

OPTATIVO 1 – PYTHON I

Prof. Ing. Aaron Zárate



Conversión de tipos de datos

La conversión de tipos de datos, también conocida como casting, es una técnica para manipular datos que no están en el tipo requerido.

Podemos hacer conversiones desde y hacia distintos tipos de datos.

- Convertir a entero → Función `int()`
- Convertir a flotante → Función `float()`
- Convertir cadena → Función `str()`
- Convertir a booleano → Función `bool()`

Conversión de tipos de datos

```
✓ #Conversion de tipos de datos
#Convertir de cadena a numero
numero_cadena = "10"
numero_entero = int(numero_cadena)
print(f"Valor numerico en cadena: {numero_cadena}")
print(f"Cadena a entero: {numero_entero}")
# Convertir de cadena a flotante
numero_cadena = "3.14"
numero_flotante = float(numero_cadena)
print(f"Valor numerico en cadena: {numero_cadena}")
print(f"Cadena a flotante: {numero_flotante}")
#Convertir de numero a cadena
numero_entero = 25
numero_cadena = str(numero_entero)
print(f"Convertir de numero a cadena: {numero_cadena}")
#Convertir a booleano
# Tipo booleano es falso en los siguiente casos
#Si el valor es 0, cadena vacia o None, entocnes regresa a false
# Regresa True , si el valor es distinto de 0, si es distinto
# de cadena vacia y tambien si es distinto de None
```

Conversión de tipos de datos

```
numero_entero = 0
booleano = bool(numero_entero)
print(f"Valor booleano de cero: {booleano}")

numero_entero = 5
booleano = bool(numero_entero)
print(f"Valor booleano de cinco: {booleano}")

cadena = "" # el largo de la cadena es cero
booleano = bool(cadena)
print(f"Valor booleano de cadena vacía: {booleano}")

cadena = "Cadena con valor"
booleano = bool(cadena)
print(f"Valor booleano de cadena NO vacía: {booleano}")

variable = None
booleano = bool(variable)
print(f"El valor booleano del tipo None es: {booleano}")
```

Ejemplo

 ejemplo_tipo_datos.py ×

```
1  #Ejemplo tipos de datos
2  #Concatenacion suma de valores
3  numero1_cadena = "10"
4  print(f"Numero 1 den cadena: {numero1_cadena}")
5  numero2_cadena= "20"
6  print(f"Numero 2 en cadena: {numero2_cadena}")
7  resultado= numero1_cadena + numero2_cadena
8  print(f"Concatenacion: {resultado}")
9  #Convertimos a tipos enteros
10 numero1_entero =int(numero1_cadena)
11 numero2_entero = int(numero2_cadena)
12 resultado = numero1_entero + numero2_entero
13 print(f"Resultado de la suma valores: {resultado}")
```

C:\Cursos\Python\Entrada_Datos_Consola\p

Numero 1 den cadena: 10

Numero 2 en cadena: 20

Concatenacion: 1020

Resultado de la suma valores: 30

Entrada de datos por consola

En Python la entrada de datos se realiza usando la función `input()` .


Esta función pausa la ejecución del programa y espera a que el usuario introduzca algún texto desde el teclado.

Una vez que el usuario presiona ENTER el texto introducido se devuelve como una cadena (str)

Características de la función `input()`

- Interactividad: Permite a los usuarios de nuestros programas proporcionar valores dinámicos, en lugar de utilizar valores en código duro o estáticos.
- Sencillez: La función es muy fácil de usar y solo necesita, opcionalmente, indicar la cadena o mensaje a mostrar al usuario, para que este sepa lo que se le esta solicitando.
- Tipo de dato: Siempre devuelve una cadena, si requiere otro tipo hay que convertirlo.

Ejemplo

 entrada_datos_consola.py ×

```
1  #entrada de datos por consola
2  nombre = input()
3  print(f"Recibiendo el valor de nombre: {nombre}")
4
```

C:\Cursos\Python\Entrada_Datos_Consola\pyt

Juan

Recibiendo el valor de nombre: Juan

Process finished with exit code 0

Ejemplo

 entrada_datos_consola.py ×

```
1  #entrada de datos por consola
2  nombre = input("Introduce tu nombre: ")
3  print(f"Recibiendo el valor de nombre: {nombre}")
4
```

```
C:\Cursos\Python\Entrada_Datos_Consola\py
Introduce tu nombre: Aaron
Recibiendo el valor de nombre: Aaron
```

```
Process finished with exit code 0
```

Ejemplo

 entrada_datos_consola.py ×

```
1  #entrada de datos por consola
2  nombre = input("Introduce tu nombre: ")
3  print(f"Recibiendo el valor de nombre: {nombre}")
4
5  # Pedir la edad del usuario (entre o como cadena y lo convertimos en numero
6  edad=int(input(f"Introduce tu edad:"))
7  print(f"Tu edad, es: {edad}, y en año tendras {edad + 1}")
8
```

! 2 ✓ 11 ^

C:\Cursos\Python\Entrada_Datos_Consola\py

Introduce tu nombre: **Aaron**

Recibiendo el valor de nombre: Aaron

Introduce tu edad: **34**

Tu edad, es: 34, y en año tendras 35

Sistema de empleados

Crea un programa para solicitar la información de un empleado, introduciendo los datos por consola.

- Nombre empleado.
- Edad del empleado (convertir a entero).
- Salario del empleado (convertir a flotante).
- Es jefe de departamento (Si/No).

Sistema de empleados



sistemas_empleados.py

```
1 print("***Sistema de emepleados***")
2 nombre_empleado= input("Ingrese nombre del empleado: ")
3 edad_empleado = int(input("Ingrese la edad del empleado: "))
4 salario_empleado = float(input("Ingrese el salario del empleado: "))
5 #tipo boleano una cadena nula da False sino da un True
6 es_jefe_departamento = input("Es jefe del departamento (Si/No)")
7 #vamos a convertir a un tipo bool la variable es_jefe_departamento
8 es_jefe_departamento = es_jefe_departamento.lower() == "si"
9 #imprimir los valores del empelado
10 print("\nDatos del empleado")
11 print(f"Nombre: {nombre_empleado}")
12 print(f"Edad empleado: {edad_empleado}")
13 print(f"Salario del empleado: {salario_empleado:.2f}")
14 print(f"Es jefe del departamento: {es_jefe_departamento}")
```



Sistema de empleados

```
C:\Cursos\Python\Entrada_Datos_Consola\pythonProject
```

```
***Sistema de empleados***
```

```
Ingrese nombre del empleado: Aaron
```

```
Ingrese la edad del empleado: 34
```

```
Ingrese el salario del empleado: 4500
```

```
Es jefe del departamento (Si/No)si
```

```
Datos del empleado
```

```
Nombre: Aaron
```

```
Edad empleado: 34
```

```
Salario del empleado: 4500.00
```

```
Es jefe del departamento: True
```

```
Process finished with exit code 0
```


Receta de cocina


Crear un programa para solicitar algunos valores importantes para una receta de cocina.

Los valores que debe introducir el usuarios son:

- Nombre de la receta
- Ingredientes
- Tiempo de preparación (en minutos)
- Dificultad(“Fácil, Media, Alta”)

Mandar a imprimir la receta.

Receta de cocina

 receta_cocina.py ×

```
1 print("***Recetas de Cocina**")
2 nombre_receta= input("Ingresa el nombre de la receta: ")
3 ingredientes=input("Ingresa los ingredientes: ")
4 tiempo_preparacion= int(input("Ingresa el tiempo de preparacion (min): "))
5 dificultad_preparacion=input("Ingresa la dificultad: ")
6 print("-----")
7 print(f"Nombre de la receta: {nombre_receta}")
8 print(f"ingredientes: {ingredientes}")
9 print(f"Tiempo de preparacion: {tiempo_preparacion}")
10 print(f"Dificultad de preparacion: {dificultad_preparacion}")
```

! 5 ✓ 23

Receta de cocina

```
C:\Cursos\Python\Entrada_Datos_Consola\pythonProject\.venv\Scr
```

```
***Recetas de Cocina***
```

```
Ingresa el nombre de la receta: Tallarinada
```

```
Ingresa los ingredientes: fideos,tomate,carne,sal y pimienta
```

```
Ingresa el tiempo de preparacion (min): 25
```

```
Ingresa la dificultad: Media
```

```
-----
```

```
Nombre de la receta: Tallarinada
```

```
ingredientes: fideos,tomate,carne,sal y pimienta
```

```
Tiempo de preparacion: 25
```

```
Dificultad depreparacion: Media
```

Generar Valores aleatorios


La función `randint()`, que es parte del modulo “random”, nos permite generar números aleatorios.

`randint(a,b)` devuelve un numero aleatorio entre a y b, incluyendo estos valores.

Es necesario importar en primer lugar el modulo random antes de usar la función `randint()` .

Para importar un modulo, usamos la sintaxis “import random”

Generar Valores aleatorios

 valores_aleatorios.py ×


```
1  # Valores aleatorios con la funcion randint
2  #tendiramos que especificar el nombre del modulo y la funcion
3  #import random
4  # Lo mas recomendable
5  from random import randint
6  #Generar numero aleatorio entre 1 y 10
7  numero =randint(1,10) # el primer valor no puede ser mayor al segundo
8  print(f"Numero aleatorio entre 1 y 10: {numero}")
9
```

! 7 ✓

```
C:\Cursos\Python\Entrada_Datos_Consola
Numero aleatorio entre 1 y 10: 5
```

```
Process finished with exit code 0
```

Generar Valores aleatorios

 valores_aleatorios.py ×

```
1  # Valores aleatorios con la funcion randint
2  #tendriamos que especificar el nombre del modulo y la funcio..
3  import random
4  # Lo mas recomendable
5  #from random import randint
6  #Generar numero aleatorio entre 1 y 10
7  numero =random.randint(1,10) # el primer valor no puede ser mayor al segundo
8  print(f"Numero aleatorio entre 1 y 10: {numero}")
9
10 #Simular un dado de 6 caras
11 dado=random.randint(1,6)
12 print(f"Resultado de lanzar el dado: {dado}")
13
```

⚠ 10 ✓ 14 ^

```
C:\Cursos\Python\Entrada_Datos_Consola\pyt
Numero aleatorio entre 1 y 10: 4
Resultado de lanzar el dado: 6
```

Sistema Generador de ID único

Se solicita crear un sistema para generar un ID único para cada persona.

El sistema debe solicitar al usuario:

- Nombre
- Apellido
- Año de nacimiento (YYYY)

Con los datos recibidos el sistema deberá realizar lo siguiente:

- Del valor recibido del nombre, usar solo las dos primeras letras y convertirlas a mayúsculas.
- Del valor de apellido, usar las dos primeras letras y convertirlas a mayúsculas.
- Del valor de año, tomar los dos últimos dígitos.

Sistema Generador de ID único

Además el sistema deberá generar un valor aleatorio de 4 dígitos, con ayuda de la función randint.

Finalmente, con los datos obtenidos generar un ID único uniendo los valores como sigue:

Ejemplo:

Nombre → Juan → JU

Apellido → Pérez → PE

Año de nacimiento → 1985 → 85

Valor aleatorio → 8952

Resultado ID único: JUPE858952

Sistema Generador de ID único

```
from random import randint
print("***Sistema generador de ID unico***")
nombre = input("Cual es tu nombre? ")
apellido = input("Cual es tu apellido? ")
anho_nacimiento = input("Cual es tu año de nacimiento (YYYY)? ")
#Normalizar los valores
#sin espacios antes y despues, mayusculas, subcadena 0 a 1
nombre_2 = nombre.strip().upper()[0:2]
apellido_2 = apellido.strip().upper()[0:2]
#tambien puede ser [2:4]
anho_nacimiento_2 = anho_nacimiento.strip()[2:]
# valor aleatorio
aleatorio = randint(1000,9999)
#Generamos el ID unico
id_unico = f"{nombre_2}{apellido_2}{anho_nacimiento_2}{aleatorio}"

print(f"""\nHola {nombre}
Tu nuevo numero de identificacion (ID) generado por el sistema es:
{id_unico}
Felicidades!""")
```


Sistema Generador de EMAIL

Se solicita crear una nueva versión del sistema generador de email. Para generar un email se debe solicitar:

Nombres → Juan Carlos

Apellido → Pérez → Gamarra Fernández

Nombre de empresa → Python Americana

Extensión dominio → .com.py

El resultado debe ser:

juan.carlos.gamarra.fernandez@pythonamericana.com.py

Sistema Generador de EMAIL

```
C:\Cursos\Python\Entrada_Datos_Consola\pythonProject\.venv\Scripts'
```

```
***Sistema generador de email***
```

```
Cuales son tus nombres Juan Carlos
```

```
Cuales son tus apellidos Gamarra Fernandez
```

```
Nombre de tu empresa Python Americana
```

```
Extension del dominio de tu empresa? .com.py
```


```
Tu nuevo email generado por es sistema es:
```

```
    juan.carlos.gamarra.fernandez@pythonamericana.com.py
```

```
    Felicidades!!!
```

```
Process finished with exit code 0
```

Sistema Generador de EMAIL

 generador_email.py ×

```
1 print("***Sistema generador de email***")
2 nombres=input("Cuales son tus nombres ")
3 apellidos=input("Cuales son tus apellidos ")
4 empresa=input("Nombre de tu empresa ")
5 extension=input("Extension del dominio de tu empresa? ")
6 #Normalizamos los valores
7 nombres= nombres.strip().lower().replace(" ", ".")
8 apellidos = apellidos.strip().lower().replace(" ", ".")
9 empresa =empresa.strip().lower().replace(" ", "")
10 extension = extension.strip().lower().replace(" ", "")
11 #Generar el email
12 email=f"{nombres}.{apellidos}@{empresa}{extension}"
13 print(f"""
14 Tu nuevo email generado por es sistema es:
15     {email}
16     Felicidades!!!
17 """)
```