

IBM SkillsBuild & SkillUp Online Presenta

Programa en Inteligencia Artificial



Agenda

- Python con IA
- Conceptos básicos de Python
- Fundamentos de programación en Python
- Laboratorio práctico de Python
- NumPy y Pandas



Why Python for AI?

Popularity

Python es el lenguaje más popular en IA y aprendizaje automático, utilizado por los mejores investigadores y empresas como Google y Facebook.

Ease of Use

Python tiene una sintaxis simple y es amigable para principiantes, lo que hace que sea relativamente fácil de aprender y usarlo en tareas de IA.

Rich Libraries

Python ofrece una amplia colección de bibliotecas diseñadas específicamente para IA, incluidas NumPy, Pandas, Scikit-learn, TensorFlow y Keras.

Conceptos básicos de Python

Tipos de datos en Python

- Enteros (Integer)
- Booleanos (Boolean)
- Cadenas (Strings)
- Listas (List)
- Tuplas (Tuples)
- Conjuntos (Sets)
- Diccionario (Dictionary)

Conceptos Python

Tipos de datos numéricos

- Python admite tres tipos de datos numéricos: **int** es la abreviatura de entero. Los números enteros no tienen ninguna parte decimal.
- Cuando los números tienen una parte decimal, como en 3,14 o 2,5, se denominan **flotantes**.
- Los números **complejos** se representan mediante una clase compleja. Se especifican como (parte real) + (parte imaginaria) j. Por ejemplo: 2+3j

Tipo de dato Booleano

Python también tiene un tipo de datos bool que representa valores booleanos. Solo puede tomar dos valores: **verdadero o falso (1 o 0)**.

String Data Types

Una cadena es una colección de caracteres. En Python, se denomina “str”.

```
In [1]: type("I am learning Python!")
```

```
Out[1]: str
```

```
In [2]: type(5)
```

```
Out[2]: int
```

```
In [3]: type(2456769790)
```

```
Out[3]: int
```

```
In [4]: type(3.4)
```

```
Out[4]: float
```

```
In [5]: type(True)
```

```
Out[5]: bool
```

Python Basics

Lista (List)

- Una lista es una colección de objetos. Los objetos pueden ser int, str, float, bool o incluso uno de los objetos de la colección. Cada objeto puede ser de un tipo diferente.
- La lista se crea con corchetes alrededor de la colección, con cada elemento de la colección separado por una coma.
- Las listas son mutables. Puedes cambiar su tamaño o forma original. Para ello, puedes agregar elementos a la lista, eliminar elementos de la lista e insertar elementos en la posición deseada de la lista.

Lista = [1, "dos", 3.5, True, 1, "dos"]

Python Basics

Tuplas (Tuple)

- Una tupla también es una colección de diversos tipos de objetos. Los objetos pueden ser **int**, **str**, **float**, **bool** o incluso uno de los objetos de la colección.
- La tupla se crea con paréntesis alrededor de la colección, con cada elemento de la colección separado por una coma.
- Las tuplas son inmutables. No se pueden modificar de su forma original. Son muy útiles cuando se desea crear un objeto de colección que no se desea que se modifique durante la vida útil de la aplicación.

Tupla = ("Lunes", "Martes", "Miércoles", "Jueves", "Viernes", "Sábado", "Domingo")

Python Basics

Conjunto (Set)

- Conjunto es también colección de diversos tipos de objetos. Los objetos pueden ser int, str, float, bool o tupla.
- Se crea un conjunto con llaves alrededor de la colección, con cada elemento de la colección separado por una coma.
- Los conjuntos son mutables. Puedes cambiarlos desde el formulario original. Agrega elementos, actualiza elementos, elimina elementos del conjunto. Pero los elementos de un conjunto tienen que ser inmutables. Es por eso que no puedes tener una lista o un conjunto como elemento dentro de un conjunto.

Set = {1,"dos", 3.5, True}

Python Basics

Diccionario (Dictionary)

- Dict es un objeto de colección especial que mantiene **pares clave-valor**. La clave es un identificador único y el valor es un objeto. Los objetos pueden ser **int, str, bool, float** o cualquier otro objeto. Puede haber más de una copia del mismo objeto. Pero **las claves tienen que ser inmutables** y tienen que ser únicas. Por lo general, se utilizan objetos str como claves. No puede haber dos objetos con la misma clave.
- Los diccionarios son mutables. Puedes cambiar su forma original. Puedes agregar nuevos pares clave-valor, actualizar el valor de una clave o eliminar un par clave-valor del diccionario.

Diccionario: {"a":"This is first object","b":1,"c":2.5,"d":True,"e":[1,2,3],"f):(4,5,6)}

Funciones en Python

Tipos de funciones

Built-in Functions: Funciones integradas en Python. Están disponibles para su uso inmediato. Hay varias funciones integradas disponibles en Python. **Print(), len(), Int()**

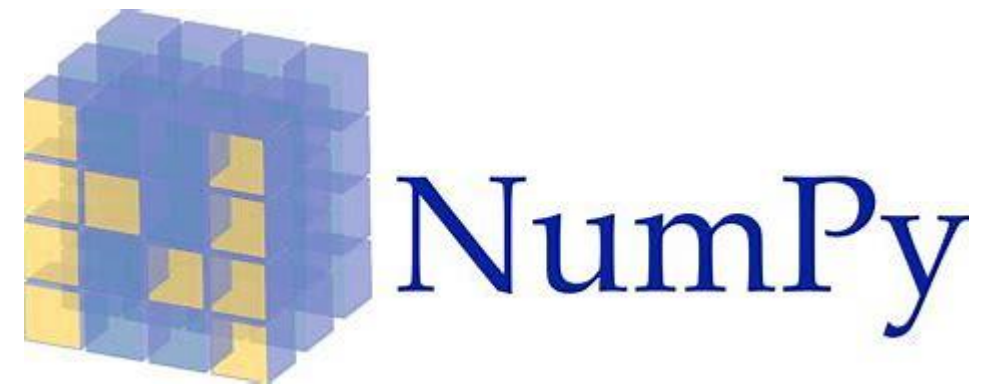
User-defined Functions: Funciones que son definidas por los usuarios.

```
def add(a,b):  
    sum = a+b  
    return sum  
  
num1 = 10  
num2 = 20  
  
print("The sum is", add(num1, num2))  
  
The sum is 30
```

NumPy Library

Numpy significa Python numérico. Proporciona muchos métodos útiles que se pueden utilizar para el procesamiento de datos, especialmente cuando los datos están en forma de una matriz (array) numérica unidimensional o bidimensional. Este paquete también se instala de forma predeterminada, pero deberá importarlo para usarlo en sus cuadernos de Python.

```
import numpy as np
```



Pandas Library

Pandas es la biblioteca principal de datos tabulares de Python. Todos los datos que se pueden estructurar en formato tabular utilizarán Pandas para procesarlos en Python. Proporciona estructuras de datos rápidas, flexibles y expresivas diseñadas para que trabajar con datos estructurados (tabulares, multidimensionales, potencialmente heterogéneos) y de series temporales sea fácil e intuitivo.

El paquete Pandas no requiere instalación. Se instala de forma predeterminada con la instalación de Anaconda. Pero deberá importar Pandas para usarlo en un notebook.



```
import pandas as pd
```

Activity: Hands-On Python LAB

[Python Basics.ipynb - Colab \(google.com\)](#)

Gracias!

Para más información, email a
ibmskillsbuild.eu@skillup.online

