TEST 4 - PYTHON

**MULTIPLE CHOICE QUESTION**

1. a) Numerical python

2. b) panel data analysis

3. c) both a and b

4. d) All of the mentioned

5. a) Syntax Error

6. a) 2

7. a) [2.3.4]

8. c) Both A and B

9. a) By passing a python list or tuples to the np.array() function

10. b) Float64

**SHORT ANSWERS**

1.Advantages of Numpy

* NumPy arrays are much faster than nested Python lists for mathematical operations.
* NumPy arrays are more memory efficient than nested Python lists.
* NumPy arrays are easier to use than nested Python lists for many tasks

2.Numpy

Numpy stands for **numerical python.** It is a Python library that adds support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It is the fundamental package for scientific computing in Python.

3. Pandas

Pandas is a Python library for data analysis. It provides a number of tools for working with tabular data, including data frames, series, and functions for manipulating and analyzing data.

Advantages of Pandas.

* + Fast and efficient
  + Flexible
  + Easy to use and understand.

4.Indicies of an array

* The **Numpy.where()** function takes a condition as input and returns two arrays: one with the indices where the condition is true, and one with the values at those indices.

5. Series from dictionary

The pandas.Series() function takes a dictionary as input and returns a Pandas Series object. The keys of the dictionary become the indices of the Series object, and the values of the dictionary become the values of the Series object.

6.adding column to a pandas.

Method 1: Using the assign() method

Ex :

import pandas as pd

df = pd.DataFrame({'Name': ['John Doe', 'Jane Doe', 'John Smith'], 'Age': [30, 25, 40]})

df = df.assign(Country='United States')

print(df)

Method 2: Using the insert() method

Ex:

import pandas as pd

df = pd.DataFrame({'Name': ['John Doe', 'Jane Doe', 'John Smith'], 'Age': [30, 25, 40]})

df = df.insert(1, 'Country', 'United States')

print(df)

7. Series and data frame

|  |  |  |
| --- | --- | --- |
| Feature | Series | Data Frame |
| Dimensionality | One-dimensional | Two-dimensional |
| Indexing | Indexed by values | Indexed by values and columns |
| Data types | Can only contain data of the same data type | Can contain data of different data types |
| Methods | Limited set of methods | Large set of methods |

8. Series as vertically

The numpy.vstack() function takes a sequence of arrays as input and returns a new array that is stacked vertically.

import numpy as np

series1 = np.array([1, 2, 3, 4, 5])

series2 = np.array([6, 7, 8, 9, 10])

stacked\_series = np.vstack([series1, series2])

print(stacked\_series)

OUTPUT

[[ 1 2 3 4 5]

[ 6 7 8 9 10]]

9. Dataframe.

The isnull () method returns a Boolean Series indicating whether each value in the Dataframe is missing.

import numpy as np

import pandas as pd

df = pd.DataFrame({'Name': ['John Doe', 'Jane Doe', 'John Smith'], 'Age': [30, np.nan, 40]})

missing\_values = df.isnull()

print(missing\_values)

OUTPUT:

Name False True False

Age False True False

dtype: bool

10. reshape an array

The numpy.reshape() method is the most common way to reshape an array in Python

Ex:

import numpy as np

array = np.array([1, 2, 3, 4, 5, 6])

new\_shape = (3, 2)

reshaped\_array = np.reshape(array, new\_shape)

print(reshaped\_array)

OUTPUT:

[[1 2]

[3 4]

[5 6]]