

```
In [12]: import numpy as np
```

Create a numpy array containing the numbers from 1 to 10, and then reshape it to a 2x5 matrix.

```
In [14]: a1 = np.arange (1,11).reshape(2,5)
print(a1)
print(a1.shape)
print(a1.ndim)
```

```
[[ 1  2  3  4  5]
 [ 6  7  8  9 10]]
(2, 5)
2
```

Create a numpy array containing the numbers from 1 to 20, and then extract the elements between the 5th and 15th index.

```
In [16]: a2 = np.arange (1,21)
print(a2)
print(a2[5:15])
```

```
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
[ 6  7  8  9 10 11 12 13 14 15]
```

Create a Pandas series with the following data: {'apples': 3, 'bananas': 2, 'oranges': 1}. Then, add a new item to the series with the key 'pears' and the value 4.

```
In [18]: import pandas as pd
```

```
In [19]: fruit = pd.Series ({'apples': 3, 'bananas': 2, 'oranges': 1})
print(fruit)
```

```
fruit["pears"] = 4  
print(fruit)
```

```
apples    3  
bananas   2  
oranges   1  
dtype: int64  
apples    3  
bananas   2  
oranges   1  
pears     4  
dtype: int64
```

Create a dataframe with the following columns: name, age, and gender. The dataframe should have 10 rows of data.

```
In [21]: df = pd.DataFrame({  
    'name': ['Alice', 'Bob', 'Charlie', 'David', 'Eve',  
            'Frank', 'Grace', 'Hannah', 'Ian', 'Jane'],  
    'age': [25, 30, 22, 35, 28, 40, 33, 27, 29, 31],  
    'gender': ['Female', 'Male', 'Male', 'Male', 'Female',  
              'Male', 'Female', 'Female', 'Male', 'Female']  
})  
df
```

Out[21]:

	name	age	gender
0	Alice	25	Female
1	Bob	30	Male
2	Charlie	22	Male
3	David	35	Male
4	Eve	28	Female
5	Frank	40	Male
6	Grace	33	Female
7	Hannah	27	Female
8	Ian	29	Male
9	Jane	31	Female

Add a new column to the data frame created in question 1, called occupation. The values for this column should be Programmer, Manager, and Analyst, corresponding to the rows in the dataframe.

```
In [23]: df['occupation'] = ['Programmer', 'Manager', 'Analyst', 'Manager', 'Programmer', 'Analyst', 'Manager', 'Analyst', 'Programmer', 'Manager']
df
```

Out[23]:

	name	age	gender	occupation
0	Alice	25	Female	Programmer
1	Bob	30	Male	Manager
2	Charlie	22	Male	Analyst
3	David	35	Male	Manager
4	Eve	28	Female	Programmer
5	Frank	40	Male	Analyst
6	Grace	33	Female	Manager
7	Hannah	27	Female	Analyst
8	Ian	29	Male	Programmer
9	Jane	31	Female	Analyst

Select the rows of the dataframe where the age is greater than or equal to 30.

```
In [25]: age_condition = df[df['age'] >= 30]
print(age_condition)
```

	name	age	gender	occupation
1	Bob	30	Male	Manager
3	David	35	Male	Manager
5	Frank	40	Male	Analyst
6	Grace	33	Female	Manager
9	Jane	31	Female	Analyst