



PYTHON

Estructura y documentación

1

ESTRUCTURA Y DOCUMENTACIÓN

- Estructura de proyectos.
 - Estructuras
 - pip
 - Módulos y paquetes.
- Estructura del código.
 - Organización de ficheros.
 - Comentarios.
- Git.
- Documentación.

ESTRUCTURA Y DOCUMENTACIÓN

- Estructura de carpetas.
 - readme.
 - /config → Ficheros de configuración
 - /test → Pruebas de software
 - /src → Código fuente (ficheros Python)
 - /doc → Documentación general del proyecto
 - /apidoc → Documentación del API

ESTRUCTURA Y DOCUMENTACIÓN

- Instalador de paquetes: pip
- Módulos:
 - E:\Python\Python310\Lib → Esta es la ubicación en el ordenador del profesor. ¿Dónde tienes instalado Python?
 - Copiar los módulos propios a dicha carpeta (pueden estar en carpetas para mejorar la organización)

ESTRUCTURA Y DOCUMENTACIÓN

- Comentarios:
 - La importancia de los comentarios.
 - Destinatarios de los comentarios.
- Documentación
 - Comentarios vs Documentación
- Mensajes para el usuario
- Logs (traza)

ESTRUCTURA Y DOCUMENTACIÓN

- Comentarios:

- De línea: #
- De bloque: “”””

- Tags:

- # TODO
- # FIXME
- # BUG

ESTRUCTURA Y DOCUMENTACIÓN

- Logs:
 - Módulo logging
 - Configuración:
 - Establecer nivel de filtrado de logs:
 - `logging.getLogger().setLevel(logging.level)`
 - *level:*

Nivel	Valor numérico
CRITICAL	50
ERROR	40
WARNING	30
INFO	20
DEBUG	10
NOTSET	0

ESTRUCTURA Y DOCUMENTACIÓN

- **Logs:**

- **Módulo logging**

- **Uso:**

```
logging.debug("Depuración")
```

```
logging.info("Información")
```

```
logging.warning("Advertencia")
```

```
logging.error("Error")
```

```
logging.critical("Error crítico")
```

```
logging.exception("Excepción") → Nivel ERROR. Sólo para excepciones.
```


ESTRUCTURA Y DOCUMENTACIÓN

- Logs:
 - Módulo logging
 - Uso Exception:

```
try:  
    b=5  
    c=0  
    a = b/c  
except ZeroDivisionError as error:  
    logging.exception(error)
```

ERROR:root:division by zero

Traceback (most recent call last):

File "h:_ProyectosPython\borrar.py", line 15, in funcion

a = b/c

ZeroDivisionError: division by zero

ESTRUCTURA Y DOCUMENTACIÓN

- **Logs:**

- Módulo logging

- Configuración salida a fichero:

```
logging.basicConfig(filename="fichero_salida.log",  
                    filemode='a',  
                    format='%(asctime)s,%(msecs)d %(name)s %(levelname)s %(message)s',  
                    datefmt='%H:%M:%S',  
                    level=logging.DEBUG)
```

- **Atributos:**

- <https://docs.python.org/3/library/logging.html#logrecord-attributes>

ESTRUCTURA Y DOCUMENTACIÓN

- Comentarios:

- De línea: #
- De bloque: """

- Sugerencias de tipo:

```
def dame_numero(numero : int) -> int:  
    """  
    Devuelve el número 10  
    """  
    return 10
```

Sugerencia de tipo (Type Hinting)

```
2  gc.dame_numero()  
    dame_numero(numero: int) -> int  
    numero: int  
    Devuelve el número 10
```

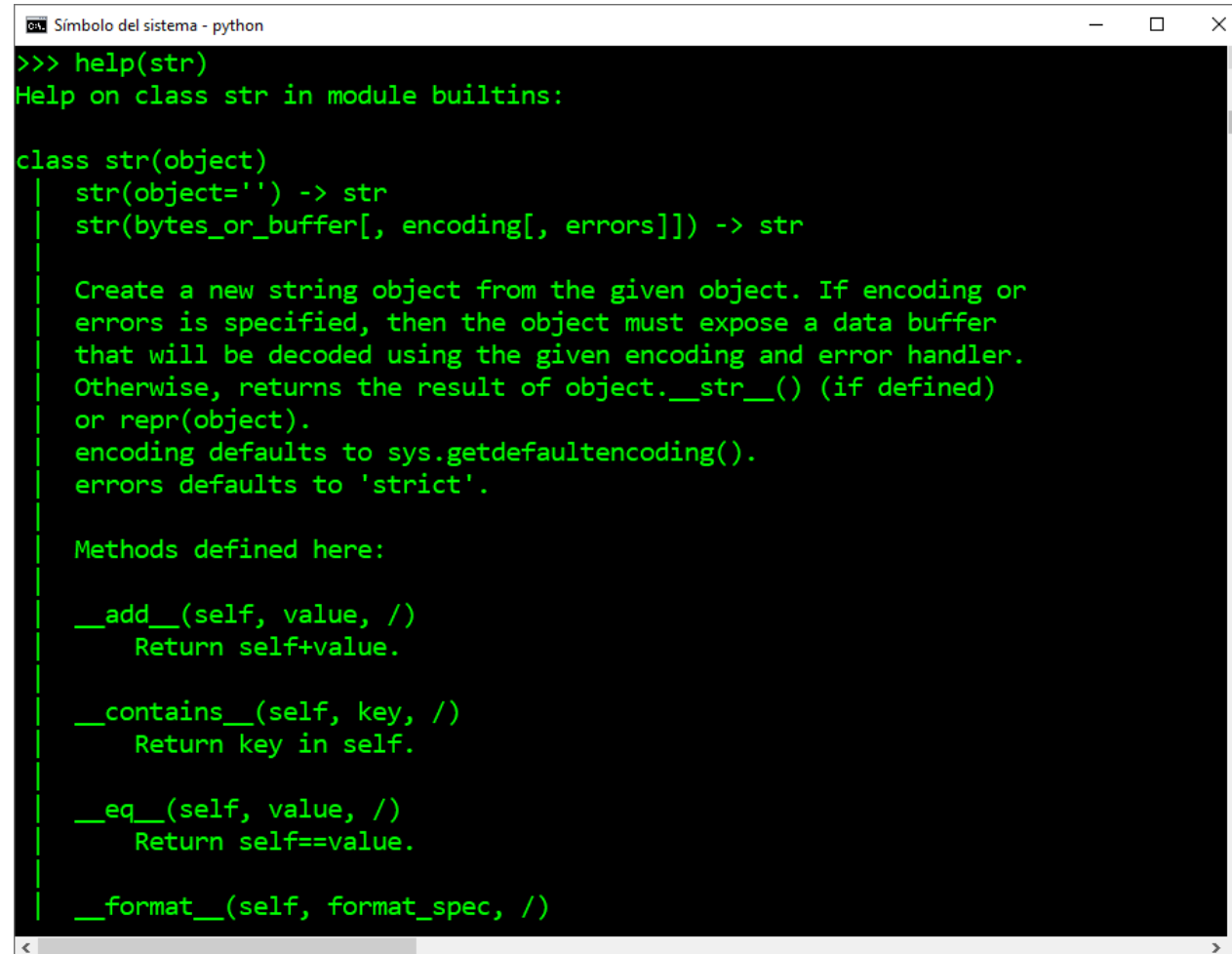
- Tags:

- # TODO
- # FIXME
- # BUG

ESTRUCTURA Y DOCUMENTACIÓN

■ Función help:

```
>>>help(str)
```



```
Símbolo del sistema - python
>>> help(str)
Help on class str in module builtins:

class str(object)
|   str(object='') -> str
|   str(bytes_or_buffer[, encoding[, errors]]) -> str
|
|   Create a new string object from the given object. If encoding or
|   errors is specified, then the object must expose a data buffer
|   that will be decoded using the given encoding and error handler.
|   Otherwise, returns the result of object.__str__() (if defined)
|   or repr(object).
|   encoding defaults to sys.getdefaultencoding().
|   errors defaults to 'strict'.
|
|   Methods defined here:
|
|   __add__(self, value, /)
|       Return self+value.
|
|   __contains__(self, key, /)
|       Return key in self.
|
|   __eq__(self, value, /)
|       Return self==value.
|
|   __format__(self, format_spec, /)
```

ESTRUCTURA Y DOCUMENTACIÓN

■ Docstring:

```
def dame_numero(numero : int) -> int:
    """
    Devuelve el número 10
    """
    return 10
```

```
>>> help(paniagua.generador_cosas)
Help on module paniagua.generador_cosas in paniagua:

NAME
    paniagua.generador_cosas

MODULE REFERENCE
    https://docs.python.org/3.9/library/paniagua.generador_cosas

    The following documentation is automatically generated from the Python
    source files. It may be incomplete, incorrect or include features that
    are considered implementation detail and may vary between Python
    implementations. When in doubt, consult the module reference at the
    location listed above.

FUNCTIONS
    dame_numero(numero: int) -> int
        Devuelve el número 10

FILE
    e:\python\python39\lib\paniagua\generador_cosas.py
```