

Introduction to NumPy and Pandas

Session 12







Introduction to NumPy

In this activity, we are going to learn about NumPy libraries of Python





NumPy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

Some of the important features of NumPy are:

- **1. N-Dimensional Array:** NumPy provides support for arrays of different shapes and dimensions, including 1-D arrays (vectors), 2-D arrays (matrices), and N-dimensional arrays.
- **1. Mathematical functions: NumPy** provides a large collection of mathematical functions that can be applied to arrays, including basic operations (addition, subtraction, multiplication, etc.), linear algebra operations, statistical functions, and more.
- **2. Broadcasting**: Broadcasting allows arrays of different shapes to be used together in operations, even if they are not of the same shape.
- **3. Indexing and slicing**: NumPy arrays can be indexed and sliced just like any other Python object, providing a convenient way to extract sub-arrays or modify individual elements.
- **4. Vectorization**: NumPy allows you to perform operations on arrays element-wise, without having to write loops. This is a great way to speed up computations compared to using a for loop in pure Python.



import numpy as np arr= np.array([1, 2, 3, 4, 5])

import pandas as pd df= pd.read_csv('data.csv')

shapes and N-dimensional arrays. matical functions that can be ultiplication, etc.), linear algebra





Introduction to Pandas

In this activity, we are going to learn about Pandas libraries of Python







Pandas

Pandas is a library for the Python programming language, providing fast, flexible, and expressive data structures designed to make working with structured data easy and intuitive.

Some of the important features of Pandas are:

Pandas provides two main data structures, Series and DataFrame. Series is a one-dimensional labelled array and DataFrame is a two-dimensional labeled data structure.

- **1. Data Structures**: In Pandas, both Series and DataFrame can be sliced and indexed using the row and column labels.
- **2. Indexing and Slicing**: Pandas provides convenient functions to handle missing data, including filling in missing values or dropping rows/columns with missing values.
- **3. Handling Missing Data**: Pandas provides various functions to clean data, including converting data types, renaming columns, and removing duplicates.
- 4. Grouping and Aggregating: Pandas provides a convenient way to group and aggregate data based on one or multiple columns.
- 5. Merging and joining: Pandas provides several functions to merge and join data from different sources, including merging two data frames, joining data frames on a common column, and merging data frames using an outer join, inner join, and more.





Activity: Basic of NumPy

Let us code to understand the applications of the NumPy library and create a new NumPy array.

- We will first import the required libraries like time and NumPy.
- Furthermore, we will then create a NumPy 1-D array with the help of np.array() function.
- We will then again initialise a NumPy 2-D array with the help of np.array() function where inside it, there will be two lists nested.
- Furthermore, we will then create a NumPy 1-D array with the help of np.array() function.
- We will then again initialise a numpy 2-D array with the help of np.array() function where inside it, there will be two lists nested.





Activity: Basic of NumPy

- We will then use the sprite.say() method and access the specific elements of the array . The syntax for the same is : one_dim[row][column]
- The one_dim.shape and two_dim.shape method can be used to print the shape of both the one-dimensional and two-dimensional array as we defined.
- The one_dim.size and two_dim.size method will be used to get the size of both the arrays that we have defined earlier.
- Note that after each step we will be writing "time.sleep(2)" method for Tobi to wait before saying another response. Tobi will wait for 2 seconds before stating another answer.





Activity: Basic of NumPy

#importing numpy								
sprite = Sprite("Tobi")								
import numpy as np								
import time								
#Creating a One-Dimensional array								
one_dim=np.array([1,2,3,4,5,6,7,8,9])								
print("One Dimensional array using numpy: \n", one_dim)								
#Creating a Two Dimensional array								
two_dim=np.array([[1,2,3], [4,5,6], [7,8,9]])								
print("Two Dimensional array using numpy: \n", two_dim)								
#Accessing Elements								
sprite.say("The first element of the 1-D array: ")								
time.sleep(2)								
<pre>sprite.say(one_dim[0])</pre>								
time.sleep(2)								
sprite.say("The second element of the first row in the 2-D array: ")								
time.sleep(2)								
<pre>sprite.say(two_dim[0][1])</pre>								
time.sleep(2)								

#shape of the array
sprite.say("Shape of 1-D array: ")
time.sleep(3)
sprite.say(one_dim.shape)
time.sleep(3)
sprite.say("Shape of 2-D array:")
time.sleep(2)
sprite.say(two_dim.shape)
time.sleep(2)
#size of the array
sprite.say("Size of 1-D array: ")
time.sleep(2)
sprite.say(one_dim.size)
time.sleep(2)
<pre>sprite.say("Size of 2-D array: ")</pre>
time.sleep(2)
sprite.say(two_dim.size)







Output Arrays



9

Basic of Pandas

Let us code to create a very basic code of Python in Pictoblox Interface to make a DataFrame using the Pandas Library.

- Importing required libraries: The libraries numpy and pandas are \bullet imported and given the abbreviated names np and pd respectively.
- Creating a dictionary of values: A dictionary data is created with three • keys 'Name', 'Age', and 'Country'. Each key is associated with a list of values.
- Initializing a pandas data frame: A pandas data frame is created from \bullet the data dictionary using the pd.DataFrame() function. The resulting data frame is stored in the df variable.





Basic of Pandas

- Creating a data frame using a numpy array: A numpy array arr is created with ${\color{black}\bullet}$ three rows and three columns. A list of column names columns and a list of row names index are also created. The pd.DataFrame() function is used to create a pandas data frame from the numpy array, the column names, and the row names. The resulting data frame is stored in the df2 variable.
- Finally, we will use the print function to print the DataFrame. \bullet





Basic of Pandas(Final Code)

import numpy as np

import pandas as pd

#Making a dictionary of values

data={'Name':['Tom', 'Jerry', 'Bob', 'Jacob'],

'Age':['14', '12', '15', '19'],

'Country':['India', 'USA', 'UK', 'Canada']}

#Initializing a pandas DataFrame

df=pd.DataFrame(data)

print(df)

#Creating a DataFrame using a numpy array

arr=np.array([[1,2,3], [4,5,6], [7,8,9]])

columns=['A', 'B', 'C']

index=['R1', 'R2', 'R3']

df2=pd.DataFrame(arr, columns=columns, index=index)

print(df2)

Pic	to	Bl	0	8	File	Edit	Tutorials	Board	Connect
Elocks				¢٩	ython		Costumes	(1) S	ounds
>>> 1	Term	ninal			Log	4 ⁻⁹ 5	Serial Monitor		
ë >>		Na	me	Age	Coun	try			
⁸ >> (т	om	14	In	día			
🗳 >> - 1	L	Jerry		12		USA			
🗳 >> 💈	2	Bob		15		UK			
ë >> 3	3	Jacob		19	Can	ada			
ë >>		А	B	C					
🗳 >> F	1.5	1	2	3					
🗳 >> F	22	4	5	6					
🗳 >> 🛛 F	3.3	7	8	9					
⁸ >>									











