**Assignment -1**

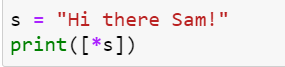
Python Programming

|  |  |
| --- | --- |
| Assignment Date | 09 September 2022 |
| Student Name | Mr. Ajith |
| Student Roll Number | 512219104001 |
| Maximum Marks | 2 Marks |

**Question-1:**

s = "Hi there Sam!"

|  |
| --- |
| **Solution:** |
|  | s = “Hi there Sam” |
|  | Print([\*s]) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |





**Question-2:**

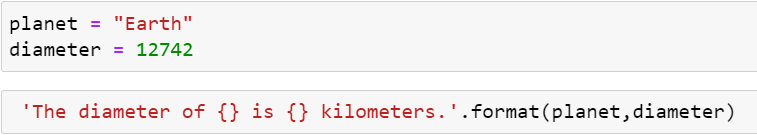
Use .format() to print the following string.

planet = "Earth"

diameter = 12742

Output should be: The diameter of Earth is 12742 kilometers.

|  |
| --- |
| **Solution:** |
|  | Planet=”Earth” |
|  | Diameter = 12742 |
|  | ‘The diameter of {} is {} kilometers.’.format(planet,diameter) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |



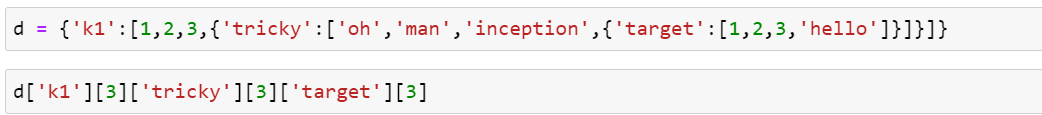


**Question-3:**

In this nest dictionary grab the word "hello"

d = {'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}

|  |
| --- |
| Solution:  d={'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}  d['k1'][3]['tricky'][3]['target'][3] |



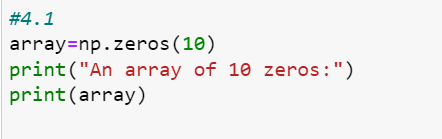


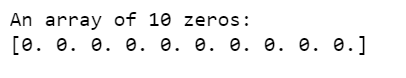
**Numpy**

**Question-4.1:**

## Create an array of 10 zeros?

|  |
| --- |
| Solution:  import numpy as np  array=np.zeros(10)  print("An array of 10 zeros:")  print(array) |

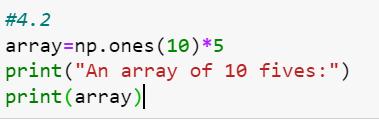


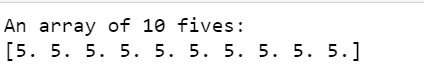


**Question-4.2:**

## Create an array of 10 fives?

|  |
| --- |
| **Solution:**    **array=np.ones(10)\*5**  **print("An array of 10 fives:")**  **print(array)** |

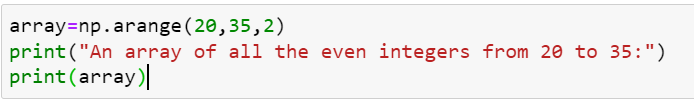


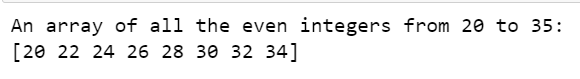


**Question-5:**

**Create an array of all the even integers from 20 to 35**

|  |
| --- |
| Solution:  array=np.arange(20,35,2)  print("An array of all the even integers from 20 to 35:")  print(array) |

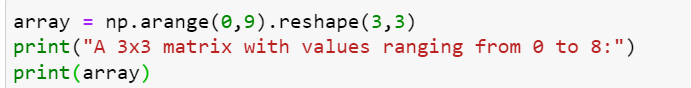


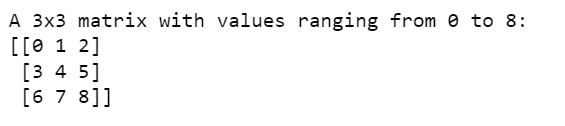


**Question-6:**

## Create a 3x3 matrix with values ranging from 0 to 8

|  |
| --- |
| **Sloution:**  **array = np.arange(0,9).reshape(3,3)**  **print("A 3x3 matrix with values ranging from 0 to 8:")**  **print(array)** |



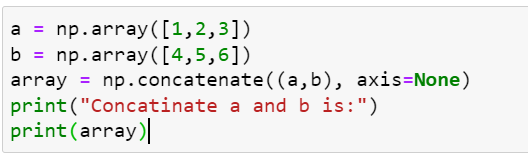


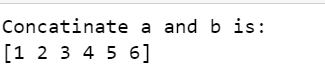
**Question-7:**

## Concatinate a and b

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

|  |
| --- |
| Solution:  a = np.array([1,2,3])  b = np.array([4,5,6])  array = np.concatenate((a,b), axis=None)  print("Concatinate a and b is:")  print(array) |



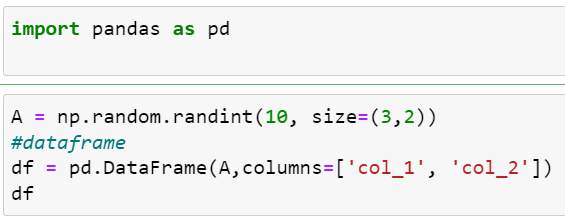


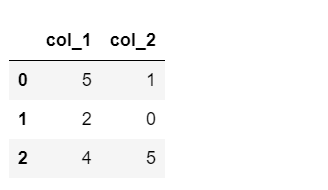
# Pandas

**Question-8:**

## Create a dataframe with 3 rows and 2 columns

|  |
| --- |
| Solution:import pandas as pdA = np.random.randint(10, size=(3,2))#dataframedf = pd.DataFrame(A,columns=['col\_1', 'col\_2'])df |

****



**Question-9:**

## Generate the series of dates from 1st Jan, 2023 to 10th Feb, 2023

|  |
| --- |
| Solution:  # calling DataFrame constructor  df = pd.DataFrame()    df['time'] = pd.date\_range(start="1/1/2023",end="2/10/2023", freq ='24H')  df['year'] = df['time'].dt.year  df['month'] = df['time'].dt.month  df['day'] = df['time'].dt.day    # Show six rows  df.head(len(df["time"])) |





**Question-10:**

## Create 2D list to DataFrame

## lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]

|  |
| --- |
| Solution:lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]df = pd.DataFrame(lists,columns = ['id','name','age'])print("2D list:")print(df) |

