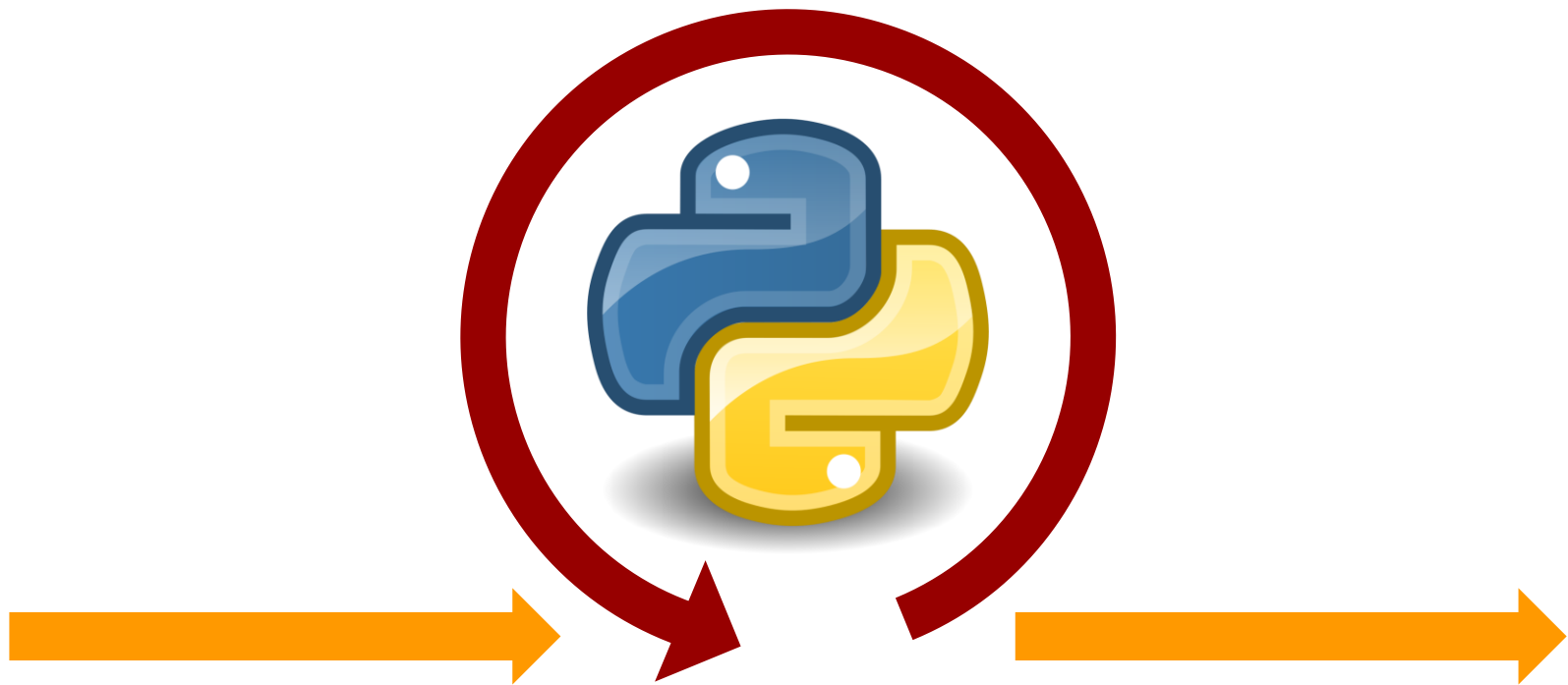


Python (III)

Programación (tics 100)
Semestre 01/2020

¿Qué veremos hoy?

Ciclos en Python



Condicionales

Problema

Escribir un programa en Python que pregunte el nombre y la edad de una persona y luego imprima por pantalla si es o no mayor de edad.

Pasos

1. Preguntar nombre
2. Recibir y almacenar el nombre del usuario
3. Preguntar edad
4. Recibir y almacenar la edad de la persona
5. Convertir y almacenar la edad de la persona a un número entero
6. Si la edad de la persona es mayor o igual a 18 imprimir que es mayor de edad. De lo contrario imprimir que aún es menor de edad

```
print("¿Cómo te llamas?")
nombre=input()
print("¿Cuántos años tienes?")
edad=input() #dato edad ingresa como
texto
edad=int(edad) #transforma edad de texto a entero
if(edad>=18):
    print(nombre, "ya eres mayor de edad")
else:
    print(nombre, "aún eres menor de edad")
```



Ojo con esta transformación: Si no la haces, no podrás manipular la variable como entero

Condicionales

Abran <https://es.akinator.com/> y jueguen 5 minutos
¿Cómo escribirían Akinator en Python?

▼ Python

```
print("¿Tu personaje tiene alguna  
placa de YouTube?")  
respuesta=input()  
if(respuesta == 'si'):  
    youtuber = True  
else:  
    youtuber = False
```



¿Qué sucede si queremos preguntarle datos a la persona hasta que adivinemos ?



Ciclos en Python

Ciclos son el equivalente a cuando, en nuestro algoritmo, escribimos

Mientras <condición> entonces

En el caso de Akinator, queremos que **Mientras** akinator no adivine **entonces** seguimos preguntando

Ciclos en Python

Hay dos maneras de escribir ciclos en Python:

- Basado en una condición -> Ciclo de tipo **while**
 - Akinator
- Basado en un número de veces fijo -> Ciclo de tipo **for**
 - Bart Simpson al ser castigado y debe escribir 100 veces en la pizarra

Ciclos en Python

El ciclo while se usa para repetir un bloque de instrucciones mientras una condición sea verdadera.

```
while <condicion>:  
    instrucciones
```



IMPORTANTE: Las instrucciones **DEBEN** contener código que haga que la condición cambie de valor

En el ciclo while, inicialmente se verifica la **condicion**. El programa ingresa al bloque de instrucciones del ciclo solo si **condicion** se evalúa como verdadera.

Después de verificar una iteración, el proceso continúa hasta que **condicion** se evalúe como falsa.

Ciclos en Python



Señaliza el inicio del ciclo while

Mientras (no adivine) Entonces
Escribir pregunta
Obtener respuesta
Si sé:
Escribir nombre
Obtener respuesta



▼ Python

```
guess = 'no'
while(guess != 'si')!
    print(pregunta())
    answer = input()
    if answer == 'si':
        print(make_a_guess())
        guess = input()
```



Las instrucciones del ciclo **while** deben estar indentadas.



Para indicar fin de ciclo, se vuelve a escribir sin indentación.

Ejemplo

Partiendo del número 1, mientras el número sea menor o igual a 10, mostrarlo en pantalla y luego sumarle 1

Mostrar los números del 1 al 10 por pantalla



▼ Python

```
i = 1
while(i < 10):
    print(i)
    i = i + 1
```



¿Qué se muestra por pantalla en este caso?



¿Cómo lo arreglamos?

Ejemplo

Partiendo del número 10, mientras el número sea mayor a 0, mostrarlo en pantalla y luego restarle 1

Mostrar los números del 10 al 1 por pantalla



▼ Python

```
i = 10
while(i > 0):
    print(i)
    i = i - 1
```



¿Qué se muestra por pantalla en este caso?

break

Python nos entrega una manera de poder terminar un ciclo de manera anticipada

▼ Python

```
while( True ):
    print('Escribe 0 para salir')
    respuesta = int( input() )
    if(respuesta == 0):
        break
```

¡La condición siempre se cumple!



Break nos permite “escapar” de un ciclo while antes que se deje de cumplir la condición



¡Ahora Ustedes!



Tiempo : 3 minutos

¿Qué se despliega por pantalla al ejecutar este programa escrito en Python cuando se ingresan los números 7 y 13?

```
print("Ingresa un numero")
num=input()
num=int(num)
while(num<10):
    print("Ingresa un numero")
    num=input()
    num=int(num)
if(num<20):
    if(num%2==0):
        print("No me digas ")
    else:
        print("Dime ")
else:
    print("Ya te dije ")
if(num>=10):
    print("que es mentira")
```

▼ Python

ALTERNATIVA A)

Ingresa un numero
7
13
Dime
que es mentira

ALTERNATIVA B)

Ingresa un numero
7
Ingresa un numero
13
Dime
que es mentira

ALTERNATIVA C)

7
Ingresa un numero
13
Ya te dije
Que es mentira

ALTERNATIVA D)

7
13
Dime

¡Ahora Ustedes!



Tiempo : 3 minutos

¿Qué se despliega por pantalla al ejecutar este programa escrito en Python cuando se ingresan los números 7 y 13?

```
print("Ingresa un numero")
num=input()
num=int(num)
while(num<10):
    print("Ingresa un numero")
    num=input()
    num=int(num)
if(num<20):
    if(num%2==0):
        print("No me digas ")
    else:
        print("Dime ")
else:
    print("Ya te dije ")
if(num>=10):
    print("que es mentira")
```

▼ Python

ALTERNATIVA A)

Ingresa un numero
7
13
Dime
que es mentira

ALTERNATIVA B)

Ingresa un numero
7
Ingresa un numero
13
Dime
que es mentira

ALTERNATIVA C)

7
Ingresa un numero
13
Ya te dije
Que es mentira

ALTERNATIVA D)

7
13
Dime

¡Ahora Ustedes!



Tiempo : 3 minutos

¿Qué dato(s) de entrada debes ingresar para que aparezca en pantalla el mensaje “No me digas que es mentira” **inmediatamente** después de ingresar un número?

```
print("Ingresa un numero")
num=input()
num=int(num)
while(num<10):
    print("Ingresa un numero")
    num=input()
    num=int(num)
if(num<20):
    if(num%2==0):
        print("No me
digas ")
    else:
        print("Dime ")
else:
    print("Ya te dije ")
if(num>=10):
    print("que es mentira")
```

▼ Python

ALTERNATIVA A)

Un número impar,
mayor o igual a 10 y
menor a 20

ALTERNATIVA B)

Un número impar,
menor que 20

ALTERNATIVA C)

Un número par,
menor que 20 y
mayor igual a 10

ALTERNATIVA D)

No hay manera de
que aparezca ese
mensaje

¡Ahora Ustedes!



Tiempo : 3 minutos

¿Qué dato(s) de entrada debes ingresar para que aparezca en pantalla el mensaje “No me digas que es mentira” **inmediatamente** después de ingresar un número?

```
print("Ingresa un numero")
num=input()
num=int(num)
while(num<10):
    print("Ingresa un numero")
    num=input()
    num=int(num)
if(num<20):
    if(num%2==0):
        print("No me digas ")
    else:
        print("Dime ")
else:
    print("Ya te dije ")
if(num>=10):
    print("que es mentira")
```

▼ Python

ALTERNATIVA A)

Un número impar, mayor o igual a 10 y menor a 20

ALTERNATIVA B)

Un número impar, menor que 20

ALTERNATIVA C)

Un número par, menor que 20 y mayor igual a 10

ALTERNATIVA D)

No hay manera de que aparezca ese mensaje

¡Ahora Ustedes!



*Tiempo : 4
minutos*

Escribe lo que va apareciendo en pantalla en la medida que se ejecuta este programa y se van ingresando los números 7, 9 y 8.

```
par=0
while(par==0):
    print("Ingresa un numero")
    num=int(input())
    if(num%2!=0):
        print("Ese numero no me gusta")
    else:
        par=1
        print("Ese numero si me gusta!")
print("Los divisores de ese numero son:")
i=1
while(i<=num):
    if(num%i==0):
        print(i)
    i=i+1
```

▼ Python

¡Ahora Ustedes!



*Tiempo : 4
minutos*

Escribe lo que va apareciendo en pantalla en la medida que se ejecuta este programa y se van ingresando los números 7, 9 y 8.

```
par=0
while(par==0):
    print("Ingresa un numero")
    num=int(input())
    if(num%2!=0):
        print("Ese numero no me gusta")
    else:
        par=1
        print("Ese numero si me gusta!")
print("Los divisores de ese numero  
son:")
i=1
while(i<=num):
    if(num%i==0):
        print(i)
    i=i+1
```

▼ Python

```
Ingresa un numero
7
Ese numero no me gusta
Ingresa un numero
9
Ese numero no me gusta
Ingresa un numero
8
Ese numero si me gusta!
Los divisores de ese numero son:
1
2
4
8
```

¡Ahora Ustedes!



*Tiempo : 8
minutos*

Escribe un programa en Python que solicite números al usuario hasta que este ingrese el número 0.

Una vez que el usuario ingresa el 0, el programa despliega el valor del mayor número ingresado (sin contar el cero).

¡Ahora Ustedes!



Tiempo : 8 minutos

Escribe un programa en Python que solicite números al usuario hasta que este ingrese el número 0.

Una vez que el usuario ingresa el 0, el programa despliega el valor del mayor número ingresado (sin contar el cero).

```
fin=0
max=1
while(fin==0):
    num=int(input("Ingresa un numero. Para terminar, ingresa 0"))
    if(num==0):
        fin=1
    else:
        if(num>max):
            max=num
print(f"El mayor valor ingresado fue: {max}")
```

▼ Python

¡Ahora Ustedes!



Tiempo : 10 minutos

Escribe un programa en Python que se mantenga solicitando números al usuario hasta que éste ingrese un 0. En ese caso, el programa le indica cuántos números ingresó (sin contar el 0) y cuánto suman. Ejemplos de ejecución del programa:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
```

```
>
Ingresa un número 1
Ingresa un número 2
Ingresa un número 3
Ingresa un número 4
Ingresa un número 5
Ingresa un número 6
Ingresa un número 7
Ingresa un número 0
Ingresaste 7 números que suman 28
>
```

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
```

```
>
Ingresa un número 0
Ingresaste 0 números que suman 0
>
```

¡Ahora Ustedes!



Tiempo : 10 minutos

Escribe un programa en Python que se mantenga solicitando números al usuario hasta que éste ingrese un 0. En ese caso, el programa le indica cuántos números ingresó (sin contar el 0) y cuánto suman. Ejemplos de ejecución del programa:

```
N=int(input("Ingresa un número"))
contador=1
suma=N

while(N!=0):
    N=int(input("Ingresa un número"))
    contador=contador+1
    suma=suma+N

print(f"Ingresaste {contador-1} números que suman {suma}")
```

▼ Python

Actividad en parejas

Piedra, papel o tijera, conocido también como cachipún, es un juego de manos en el que existen tres elementos: la piedra que vence a la tijera rompiéndola, la tijera que vence al papel cortándolo y el papel que vence a la piedra envolviéndola.

Desarrolle un programa en python que permita al usuario ingresar la cantidad de rondas a jugar; la opción seleccionada (piedra, papel o tijera) por el jugador, si existe algún ganador indicar quién gano la ronda (o empate). Al finalizar el juego mostrar el jugador ganador (o si hay empate).

Ejemplo 1 de juego:

```
Indique cantidad de rondas a jugar:3
Ingrese elección del Jugador 1 (piedra, papel o tijera):piedra
Ingrese elección del Jugador 2 (piedra, papel o tijera):papel
Ronda ganada por el Jugador 2

Ingrese elección del Jugador 1 (piedra, papel o tijera):tijera
Ingrese elección del Jugador 2 (piedra, papel o tijera):piedra
Ronda ganada por el Jugador 2

Ingrese elección del Jugador 1 (piedra, papel o tijera):papel
Ingrese elección del Jugador 2 (piedra, papel o tijera):piedra
Ronda ganada por el Jugador 1

Juego ganado por el Jugador 2 con 2 rodas ganadas
```

Nota: considerar solo minúsculas para el ingreso de las opciones piedra, papel o tijera.