

Clase 03

Ejercicio 1

Imprimir todos los números pares y su cuadrado del 0 al 100.

```
for i in range(0,101,2):
    print(i, " ", i**2)
```

Alternativa con *while*

```
i = 0
while i <= 100:
    print ("El número es: " + str(i))
    print ("Y su cuadrado es: " + str(i**2))
    print ("-----")
    # incrementa en 2
    i = i+2
```

Ejercicio 2

Dada una palabra ingresada por el usuario contar cuantas ocurrencias de la letra "a" tiene.

```
palabra = input("Ingrese una palabra: ")
i = 0
ocurrencia = 0

while i < len(palabra):
    if palabra[i] == "a":
        ocurrencia = ocurrencia + 1
    i = i+1

print("Hay {} ocurrencias de la letra 'a' en la palabra {}".format(ocurrencia, palabra))
```

Alternativa con *for*

```
palabra = input("ingresar palabra")
conteo = 0

for i in palabra:
    if i=="a":
        conteo+=1

print("Hay {} ocurrencias de la letra 'a' en la palabra {}".format(conteo, palabra))
```

Ejercicio 3

Dada una cadena de caracteres:

Imprima dicha cadena cada dos caracteres. Ej.: 'recta' debería imprimir 'rca'

```
palabra = input("ingresar palabra")
# inicializa un string vacío
npal = ""
for i in range(0,len(palabra),2):
    npal = npal+palabra[i]

print(npal)
```

Alternativa con *while*

```

palabra = input("ingresar palabra")
i = 1
resultado = ""
while i <= len(palabra):
    resultado = resultado + palabra[i-1]
    i = i+2
print(resultado)

```

¿El índice es par?

```

cont = 0
txt = input("Ingrese palabra: ")
for x in txt:
    if(not cont%2):
        print(x, end="")
    cont+=1

```

Ejercicio 4

Manejo de contraseñas

a) Escribir un programa que contenga una contraseña inventada y que se la pregunte al usuario. No permitirá continuar hasta que haya ingresado correctamente.

```

pas = "1234"
txt = ""
while pas!=txt:
    txt = input("Ingrese contraseña: ")

```

b) Modificar el programa anterior para que solamente permita una cantidad fija de intentos.

```

intentos = 4
pas = "1234"
txt = input("Ingrese contraseña: ")
while pas!=txt and (intentos-1) > 0:
    print("Intentos restantes", (intentos-1))
    txt = input("Ingrese contraseña: ")
    intentos-=1

```

c) Modificar el programa anterior para que después de cada intento agregue una pausa cada vez mayor, utilizando la función sleep del módulo time.

import time

time.sleep(secs)

```

import time

intentos = 0
TOTAL_PERMITIDO = 4
pas = "1234"
txt = input("Ingrese contraseña: ")
while pas != txt and intentos < (TOTAL_PERMITIDO -1):
    if txt != pas:
        time.sleep(1)
        print("Intentos restantes", (TOTAL_PERMITIDO-1) - intentos)
        txt = input("Ingrese contraseña: ")

    intentos += 1

```

```
while (clave != contrasena):
    time.sleep(segundos**2)
    clave = input("ingrese la contraseña: ")
    restantes = restantes - 1
    if(restantes == 0):
        print("se ingreso muchas veces la clave de forma incorrecta")
        break

if(restantes != 0):
    print("clave ingresada correctamente")
```

Ejercicio 5

Utilizando la función `randrange` del módulo `random`, escribir un programa que obtenga un número aleatorio secreto, y luego permita al usuario ingresar números y le indique si son menores o mayores que el número a adivinar, hasta que el usuario ingrese el número correcto.

import random

valor = random.randrange(start, stop)

```
import random
nro = random.randrange(1, 10)
adivina = False

while not adivina:
    ing = int(input("Ingrese numero: "))
    if ing < nro:
        print("El número ingresado es menor.")
    elif ing > nro:
        print("El número ingresado es mayor.")
    else:
        print("Muy bien!!")
        adivina = True
```