

Program 7 and 8

September 15, 2023

P7:Implement NumPy features Create an Employee table using a Structured array and answer the questions.

```
(Emp_Id,Last_Name,First_Name,Gender,Title)
```

```
(1000, "Torbati", "Yolanda", "F", "Programmer")
(1001, "Kleinn", "Joel", "M", "Programmer")
(1002, "Ginsburg", "Laura", "F", "President")
(1003, "Cox", "Jennifer", "F", "Programmer")
(1005, "Ziada", "Mauri", "M", "Product Designer")
(1006, "Keyser", "Cara", "F", "Account Executive")
(1010, "Smith", "Roxie", "M", "Programmer")
(1011, "Nelson", "Robert", "M", "Programmer")
(1012, "Sachsen", "Lars", "M", "Support Technician")
(1013, "Shannon", "Don", "M", "Product Designer")
```

1. How many Male employees are in a company?
2. Display the details of employees whose Last_Name starts with S.
3. Sort the Female Employee details in descending order based on First_Name.
4. Extract 1D array and reshape it into 2D array.
5. Extract the below matrix using Boolean and Fancy indexing.

```
[ ]: import numpy as np

dt = np.dtype([('emp_id', int), ('last_name', 'U20'), ('first_name', 'U20'),
               ('gender', 'U1'), ('title', 'U20')])

employee_data = np.array([
    (1000, "Torbati", "Yolanda", "F", "Programmer"),
    (1001, "Kleinn", "Joel", "M", "Programmer"),
    (1002, "Ginsburg", "Laura", "F", "President"),
    (1003, "Cox", "Jennifer", "F", "Programmer"),
    (1005, "Ziada", "Mauri", "M", "Product Designer"),
    (1006, "Keyser", "Cara", "F", "Account Executive"),
    (1010, "Smith", "Roxie", "M", "Programmer"),
    (1011, "Nelson", "Robert", "M", "Programmer"),
    (1012, "Sachsen", "Lars", "M", "Support Technician"),
    (1013, "Shannon", "Don", "M", "Product Designer")
], dtype=dt)
```

```

def print_employee_details(employee):
    print(f"Employee ID: {employee['emp_id']}, Last Name:␣
    ↳{employee['last_name']}, First Name: {employee['first_name']}, Gender:␣
    ↳{employee['gender']}, Title: {employee['title']}")

male_count = np.sum(employee_data['gender'] == 'M')
print("Q1. Number of male employees:", male_count)

matching_employees_S = employee_data[np.char.
    ↳startswith(employee_data['last_name'], 'S')]
print("Q2. Employees whose Last_Name starts with 'S':")
for employee in matching_employees_S:
    print_employee_details(employee)

female_employees = employee_data[employee_data['gender'] == 'F']
sorted_female_employees = np.sort(female_employees, order='first_name')[::-1]
print("Q3. Female Employees sorted in descending order:")
for employee in sorted_female_employees:
    print_employee_details(employee)

emp_id_1d = employee_data['emp_id']
emp_id_2d = emp_id_1d.reshape(-1, 1)
print("Q4. 1D array (emp_id_1d):")
print(emp_id_1d)
print("\n2D array (emp_id_2d):")
print(emp_id_2d)

selected_matrix = employee_data[(employee_data['emp_id'] >= 1002) &␣
    ↳(employee_data['emp_id'] <= 1012)]
print("Q5. Matrix for emp_id between 1002 and 1012:")
for employee in selected_matrix:
    print_employee_details(employee)

```

Q1. Number of male employees: 6

Q2. Employees whose Last_Name starts with 'S':

Employee ID: 1010, Last Name: Smith, First Name: Roxie, Gender: M, Title:
Programmer

Employee ID: 1012, Last Name: Sachsen, First Name: Lars, Gender: M, Title:
Support Technician

Employee ID: 1013, Last Name: Shannon, First Name: Don, Gender: M, Title:
Product Designer

Q3. Female Employees sorted in descending order:

Employee ID: 1000, Last Name: Torbati, First Name: Yolanda, Gender: F, Title:
Programmer

Employee ID: 1002, Last Name: Ginsburg, First Name: Laura, Gender: F, Title:
President

Employee ID: 1003, Last Name: Cox, First Name: Jennifer, Gender: F, Title: Programmer

Employee ID: 1006, Last Name: Keyser, First Name: Cara, Gender: F, Title: Account Executive

Q4. 1D array (emp_id_1d):

```
[1000 1001 1002 1003 1005 1006 1010 1011 1012 1013]
```

2D array (emp_id_2d):

```
[[1000]  
 [1001]  
 [1002]  
 [1003]  
 [1005]  
 [1006]  
 [1010]  
 [1011]  
 [1012]  
 [1013]]
```

Q5. Matrix for emp_id between 1002 and 1012:

Employee ID: 1002, Last Name: Ginsburg, First Name: Laura, Gender: F, Title: President

Employee ID: 1003, Last Name: Cox, First Name: Jennifer, Gender: F, Title: Programmer

Employee ID: 1005, Last Name: Ziada, First Name: Mauri, Gender: M, Title: Product Designer

Employee ID: 1006, Last Name: Keyser, First Name: Cara, Gender: F, Title: Account Executive

Employee ID: 1010, Last Name: Smith, First Name: Roxie, Gender: M, Title: Programmer

Employee ID: 1011, Last Name: Nelson, First Name: Robert, Gender: M, Title: Programmer

Employee ID: 1012, Last Name: Sachsen, First Name: Lars, Gender: M, Title: Support Technician

P8: Demonstrate Pandas with its operations 1. Import the domain dataset that you identified with missing values and perform the following. For each output, give the interpretation with respect to the imported dataset. 2. Read the csv file and create and understand the data frame using describe(), shape, info(). 3. Find if any missing values (null values) are in the data, handle all the rows with missing data in four different ways (delete, replace, fill, dropna), and print the data frame. 4. Filter based on any column using groupby(). 5. Select 20 samples randomly and Create a data frame with Hierarchical Index

```
[ ]: import pandas as pd  
df=pd.read_csv('C:/Users/Yash/Downloads/place/placement.csv')  
print(df.head(5))  
print(df.describe())  
print(df.shape)  
print(df.info())
```

	Age	Gender	Stream	Internships	CGPA	Hostel	\
0	22	Male	Electronics And Communication	NaN	8.0	1.0	
1	21	Female	Computer Science	0.0	7.0	1.0	
2	22	Female	Information Technology	1.0	NaN	0.0	
3	21	Male	NaN	0.0	8.0	0.0	
4	22	Male	Mechanical	0.0	8.0	1.0	

	HistoryOfBacklogs	PlacedOrNot
0	1.0	1.0
1	1.0	1.0
2	0.0	1.0
3	1.0	1.0
4	0.0	1.0

	Age	Internships	CGPA	Hostel	HistoryOfBacklogs	\
count	2966.000000	2965.000000	2965.000000	2965.000000	2965.000000	
mean	21.485840	0.703541	7.074199	0.269140	0.192243	
std	1.324933	0.740302	0.967710	0.443588	0.394129	
min	19.000000	0.000000	5.000000	0.000000	0.000000	
25%	21.000000	0.000000	6.000000	0.000000	0.000000	
50%	21.000000	1.000000	7.000000	0.000000	0.000000	
75%	22.000000	1.000000	8.000000	1.000000	0.000000	
max	30.000000	3.000000	9.000000	1.000000	1.000000	

	PlacedOrNot
count	2965.000000
mean	0.552445
std	0.497326
min	0.000000
25%	0.000000
50%	1.000000
75%	1.000000
max	1.000000

(2966, 8)

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 2966 entries, 0 to 2965

Data columns (total 8 columns):

#	Column	Non-Null Count	Dtype
0	Age	2966 non-null	int64
1	Gender	2965 non-null	object
2	Stream	2965 non-null	object
3	Internships	2965 non-null	float64
4	CGPA	2965 non-null	float64
5	Hostel	2965 non-null	float64
6	HistoryOfBacklogs	2965 non-null	float64
7	PlacedOrNot	2965 non-null	float64

dtypes: float64(5), int64(1), object(2)

memory usage: 185.5+ KB

None

```
[ ]: print(df.isnull().sum())
```

```
Age                0
Gender             1
Stream             1
Internships        1
CGPA               1
Hostel             1
HistoryOfBacklogs  1
PlacedOrNot        1
dtype: int64
```

```
[ ]: df_drop=df.dropna()
print(df)
```

	Age	Gender	Stream	Internships	CGPA	Hostel	\
0	22	Male	Electronics And Communication	NaN	8.0	1.0	
1	21	Female	Computer Science	0.0	7.0	1.0	
2	22	Female	Information Technology	1.0	NaN	0.0	
3	21	Male	NaN	0.0	8.0	0.0	
4	22	Male	Mechanical	0.0	8.0	1.0	
...	
2961	23	Male	Information Technology	0.0	7.0	0.0	
2962	23	Male	Mechanical	1.0	7.0	1.0	
2963	22	Male	Information Technology	1.0	7.0	NaN	
2964	22	Male	Computer Science	1.0	7.0	0.0	
2965	23	Male	Civil	0.0	8.0	0.0	

	HistoryOfBacklogs	PlacedOrNot
0	1.0	1.0
1	1.0	1.0
2	0.0	1.0
3	1.0	1.0
4	0.0	1.0
...
2961	NaN	0.0
2962	0.0	0.0
2963	0.0	0.0
2964	0.0	0.0
2965	0.0	0.0

[2966 rows x 8 columns]

```
[ ]: df_ffill=df.ffill()
df_bfill=df.bfill()
print(df)
```

	Age	Gender	Stream	Internships	CGPA	Hostel	\
0	22	Male	Electronics And Communication	NaN	8.0	1.0	
1	21	Female	Computer Science	0.0	7.0	1.0	
2	22	Female	Information Technology	1.0	NaN	0.0	
3	21	Male	NaN	0.0	8.0	0.0	
4	22	Male	Mechanical	0.0	8.0	1.0	
...	
2961	23	Male	Information Technology	0.0	7.0	0.0	
2962	23	Male	Mechanical	1.0	7.0	1.0	
2963	22	Male	Information Technology	1.0	7.0	NaN	
2964	22	Male	Computer Science	1.0	7.0	0.0	
2965	23	Male	Civil	0.0	8.0	0.0	

	HistoryOfBacklogs	PlacedOrNot
0	1.0	1.0
1	1.0	1.0
2	0.0	1.0
3	1.0	1.0
4	0.0	1.0
...
2961	NaN	0.0
2962	0.0	0.0
2963	0.0	0.0
2964	0.0	0.0
2965	0.0	0.0

[2966 rows x 8 columns]

```
[ ]: grouped_data=df.groupby('Age')
```

```
[ ]: sample1=df.sample(n=20)
hi=sample1.set_index(['Internships', 'Age'])
print("\nDataFrame with hierarchical index for 20 random samples:")
print(hi)
```

DataFrame with hierarchical index for 20 random samples:

	Gender	Stream	CGPA	Hostel	\
Internships Age					
1.0 22	Male	Computer Science	7.0	0.0	
21	Male	Mechanical	7.0	1.0	
2.0 20	Male	Information Technology	6.0	0.0	
22	Male	Civil	6.0	1.0	
1.0 21	Male	Electronics And Communication	7.0	1.0	
21	Male	Civil	7.0	0.0	
2.0 22	Male	Information Technology	6.0	0.0	
1.0 19	Male	Electronics And Communication	6.0	1.0	
23	Female	Mechanical	6.0	1.0	

0.0	21	Male	Mechanical	8.0	0.0
	19	Male	Mechanical	7.0	1.0
	22	Male	Civil	6.0	1.0
	20	Female	Computer Science	6.0	1.0
1.0	21	Male	Information Technology	7.0	1.0
2.0	21	Male	Information Technology	8.0	0.0
0.0	21	Male	NaN	8.0	0.0
	21	Male	Electrical	7.0	0.0
	21	Male	Civil	8.0	1.0
	22	Female	Electrical	7.0	1.0
1.0	20	Male	Information Technology	7.0	0.0

		HistoryOfBacklogs	PlacedOrNot
Internships	Age		
1.0	22	0.0	0.0
	21	0.0	0.0
2.0	20	0.0	1.0
	22	0.0	1.0
1.0	21	1.0	0.0
	21	0.0	0.0
2.0	22	0.0	1.0
1.0	19	0.0	0.0
	23	0.0	0.0
0.0	21	0.0	1.0
	19	0.0	0.0
	22	0.0	0.0
	20	0.0	0.0
1.0	21	0.0	0.0
2.0	21	1.0	1.0
0.0	21	1.0	1.0
	21	0.0	0.0
	21	0.0	1.0
	22	1.0	0.0
1.0	20	0.0	1.0