 

**Python para Ingeniería de Telecomunicación Javier Gracia**

# Práctica 2: Cadenas

En esta primera práctica se van a dar los primeros pasos en programación de Python y consta de **una sesión de laboratorio**. Se deben realizar 7 ejercicios en total que podrán ser entregados en la correspondiente tarea de Poliformat. Se puede disponer de todo tipo de material para resolver los problemas y los ejercicios se pueden entregar por parejas o individualmente. El objetivo es utilizar cadenas básicas de Python mediante ejercicios comunes de iniciación para dar nuestros siguientes pasos en este lenguaje de programación.

## *Ejercicio 1 (1 punto)*

Escribir un programa que pregunte el nombre del usuario en la consola y un número entero e imprima por pantalla en líneas distintas el nombre del usuario tantas veces como el número introducido.

nombre = input("Por favor, dime tu nombre: ")

numero = int(input("Dime un número entero: "))

for i in range(numero):

print(nombre)

Texto

Descripción generada automáticamente

## *Ejercicio 2 (1 punto)*

Escribir un programa que pregunte el nombre completo del usuario en la consola y después muestre por pantalla el nombre completo del usuario tres veces, una con todas las letras minúsculas, otra con todas las letras mayúsculas y otra solo con la primera letra del nombre y de los apellidos en mayúscula. El usuario puede introducir su nombre combinando mayúsculas y minúsculas como quiera.

*(Pista: name.lower, name.upper, name.title)*

nombre = input("Por favor, dime tu nombre: ")

print(nombre.lower())

print(nombre.upper())

print(nombre.title())

Texto

Descripción generada automáticamente

## *Ejercicio 3 (1 punto)*

Escribir un programa que pregunte el nombre del usuario en la consola y después de que el usuario lo introduzca muestre por pantalla la cadena ¡Hola <nombre>!, donde <nombre> es el nombre que el usuario haya introducido. Y mostrar el número de letras que tiene <nombre>

### (Pista: len)

nombre = input("Por favor, dime tu nombre: ")

print("¡Hola " + nombre + "!")

numero\_letras = len(nombre)

print(f"Tu nombre tiene {numero\_letras} letras")

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

## Ejercicio 4 (1 punto)

Escribir un programa que pida al usuario que introduzca una frase en la consola y muestre por pantalla la frase invertida.

Pista:

def invertir\_cadena(cadena):

return cadena[::-1]

frase = input("Por favor, introduce una frase: ")

frase\_invertida = frase[::-1]

print(frase\_invertida)



## Ejercicio 5 (2 puntos)

Escribir un programa que pida al usuario que introduzca una frase en la consola y una vocal, y después muestre por pantalla la misma frase pero con la vocal introducida en mayúscula.

Pista: Buscar *replace()* en esta interesante [página.](https://www.programiz.com/python-programming/methods/string)

frase = input("Por favor, dime una frase: ")

vocal = input("Por favor, dime una vocal: ")

frase\_modificada = frase.replace(vocal, vocal.upper())

print(frase\_modificada)

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

## Ejercicio 6 (2 puntos)

Escribir un programa que pregunte por consola el precio de un producto en euros con dos decimales y muestre por pantalla el número de euros y el número de céntimos del precio introducido.

Pista: Cadena *find*

precio = input("Dime el precio con 2 decimales ")

posicion = precio.find(".")

euros = precio[:posicion]

centimos = precio[posicion+1:]

print("El precio es de", euros, "euros y", centimos, "céntimos.")



## Ejercicio 7 (2 puntos)

Escribir un programa que pregunte por consola por los productos de una cesta de la compra, separados por comas, y muestre por pantalla cada uno de los productos en una línea distinta.

Pista: Cadena *Split.* Recordad igualmente que en Python, \n es el carácter de salto de línea y se usa para indicar el fin de una línea de texto y el inicio de una línea nueva.

compra = input("Dime tu cesta de la compra: ")

productos = compra.split(",")

for producto in productos:

print(producto)

Texto

Descripción generada automáticamente