UIT2502---Data Analytics and Visualization

Ex 1 a: Basic Numpy Exercise

NAME: Shyam Saran P

REG NO: 3122235002123

1. Import numpy as np and see the version

```
import numpy as np
print(np.__version__)
```

2. How to create a 1D array?

```
arr = np.array([1, 2, 3, 4, 5])
arr = np.arange(5)  # Creates array([0, 1, 2, 3, 4])
```

3. How to create a boolean array?

```
bool_arr = np.array([True, False, True])
arr = np.array([1, 2, 3, 4])
bool_arr = arr > 2
```

4. How to extract items that satisfy a given condition from 1D array?

```
arr = np.array([1, 2, 3, 4, 5])
condition = arr > 3
result = arr[condition]  # array([4, 5])
result = arr[arr > 3]
```

5. How to replace items that satisfy a condition with another value in numpy array?

```
arr = np.array([1, 2, 3, 4, 5])
arr[arr > 3] = 0
```

6. How to replace items that satisfy a condition without affecting the original array?

```
arr = np.arange(6)  # array([0, 1, 2, 3, 4, 5])
reshaped = arr.reshape(2, 3)
```

7. How to reshape an array?

```
arr = np.arange(6)  # array([0, 1, 2, 3, 4, 5])
reshaped = arr.reshape(2, 3)
```

8. How to stack two arrays vertically?

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

9. How to stack two arrays horizontally?

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

10. How to generate custom sequences in numpy without hardcoding?

```
linear = np.arange(0, 10, 2)  # array([0, 2, 4, 6, 8])
spaced = np.linspace(0, 1, 5)  # array([0., 0.25, 0.5, 0.75, 1.])
geom = np.geomspace(1, 8, 4)
```

11. How to get the common items between two python numpy arrays?

```
a = np.array([1, 2, 3, 4])
b = np.array([3, 4, 5, 6])
common = np.intersect1d(a, b)
```

12. How to remove from one array those items that exist in another?

```
a = np.array([1, 2, 3, 4])
b = np.array([3, 4, 5, 6])
result = np.setdiff1d(a, b)
```

13. How to get the positions where elements of two arrays match?

```
a = np.array([1, 2, 3, 4])
b = np.array([1, 3, 3, 5])
matches = np.where(a == b)
```

14. How to extract all numbers between a given range from a numpy array?

```
arr = np.array([1, 2, 3, 4, 5, 6])
result = arr[(arr >= 2) & (arr <= 5)]
```

15. How to make a python function that handles scalars to work on numpy arrays?

```
def my_func(x):
return x**2 + 1
vectorized_func = np.vectorize(my_func)
arr = np.array([1, 2, 3])
result = vectorized_func(arr)  # array([2, 5, 10])
```

16. How to swap two columns in a 2d numpy array?

```
arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
arr[:, [0, 2]] = arr[:, [2, 0]]  # Swaps columns 0 and 2
```

17. How to swap two rows in a 2d numpy array?

```
arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
arr[[0, 2]] = arr[[2, 0]]  # Swaps rows 0 and 2
```

18. How to reverse the rows of a 2D array?

```
arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
reversed_rows = arr[::-1]  # Reverse row order
```

19. How to reverse the columns of a 2D array?

```
arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
reversed_cols = arr[:, ::-1]  # Reverse column order
```

20. How to create a 2D array containing random floats between 5 and 10?

```
random_floats = np.random.uniform(5, 10, size=(3, 4))  # 3x4 array with values
```

21. How to print only 3 decimal places in python numpy array?

```
arr = np.random.rand(3, 3)
np.set_printoptions(precision=3)
print(arr)
```

22. How to pretty print a numpy array by suppressing the scientific notation (like 1e10)?

```
arr = np.array([1.23e-5, 1.23e10])
np.set_printoptions(suppress=True)  # Suppress scientific notation
print(arr)
```

23. How to limit the number of items printed in output of numpy array?

```
large_arr = np.arange(1000)
np.set_printoptions(threshold=10)  # Show only first and last few elements
```

```
print(large_arr)
```

24. How to print the full numpy array without truncating.

```
large_arr = np.arange(1000)

np.set_printoptions(threshold=np.inf)  # Show all elements

print(large_arr)
```

25. How to import a dataset with numbers and texts keeping the text intact in python numpy?

```
# Using numpy's genfromtxt with dtype=None for automatic type detection

data = np.genfromtxt('data.csv', delimiter=',', dtype=None, encoding='utf-8')

# Or better for mixed data: use pandas

import pandas as pd

data = pd.read_csv('data.csv').values
```

26. How to extract a particular column from 1D array of tuples?

```
tuple_arr = np.array([(1, 'a'), (2, 'b'), (3, 'c')], dtype=[('num', 'i4'), ('char', 'U1')])

col = tuple_arr['char']  # Gets the 'char' column: array(['a', 'b', 'c'], dtype='<U1')
```

27. How to convert a 1d array of tuples to a 2d numpy array?

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

# Method 1:

arr_2d = np.vstack(tuple_arr)

# Method 2:

arr_2d = np.array([list(x) for x in tuple_arr])
```

28. How to compute the mean, median, standard deviation of a numpy array?

```
arr = np.array([1, 2, 3, 4, 5])
mean = np.mean(arr)
median = np.median(arr)
std_dev = np.std(arr)
```

29. How to normalize an array so the values range exactly between 0 and 1?

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

normalized = (arr - arr.min()) / (arr.max() - arr.min())
```

30. How to compute the softmax score?

```
def softmax(x):
```

```
e_x = np.exp(x - np.max(x))  # For numerical stability
return e_x / e_x.sum()
arr = np.array([1.0, 2.0, 3.0])
softmax_scores = softmax(arr)
```

31. How to find the percentile scores of a numpy array?

```
arr = np.array([1, 2, 3, 4, 5])
percentiles = np.percentile(arr, [25, 50, 75])  # 25th, 50th, 75th percentiles
```

32. How to insert values at random positions in an array?

```
arr = np.zeros(10)
indices = np.random.choice(10, size=3, replace=False)  # 3 random positions
arr[indices] = 1  # Insert 1s at random positions
```

33. How to find the position of missing values in numpy array?

```
arr = np.array([1, 2, np.nan, 4, np.nan])
missing_positions = np.where(np.isnan(arr))[0]
```

34. How to filter a numpy array based on two or more conditions?

```
arr = np.array([1, 2, 3, 4, 5])
filtered = arr[(arr > 2) & (arr < 5)]  # Elements >2 AND <5
```

35. How to drop rows that contain a missing value from a numpy array?

```
import numpy as np
arr = np.array([[1, 2], [3, np.nan], [7, 6]])
clean_arr = arr[~np.isnan(arr).any(axis=1)]
```

36. How to find the correlation between two columns of a numpy array?

```
arr = np.array([[1, 2], [3, 4], [5, 6]])
correlation = np.corrcoef(arr[:, 0], arr[:, 1])[0, 1]
```

37. How to get the second largest value of an array when grouped by another array?

```
import numpy as np
import pandas as pd
a = np.array(['a', 'a', 'b', 'b', 'c', 'c'])
```

```
b = np.array([10, 20, 30, 25, 50, 40])
df = pd.DataFrame({'group': a, 'value': b})
second_largest = df.groupby('group')['value'].apply(lambda x: x.nlargest(2).iloc[-1])
```

38. How to sort a 2D array by a column

```
arr = np.array([[1, 3], [4, 2], [2, 1]])
sorted_arr = arr[arr[:, 1].argsort()]
```

39. How to find the most frequent value in a numpy array?

```
arr = np.array([1, 2, 3, 2, 2, 3, 1])
values, counts = np.unique(arr, return_counts=True)
most_frequent = values[np.argmax(counts)]
```

40. How to find the position of the first occurrence of a value greater than a given value?

```
arr = np.array([1, 3, 7, 2, 5])
pos = np.argmax(arr > 4)  # finds first occurrence > 4
```

41. How to replace all values greater than a given value to a given cutoff?

```
arr = np.array([1, 3, 7, 2, 5])
arr[arr > 4] = 4  # replaces values > 4 with 4
```

42. How to get the positions of top n values from a numpy array?

```
arr = np.array([10, 20, 5, 30, 40])
n = 3
top_n_indices = arr.argsort()[-n:][::-1]  # positions of top 3 values
```

43. How to compute the row wise counts of all possible values in an array?

```
arr = np.array([[1, 2, 1], [3, 3, 1]])
unique_vals = np.unique(arr)
counts = np.array([(row[:, None] == unique_vals).sum(axis=0) for row in arr])
```

44. How to convert an array of arrays into a flat 1d array?

```
arr = np.array([[1, 2], [3, 4]])
flat_arr = arr.flatten()
```
