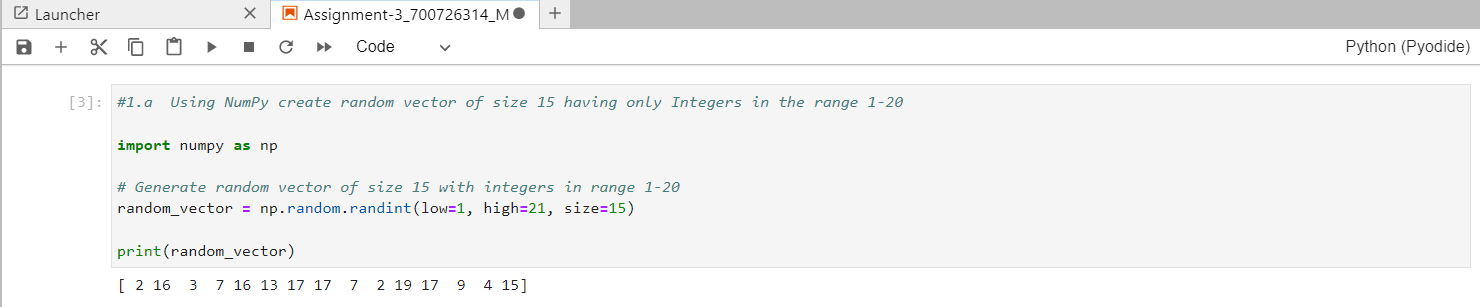
1Q:



Explanation:

* This will create a NumPy array random\_vector with 15 random integers between 1 and 20 (inclusive).
* The low parameter specifies the lower limit of the range,the high parameter specifies the upper limit, and the size parameter specifies the size of the array.

1.1

Text

Description automatically generated with low confidence

* I have used the reshape() method to reshape the NumPy array to a 3 by 5 matrix.
* This will create a new NumPy array matrix\_3x5 with the same values as random\_vector, but reshaped into a 3 by 5 matrix.
* The reshape() method takes the desired shape of the new array as arguments, in this case (3, 5), Note that the total number of elements in the new array must be the same as the original array.

1.2 Text

Description automatically generated

* I have used the shape attribute of the NumPy array to print its shape.
* This will print the shape of the matrix\_3x5 array, which should be (3, 5).
* The shape attribute returns a tuple containing the dimensions of the array, in this case (3, 5).

1.3 Text

Description automatically generated

* use the argmax() method to find the index of the maximum value in each row, and then use NumPy indexing to replace that value with 0.
* This will create a new NumPy array matrix\_3x5 with the same values as the original,but with the maximum value in each row replaced with 0.
* The argmax() method takes the axis parameter to specify which axis to find the maximum along.
* In this case, axis=1 means we want to find the maximum in each row.
* We then use a for loop to iterate over the rows and replace the maximum value with 0 using NumPy indexing.

1.b Graphical user interface, text, application, email

Description automatically generated

* This will create a NumPy array array\_2d of size 4 x 3 with 4-byte integer elements and print its shape (4, 3), type <class 'numpy.ndarray'>, and data type int32.
* Note that we use the dtype parameter to specify the data type of the array elements, in this case np.int32 for 4-byte integers.

1.b Graphical user interface, text, application

Description automatically generated

* use the eig() function from NumPy's linear algebra module numpy.linalg to compute the eigenvalues and right eigenvectors of a square array.This will compute the eigenvalues and right eigenvectors of the given square array and print them.
* The eig() function returns a tuple containing the eigenvalues and eigenvectors. The eigenvalues are stored in an array eigenvalue, and the eigenvectors are stored in a 2-dimensional array eigenvectors where each column represents an eigenvector.

1.c Graphical user interface, text, application, email

Description automatically generated

* we use NumPy's trace() function to compute the sum of the diagonal elements of a given array.
* This will compute the sum of the diagonal elements of the given array using the trace() function and print it.
* The trace() function returns the sum of the diagonal elements of a 2-dimensional array.

1.d Graphical user interface, text, application

Description automatically generated

* use the reshape() function from NumPy to create a new shape for an array without changing its data This will reshape the original array from 3x2 to 2x3 using the reshape() function and store the result in a new variable new\_shape. The original array array is not modified.The reshape() function returns a new array with the specified shape, without changing the data.

2Q: Chart, pie chart

Description automatically generated

* This code will generate a pie chart with the given data, where the "Java" slice is slightly separated from the rest. We can adjust the sizes of the slices and the degree of separation by changing the values of the **sizes** and **explode** lists.
* We can also customize the labels, colors, and other properties of the chart by modifying the parameters passed to the **ax1.pie()** method. Finally, we call **plt.show()** to display the chart in a window.