**_1type/_2bool.py**

```python
# --- Funciones booleanas ---
es_instance = isinstance(10, int)      # True
es_subclass = issubclass(bool, int)    # True

# --- == != > < >= <=
# --- is / is not       in / not in
('1 is 1', 1 is 1)                   # True
('1 is not 2', 1 is not 2)           # True
('A in Asabeneh', 'A' in 'Asabeneh') # True
('B in Asabeneh', 'B' in 'Asabeneh') # False
('a in an:', 'a' not in 'an')        # False

# --- and or not ---
(not 3 > 2)       # False - cause 3 > 2 is true, then not True gives False
(not True)        # False - Negation, the not operator turns true to false
(not False)       # True
(not not True)    # True
(not not False)   # False

# --- Comparación booleana ---
x = ""        # False
y = 0         # False
x = 5         # True
x = None      # False
x = []        # False
x = [1, 2, 3] #True
print(True == 1)     # True
print(False == 0)    # True
print(True + True)   # 2
print(False + True)  # 1
and_operacion = x and y  # False
or_operacion = x or y    # True
not_operacion = not x    # False
class CustomObject:   #False
    def __bool__(self):
        return False

# --- Booleanos y operaciones bit a bit ---
bitwise_and = True & False  # False
bitwise_or = True | False   # True
bitwise_xor = True ^ False  # True
bitwise_not = not True      # False

# --- Ejemplo práctico: función de validación ---
def es_mayor_de_edad(edad):
    return edad >= 18

print(es_mayor_de_edad(20))  # True
print(es_mayor_de_edad(16))  # False

```