

TUTORED & PRACTICAL WORK N°03

BASIC TYPES AND STRINGS IN PYTHON

Exercise 1 (Basic types)

Give the output of this program:

```
# The number 26, in decimal
decVal = int(26)
print("Decimal value: ", decVal)

# The number 26, in hexadecimal
hexVal = 0x1a
print("Decimal value of a hexadecimal variable: ", hexVal)
hexVal = int("0x1a",16)
print("Decimal value of a hexadecimal variable: ", hexVal)
print("Hexadecimal value of a hexadecimal variable: ", hex(hexVal))
hexVal = hex(0x1a)
print("Hexadecimal value of a hexadecimal variable: ", hexVal)
hexVal = hex(26)
print("Hexadecimal value of a hexadecimal variable: ", hexVal)

# The number 26, in octal
octVal = 0o32
print("Decimal value of an octal variable: ", octVal)
octVal = int("0o32",8)
print("Decimal value of an octal variable: ", octVal)
print("Octal value of an octal variable: ", oct(octVal))
octVal = oct(0o32)
print("Octal value of an octal variable: ", octVal)
octVal = oct(26)
print("Octal value of an octal variable: ", octVal)

# The number 26, in binary
binVal = 0b11010
print("Decimal value of a binary variable: ", binVal)
binVal = int("0b11010",2)
print("Decimal value of a binary variable: ", binVal)
print("Binary value of a binary variable: ", bin(binVal))
binVal = bin(0x1a)
print("Binary value of a binary variable: ", binVal)
binVal = bin(26)
print("Binary value of a binary variable: ", binVal)

## Double is coded on 32 bits
## 1 bit for the sign, 8 bits for the exponent, and 23 bits for the value.
d1 = 123.4
print("Double value of a double variable: ", d1)

# same value as d1, but in scientific notation
d2 = 1.234e2
print("Double value of a scientific notation: ", d2)

## Float is more precise than double, it is coded on 64 bits
## 1 bit for the sign, 11 bits for the exponent, and 52 bits for the value.
f1 = float(1234)
print("Float value of a float variable: ", f1)
```

Exercise 2 (Operators)

Examine the following program and give the display after its execution.

```
x = 5

print("x % 2: ", x % 2)
print("x // 2: ", x // 2)
print("x ** 2: ", x ** 2)
print("x & 2: ", x & 2)
print("x | 2: ", x | 2)
print("x ^ 2: ", x ^ 3)
print("x >> 1: ", x >> 1)
print("x >> 1: ", x << 1)
```

Exercise 3 (Strings)

Write a program that asks the user for a string of characters. If the string is empty display an error message, otherwise display its length. Loop until the string contains "quit".