

INTRODUCCION

- **Python :**
 - un lenguaje de programación **interpretado**.
 - un lenguaje de programación **multi-paradigma** que permite varios estilos: programación **orientada a objetos**, programación **imperativa** y programación **funcional**.
 - verificación del tipo de datos es **dinámica**.
 - y tiene soporte **multiplataforma**.
- Python permite escribir programas mas compactos y cortos que con C, C++ o Java
 - No es necesario declarar variables.
 - La agrupación de instrucciones se hace por sangría en vez de llaves de apertura y cierre

```
lista= []  
for i in range(0,10):  
    lista.append(random.random())
```

- Las operaciones complejas se pueden expresar en un solo comando.

```
>>> lista1 = [4, 5, 6, 7, 8, 9]  
>>> lista2 = [ lista1.index(x) for x in lista1 if x>6 ]  
>>> lista2  
[3, 4, 5]
```

- web de Python (<http://www.python.org/>) y documentación: <https://docs.python.org/2/download.html>

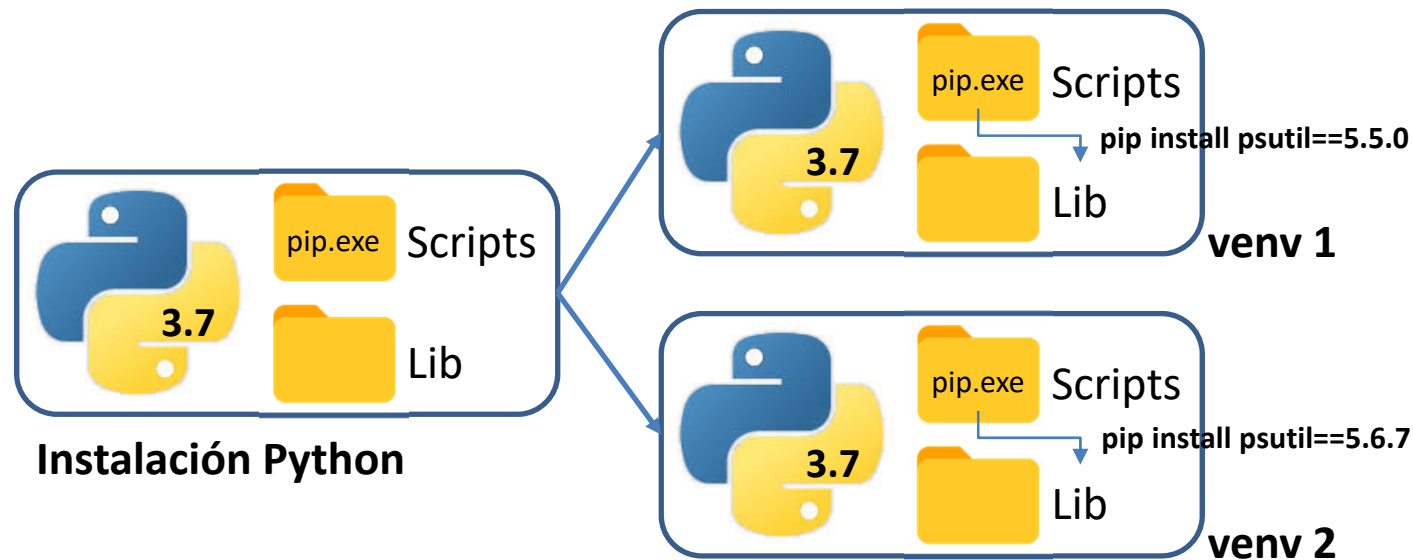
PIP: ADMINISTRADOR DE LIBRERÍAS

- Para instalar librerías en Python existen dos maneras:
 1. Descargarse el código fuente de la librería y ejecutar el siguiente comando: **python setup.py install**
 2. Utilizar el administrador de paquetes **PIP**. Por ejemplo, para instalar la librería *psutil*: **pip install psutil**

El ejecutable **pip** esta en la carpeta **\$PYTHONHOME/Scripts**; de acuerdo con nuestra configuración de instalación en Windows: **C:\Python37\Scripts\pip.exe**

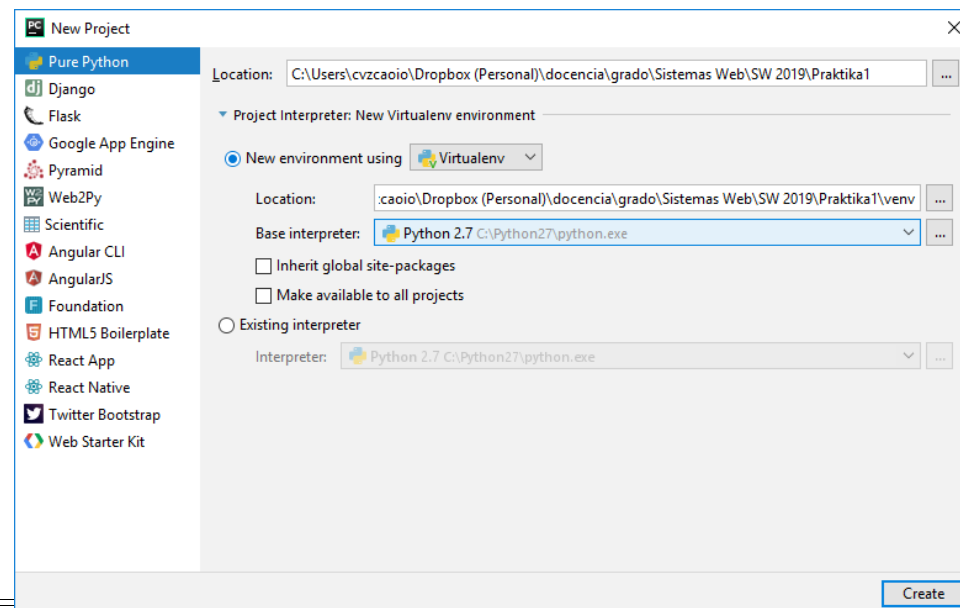
VENV: ENTORNO VIRTUAL

- Suponga que se desea desarrollar con la misma instalación de Python dos proyectos, ambos usan una librería particular (por ejemplo, *psutil*) pero versiones diferentes de esta librería (por ejemplo, v5.5.0 y v.5.6.7)
- A partir de una instalación de Python, los entornos virtuales permiten configurar entornos de desarrollo separados.



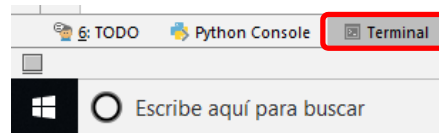
PYCHARM: CREACIÓN DE UN PROYECTO

- En la ventana de configuración de datos del proyecto, en el campo **Location** elije un lugar conocido y accesible para guardar el proyecto.
- Sin salir de la ventana de datos del proyecto, elige el **intérprete** que se utilizará en el proyecto. Para esto, se establecerá un entorno virtual:
 - Comprueba que en la zona despegable de **Python Interpreter** : Esta seleccionado en **New Virtual environment** esta seleccionado.
 - Además, verifica que **Python 3.11** se muestre en el campo **Base interpreter** .
- Finalmente, crea el proyecto. **NOTA: PyCharm tardará un tiempo en crear el proyecto.**

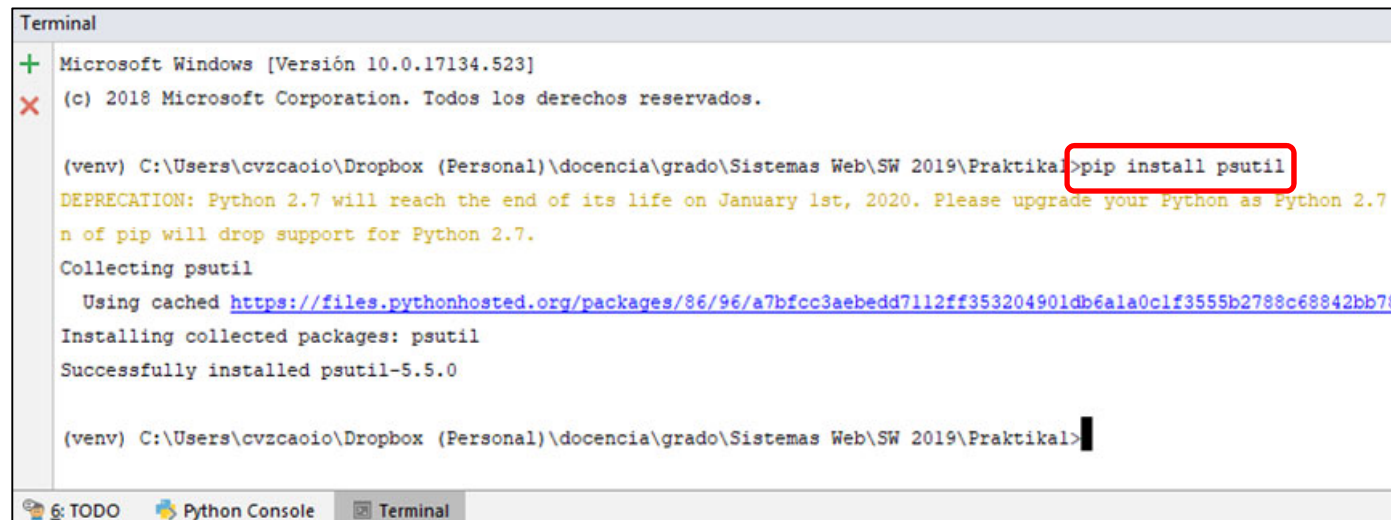


PYCHARM: INSTALACIÓN DE UNA LIBRERÍA (1/1)

- La librería **psutil** se utiliza para extraer datos sobre el uso del sistema de un ordenador (CPU, memoria, discos, red, sensores).
- Para instalar una librería en PyCharm tenemos dos opciones:
 1. Seleccionar el botón *Terminal* que aparece en la parte inferior de la ventana



y ejecutar el siguiente comando: **pip install psutil**

A screenshot of the PyCharm Terminal window. The terminal shows the command 'pip install psutil' being executed. The output indicates that the package is successfully installed. The command and its output are highlighted with red rectangular boxes. The terminal also shows a deprecation warning for Python 2.7.

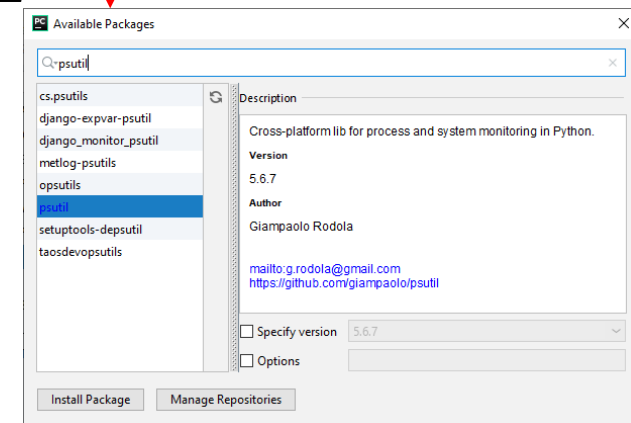
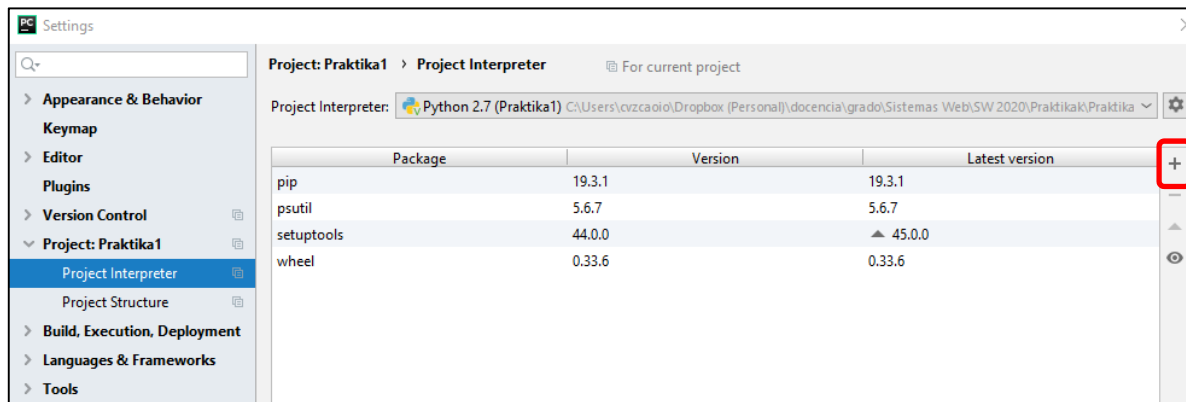
```
Terminal
+ Microsoft Windows [Versión 10.0.17134.523]
X (c) 2018 Microsoft Corporation. Todos los derechos reservados.

(venv) C:\Users\cvzcaio\Dropbox (Personal)\docencia\grado\Sistemas Web\SW 2019\Praktikal>pip install psutil
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7
n of pip will drop support for Python 2.7.
Collecting psutil
  Using cached https://files.pythonhosted.org/packages/86/96/a7bfcc3aebdd7112ff353204901db6a1a0clf3555b2788c68842bb78
Installing collected packages: psutil
Successfully installed psutil-5.5.0

(venv) C:\Users\cvzcaio\Dropbox (Personal)\docencia\grado\Sistemas Web\SW 2019\Praktikal>
```

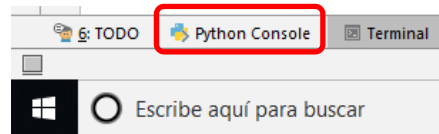
PYCHARM: INSTALACIÓN DE UNA LIBRERÍA (2/3)

- La segunda opción para instalar una librería en PyCharm:
 - Utilizar el administrador de librerías de PyCharm:
File → Settings → Project → Project Interpreter



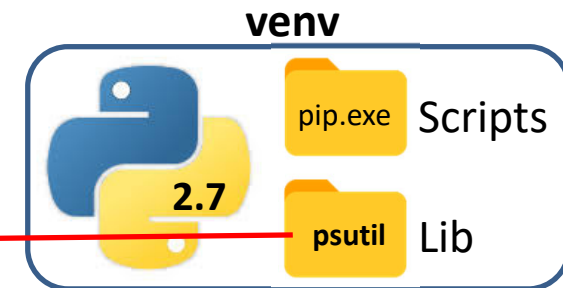
PYCHARM: INSTALACIÓN DE UNA LIBRERÍA (3/3)

- Verificaremos que **psutil** esté instalado en el lugar apropiado (en el entorno virtual)
- Seleccionar el botón *Python Console* que aparece en la parte inferior de la ventana



E importamos la librería

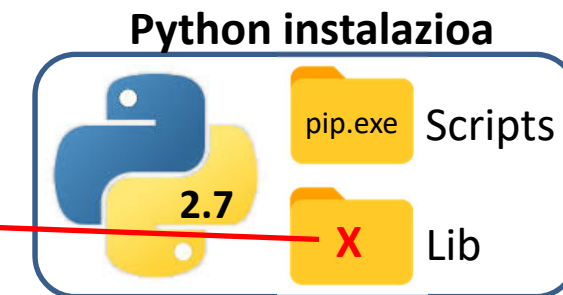
```
Python Console x
"C:\Users\cvzcaio\Dropbox (Personal)\docencia\grado\Sistemas Web\SW 2020\Praktik\Praktikal\venv\Scripts\python.exe"
import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['C:\Users\cvzcaio\Dropbox (Personal)\docencia\grado\Sistemas Web\SW 2020\Praktik\Praktikal\venv\Scripts'])
PyDev console: starting.
Python 2.7.17 (v2.7.17:c2f86d86e6, Oct 19 2019, 20:49:36) [MSC v.1500 32 bit (Intel)] on win32
>>> import psutil
>>>|
```



- Abra una terminal en el sistema operativo, abra una consola de Python e importe la librería:

```
C:\windows\system32\cmd.exe - C:\Python27\python.exe
Microsoft Windows [Versión 10.0.17763.914]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\cvzcaio>C:\Python27\python.exe
Python 2.7.17 (v2.7.17:c2f86d86e6, Oct 19 2019, 20:49:36) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import psutil
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named psutil
>>>
```



ESTRUCTURA DE PROGRAMA

```
import libreria

def metodo():
    CODIGO #comentario

if __name__ == "__main__":
    CODIGO
    metodo()
    CODIGO
```

Ejemplo

```
import time

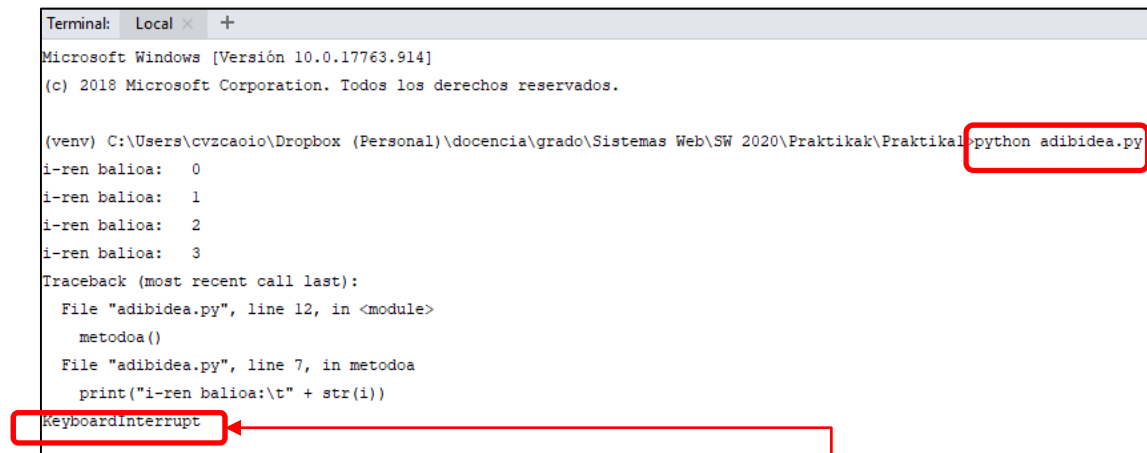
def metodo():
    i = 0
    while True:
        print("valor de i:\t" + str(i))
        i = i + 1
        time.sleep(1)

if __name__ == "__main__":
    metodo()
```

- El nivel de anidamiento se indica mediante sangría, se recomienda utilizar un sangría de cuatro espacios (no utilizar tabuladores).
- Además, entre las secciones principales (declaración de librerías, definición de método y llamada principal), se recomiendan dos líneas en blanco.

PYCHARM: EJECUCIÓN DE UN PROGRAMA

- En Python existen dos formas de ejecutar un programa
 1. **Terminal:** Seleccionar el botón *Terminal* que aparece en la parte inferior de la ventana de PyCharm y luego llame al intérprete de Python pasando el nombre del programa como parámetro



```
Terminal: Local x +
Microsoft Windows [Versión 10.0.17763.914]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

(venv) C:\Users\cvzcaoio\Dropbox (Personal)\docencia\grado\Sistemas Web\SW 2020\Praktikak\Praktikal>python adibidea.py
i-ren balioa: 0
i-ren balioa: 1
i-ren balioa: 2
i-ren balioa: 3
Traceback (most recent call last):
  File "adibidea.py", line 12, in <module>
    metooda()
  File "adibidea.py", line 7, in metooda
    print("i-ren balioa:\t" + str(i))
KeyboardInterrupt
```

Terminar la ejecución de un programa se utiliza Ctrl+C .

2. **Run → Run...** El programa se ejecuta en la ventana *Run*.

En este caso, al presionar Ctrl + C no termina la ejecución del programa. Porque PyCharm entiende Ctrl + C como el comando "copiar."

ESTRUCTURAS DE DATOS: TUPLAS, LISTAS Y DICCIONARIOS

- Tuplas: **this_tuple = ("apple ", 1.2 , "cherry")**
 - Se definen entre paréntesis.
 - es un conjunto **ordenado** e **inmutable** de datos del mismo o diferente tipo.
 - Para acceder aun elemento de la tupla se utiliza el índice: **this_tuple [1]=1,2**
- Listas: **this_list = ["orange", "kiwi", "melon"]**
 - Se define entre corchetes
 - Es un conjunto **ordenado** y **modificable** de datos del mismo o diferente tipo.
 - Para acceder aun elemento de la lista se utiliza el índice.
- Diccionario : **this_dict = {"nombre": "mango",
 "origen": "tropikoa",
 "color": [" amarillo", " naranja", " rojo", " verde"]} }**
 - Se definen entre llaves
 - Es un conjunto **no ordenado** de **pares clave-valor**, siendo las claves únicas dentro de un mismo diccionario.
 - Los claves deben ser cadenas y los valores pueden ser de cualquier tipo de datos
 - Se accede a los datos por el nombre de la clave.

DICCIONARIOS

<pre>persona = { 'nombre': 'Luz', 'apellido1': 'Alvarez', 'apellido2': 'Gutierrez'} print (persona)</pre>	<pre>{'nombre': 'Luz', 'apellido1': 'Alvarez', 'apellido2': 'Gutierrez'}</pre>
<pre>print (persona['nombre']) print (persona['apellido1'])</pre>	<pre>Luz Alvarez</pre>
<pre>for key in persona.keys(): print (key + ' = ' + persona[key])</pre>	<pre>nombre = Luz apellido1 = Alvarez apellido2 = Gutierrez</pre>
<pre>persona['ciudad'] = 'Bilbao' print (persona)</pre>	<pre>'nombre': 'Luz', 'apellido1': 'Alvarez', 'ciudad': 'Bilbao', 'apellido2': 'Gutierrez'}</pre>
<pre>otros_datos = {'fecha-inicio': '2016-01-01', 'NAN': '12345678-A'} persona.update(otros_datos) print (persona)</pre>	<pre>{'apellido2': 'Gutierrez', 'apellido1': 'Alvarez', 'nombre': 'Luz', 'NAN': '12345678-A', 'ciudad': 'Bilbao', 'fecha- inicio': '2016-01-01'}</pre>
Código	Salida

LISTAS (1/2)

<pre>asignaturas= ['Sistemas Web', 'Introducción a Redes de Computadoras', 'Diseno de Bases de Datos', 'Administracion de Base de Datos', 'Estructuras de Datos y Algoritmos'] print (asignaturas)</pre>	<pre>['Sistemas Web', 'Introducción a Redes de Computadoras', 'Diseño de Bases de Datos', 'Administración de Base de datos', 'Estructuras de Datos y Algoritmos']</pre>
<pre>print (asignaturas[0]) print (asignaturas[1]) print (asignaturas[-1]) print (asignaturas[-2]) print (asignaturas[:2]) print (asignaturas[2:])</pre>	<pre>Sistemas Web Introducción a Redes de Computadoras Estructuras de Datos y Algoritmos Administración de Base de Datos ['Sistemas Web', 'Introducción a Redes de computadoras'] ['Diseño de Bases de Datos', 'Administración de Base de datos', 'Estructuras de Datos y Algoritmos']</pre>
<pre>for asignatura in asignaturas : print (asignatura)</pre> <p>Código</p>	<pre>Sistemas Web Introducción a Redes de Computadoras Diseño de Bases de Datos Administración de Base de datos Estructuras de Datos y Algoritmos</pre> <p>Salida</p>

LISTAS (2/2)

<pre>nueva_lista= [i for i in asignaturas if i.find('Dato')!=-1] print (nueva_lista)</pre>	<pre>['Diseno de Bases de Datos', 'Administracion de Base de Datos', 'Estructuras de Datos y Algoritmos']</pre>
<pre>nueva_lista.append('Gestion de Proyectos') print (nueva_lista)</pre>	<pre>['Diseno de Bases de Datos', 'Administracion de Base de Datos', 'Estructuras de Datos y Algoritmos', 'Gestion de Proyectos']</pre>
<pre>print (nueva_lista.index ('Gestion de Proyectos'))</pre>	<pre>3</pre>
<pre>nueva_lista.remove('Estructuras de Datos y Algoritmos') print (nueva_lista)</pre>	<pre>['Diseno de Bases de Datos', 'Administracion de Base de Datos', 'Gestion de Proyectos']</pre>
<pre>if 'Gestion de Proyectos' in nueva_lista: print ('Esta en lista') else: print ('No esta en lista')</pre>	<pre>Esta en lista</pre>
Código	Salida

DICCIONARIOS Y LISTAS (1/2)

```
persona = {
    'id': 'mlalvgut',
    'desc': {
        'firstName': 'Luz',
        'lastName': ['Alvarez', 'Gutierrez']
    },
    'contact': {
        'email': 'marialuz.alvarez@ehu.es',
        'phone': '946014462'
    },
    'courses': [
        {
            'code': 27699,
            'desc': 'Introduction to Computer Networks'
        },
        {
            'code': 27702,
            'desc': 'Web Systems'
        }
    ]
}
print (persona)
```

Código

```
{'courses': [{'code': 27699, 'desc': 'Introduction to
Computer Networks'}, {'code': 27702, 'desc': 'Web
Systems'}], 'contact': {'phone': '946014462', 'email':
'marialuz.alvarez@ehu.es'}, 'id': 'mlalvgut', 'desc':
{'lastName': ['Alvarez', 'Gutierrez'], 'firstName': 'Luz'}}
```

Salida

DICCIONARIOS Y LISTAS (2/2)

<pre>id = persona['id'] print (id)</pre>	mlalvgut
<pre>firstName = persona['desc']['firstName'] print (firstName)</pre>	Luz
<pre>lastName1 = persona['desc']['lastName'][0] lastName2 = persona['desc']['lastName'][1] print (lastName1 + ' ' + lastName2)</pre>	Alvarez Gutierrez
<pre>email = persona['contact']['email'] print (email)</pre>	marialuz.alvarez@ehu.es
<pre>phone = persona['contact']['phone'] print (phone)</pre>	946014462
<pre>course_id = persona['courses'][1]['code'] print (course_id)</pre>	27702
Código	Salida