Name: Taha Hasnain, Section: 4,

1. **What will be the output of the following code?**

```
x = [1, 2, 3, 4]
y = x
y.append(5)
print(x)
```

a) `[1, 2, 3, 4]`
b) `[1, 2, 3, 4, 5]`
c) `[1, 2, 3, 4, 5, 5]`
d) `None of the above`

Question 1 (Ans : b)

2. **Which of the following is True for Python sets?** a) Sets are ordered collections.
   b) Sets allow duplicate elements.
   c) Sets are mutable.
   d) Sets can have lists as elements.

Question 2 : (Ans: d)

3. **What will be the output of the following code?**

```
def func(a, b=[]):
    b.append(a)
    return b

print(func(1))
print(func(2))
```

a) `[1] [2]`
b) `[1] [1, 2]`
c) `[1, 1] [2, 2]`
d) `None of the above`

Question 3: (Ans : a)

4. **Which of the following will not result in an error?**

```
a) my_dict = {[1, 2, 3]: "val"}
b) my_dict = {(1, 2, 3): "val"}
c) my_dict = {{1, 2, 3}: "val"}
d) my_dict = {{1: "a"}: "val"}
```

Question 4: (Ans: d)

5. **Which of the following statements is correct?** a) `input()` returns an integer by default.
   b) `input()` always returns a string.
   c) `input()` accepts multiple arguments.
   d) `input()` converts input to a float.

Question 5: (Ans: b)

6. **What is the output of the following code?**

```
a = 10
def func():
    print(a)
    a = 5
func()
```

   a) `10`
   b) `5`

Question 6: (Ans: a)

7. **Which method is used to convert an object to its string representation?** a) `__str__()`
   b) `__repr__()`
   c) `__format__()`
   d) `__convert__()`

Question 7: (Ans: a)

8. **What is the output of the following code?**

```
a = [1, 2, 3]
b = [1, 2, 3]
print(a is b)
```

   a) `True`
   b) `False`
   c) `None`
   d) `SyntaxError`

Question 8: (Ans: b)

9. **Which of the following is a correct way to define a dictionary?** a) `d = {"one": 1, "two": 2}`
   b) `d = ["one": 1, "two": 2]`
   c) `d = ({"one": 1, "two": 2})`
   d) `d = {"one"-1, "two"-2}`

Question 9: (Ans: a)

10. **What does the pass keyword do in Python?** a) Exits a function
    b) Does nothing; acts as a placeholder
    c) Terminates the loop
    d) Returns `None`

Question 10: (Ans: b)

Pandas

11. **How can you read a CSV file in Pandas?** a) `pandas.read_csv("file.csv")`
    b) `pandas.read("file.csv")`
    c) `pandas.load_csv("file.csv")`
    d) `pandas.csv("file.csv")`

Question 11: (Ans: a)

12. **Which method is used to get the first n rows of a DataFrame?** a) `.tail(n)`
    b) `.top(n)`
    c) `.first(n)`
    d) `.head(n)`

Question 12: (Ans: d)

13. **What will be the output of the following code?**

```
python

df = pd.DataFrame({
    'A': [1, 2, 3],
    'B': [4, 5, 6]
})
print(df.loc[0, 'A'])
```

a) `1`
b) `4`

Question 13: (Ans: a)

14. **How can you add a new column to an existing DataFrame?** a)
    `df.add_column('C', [7, 8, 9])`
    b) `df['C'] = [7, 8, 9]`
    c) `df.new_column('C', [7, 8, 9])`
    d) `df.column('C', [7, 8, 9])`

Question 14: (Ans: d)

15. **Which of the following is True for the `dropna()` method in Pandas?** a) Removes all rows with NaN values by default
    b) Removes all columns with NaN values by default
    c) Replaces NaN values with zeros
    d) Removes duplicate rows

Question 15: (Ans: b)

16. **What does the `apply()` method do in Pandas?** a) Applies a function element-wise to a DataFrame
    b) Applies a function along an axis of the DataFrame
    c) Applies a function to each row
    d) None of the above

Question 16: (Ans: c)

17. **How can you group a DataFrame by a column in Pandas?** a)
    `df.groupby(by='column_name')`
    b) `df.sort(by='column_name')`
    c) `df.group('column_name')`
    d) `df.aggregate('column_name')`

Question 17: (Ans: a)

18. **What will be the result of the following code?**

```
df = pd.DataFrame({
    'A': [1, 2, 3],
    'B': [4, 5, 6]
})
print(df.iloc[1])
```

a) The first row of the DataFrame
b) The second row of the DataFrame
c) The last row of the DataFrame
d) An error

Question: 18: (Ans:b)

19. **Which method in Pandas is used to concatenate two DataFrames vertically?** a) `pd.concat([df1, df2], axis=1)`
b) `pd.concat([df1, df2], axis=0)`
c) `pd.join([df1, df2], axis=0)`
d) `pd.merge([df1, df2], axis=0)`

Question: 19: (Ans: c)

20. **What is the purpose of the `describe()` method in Pandas?** a) To provide summary statistics of a DataFrame
b) To describe the structure of a DataFrame
c) To return column names
d) To describe missing data

Question 20: (Ans: b)

NumPy

21. **Which of the following creates a NumPy array with values from 0 to 9?** a) `np.array(10)`

b) `np.arange(10)`
c) `np.arange(0, 9)`
d) `np.linspace(0, 9)`

Question 21: (Ans: a)

22. **What is the result of the following code?**

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

a) `[3, 4, 5, 6]`
b) `[3, 4, 5]`
c) `[1, 2, 3, 4, 2]`
d) `Error`

Question: 22 (Ans: a)

23. **How can you create a 3x3 matrix of zeros using NumPy?** a) `np.zeros(3)`
b) `np.zeros([3, 3])`
c) `np.zeros((3, 3))`
d) `np.zeros([3])`

Question: 23: (Ans: b)

24. **What does the `shape` attribute of a NumPy array return?** a) The data type of the elements in the array
b) The number of elements in the array
c) The dimensions of the array
d) The size of each dimension

Question: 24: (Ans: c)

25. **How can you find the maximum value in a NumPy array?** a) `np.max(arr)`
b) `arr.max()`
c) Both a) and b)
d) None of the above

Question: 25: (Ans: a)

26. **What will be the output of the following code?**

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

a) `[1, 2, 3]`
b) `[2, 3, 4]`
c) `[2, 3, 4, 5]`
d) `[1, 2, 3, 4]`

Question: 26: (Ans: b)

27. **Which of the following methods is used to change the shape of a NumPy array?**
a) `reshape()`
b) `resize()`
c) `reformat()`
d) `reindex()`

Question: 27: (Ans: a)

28. **How can you multiply two NumPy arrays element-wise?**

a) `np.dot(arr1, arr2)`
b) `np.matmul(arr1, arr2)`
c) `arr1 * arr2`
d) `arr1 ** arr2`

Question: 28: (Ans: a)

29. **What is the output of the following code?**

```
arr = np.array([10, 20, 30, 40])
print(arr > 25)
```

a) `[10, 20, 30, 40]`
b) `[True, True, True, True]`
c) `[False, False, True, True]`
d) `[25, 25, 30, 40]`

Question: 29: (Ans: c)

30. **What does the `flatten()` method do in NumPy?** a) Converts a multi-dimensional array into a 1D array
b) Flattens only the first dimension of an array
c) Converts a 1D array into a 2D array
d) Transposes the array

Question: 30: (Ans: a)

Programming Tasks

Task 1: Write a program in python to find even and odd numbers between 1 to 100 using loop.

```
for i in range(1, 101):
    if i%2 == 0:
        print(f"{i} is Even")
    else:
        print(f"{i} is Odd")
```

Task2: Write a program to generate two NumPy arrays and find the product and dot product of the matrices.

```
import numpy as np
arr1 = np.array([1, 2, 3, 4])
arr2 = np.array([6, 7, 8, 9])
prod = arr1*arr2
dotProd = arr1.dot(arr2)

print(prod)
print(dotProd)
```

Task3: Read any given datasets (Olx_Car_data, titanic_dataset) using pandas' library and perform following operations

a. Print last 10 rows of the dataset
b. Print the shape of the dataset
c. Find missing values in the dataset
d. Drop the 3rd column of the dataset

```python
import pandas as pd

df = pd.read_csv('./olx_car_dataset_csv.csv')

print(df.tail(10))

print(df.shape)

missing_values = df.isnull().sum()

print(missing_values)

df = df.drop(df.columns[2], axis=1)

print(df)
```