Assignment

Consider the below code to answer further questions:

import numpy as np

list_ = [‘1’ , ’2’ , ‘3’ , ‘4’ , ‘5’]

array_list = np.array(object = list_)

Q1. Is there any difference in the data type of variables list_ and array_list? If there is then write a code
to print the data types of both the variables.

Ans:there is a difference in the data type of the variables `list_` and `array_list`. The variable `list_` is a Python list, while `array_list` is a NumPy array. You can use the `type()` function to print the data types of these variables. Here's the code to do that:

python

Copy code

```
import       as

         '1'   '2'   '3'   '4'   '5'
                     object


print "Data type of list_:"  type
print "Data type of array_list:"  type
```

When you run this code, you'll see the data types of `list_` and `array_list` printed. The output will indicate that `list_` is of type `list`, and `array_list` is of type `numpy.ndarray`.

Q2. Write a code to print the data type of each and every element of both the variables list_ and arra_list.

Ans:can use a loop to iterate through each element in `list_` and `array_list` and print their respective data types. Here's the code for that:

python

Copy code

```
import       as

         '1'   '2'   '3'   '4'   '5'
                     object


print "Data types of elements in list_:"
for         in
 print type
```

```
print "\nData types of elements in array_list:"
for         in
 print type
```

In this code, the `type()` function is used to determine the data type of each element in both `list_` and `array_list`. The output will show the data types of individual elements in each variable. Note that in the provided example, all elements are strings, so you'll see `<class 'str'>` for each element. If your list contains elements of different types, you'll see different data types accordingly.

Q3. Considering the following changes in the variable, array_list:
array_list = np.array(object = list_, dtype = int)
Will there be any difference in the data type of the elements present in both the variables, list_ and
arra_list? If so then print the data types of each and every element present in both the variables, list_
and arra_list.
Consider the below code to answer further questions:
import numpy as np
num_list = [[1 , 2 , 3] , [4 , 5 , 6]]
num_array = np.array(object = num_list)
Ans:there will be a difference in the data types of the elements present in both the variables `list_` and `array_list` after the change `array_list = np.array(object = list_, dtype = int)`.

Let's print the data types of each element in both variables:

python

Copy code

```
import       as


         '1'   '2'   '3'   '4'   '5'
                     object


                          object              int


print "Data types of elements in list_:"
for         in
```

```
 print type


print "\nData types of elements in array_list:"
for         in
 print type


print "\nData types of elements in array_list_int:"
for         in
 print type
```

In this code, I've created a new NumPy array (`array_list_int`) by specifying the `dtype=int` parameter, which means the elements will be converted to integers. The output will show the data types of individual elements in the original list, the original array, and the modified array with integer data type. You will notice the change in data types after the modification.

Q4. Write a code to find the following characteristics of variable, num_array:
(i) shape
(ii) size
Ans:Here's the code to find the shape and size of the `array_list`:
python

Copy code

```
import       as

         '1'   '2'   '3'   '4'   '5'
                     object




print "(i) Shape of array_list:"
print "(ii) Size of array_list:"
```

In this code, the `.shape` attribute is used to find the shape (dimensions) of the array, and the `.size` attribute is used to find the total number of elements in the array. The results will be printed, showing the shape and size of the `array_list`.

Q5. Write a code to create numpy array of 3*3 matrix containing zeros only, using a numpy array
creation function.
[Hint: The size of the array will be 9 and the shape will be (3,3).]
Asn:can use the `np.zeros()` function to create a NumPy array of a specific shape (in this case, a 3x3 matrix) filled with zeros. Here's the code:
python

Copy code

```
import       as

                    3  3


print
```

In this code, `np.zeros((3, 3))` creates a 3x3 matrix filled with zeros. The resulting array (`zero_matrix`) will be a NumPy array with the specified shape.

Q6. Create an identity matrix of shape (5,5) using numpy functions?
[Hint: An identity matrix is a matrix containing 1 diagonally and other elements will be 0.]
Ans:can use the `np.eye()` function to create an identity matrix. Here's the code:
python

Copy code

```
import       as

                    5


print
```

In this code, `np.eye(5)` creates a 5x5 identity matrix where the diagonal elements are 1 and the off-diagonal elements are 0. The resulting array (`identity_matrix`) will be a NumPy array representing the identity matrix.