

00.Operaciones_comandos_basicos

October 21, 2022

1 Operaciones y comandos básicos

Al igual que en otros lenguajes de programación hay un conjunto diverso de operadores que están disponibles:

- Aritméticos: +, -, *, /, **, %, //
- Lógicos: and, or, not

1.1 Usando Python como una calculadora

```
[ ]: 5+6
```

```
[ ]: 11
```

```
[ ]: 20-10
```

```
[ ]: 10
```

```
[ ]: 10/5
```

```
[ ]: 2.0
```

```
[ ]: 8*10
```

```
[ ]: 80
```

1.2 Comentarios python

Para los códigos en python se usa #

```
[ ]: # Ingresa un comentario
```

1.3 Funciones internas de python

El intérprete de Python cuenta con funciones internas que siempre se hallan disponibles.

Por ejemplo:

```
print() # imprime una variable.
abs()   # devuelve el valor absoluto.
dir()   # Devuelve la lista de nombres de la tabla de símbolos local actual.
len()   # Devuelve la longitud (el número de elementos) de un objeto.
zip()   # Esta función devuelve una lista de tuplas.

# etc.
# etc.
# etc.
```

Puedes tener una lista completa de las funciones internas haciendo click [aquí](#)

1.4 Obtener ayuda en Python

```
[ ]: help(print)
```

Help on built-in function print in module builtins:

```
print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file: a file-like object (stream); defaults to the current sys.stdout.
    sep:   string inserted between values, default a space.
    end:   string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.
```

1.5 Retornar al [índice](#)