# IBM Python Exercise 5: Numpy

1) <u>code:</u>

```
import numpy as np
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
b= np.where(arr%2!=0, -1, arr)
print(b)
```

<u>output:</u>

```
[ 0 -1  2 -1  4 -1  6 -1  8 -1]
```

2) <u>code:</u>

```
import numpy as np
x=np.arange(10)
print(x)
y=np.reshape(x,(-1,5))
print(y)
```

<u>output:</u>

```
[0 1 2 3 4 5 6 7 8 9]

[[0 1 2 3 4]
 [5 6 7 8 9]]
```

3) <u>code:</u>

```
import numpy as np
l = [1, 2, 3, 4]
print("Original List:",l)
a = np.array(l)
print("One-dimensional NumPy array: ",a)
```

<u>output</u>:

```
Original List: [1, 2, 3, 4]
One-dimensional NumPy array:  [1 2 3 4]
```

## 4) code:

```python
import numpy as np

x=np.arange(2,11)

y=np.reshape(x,(-1,3))

print(y)
```

## output:

```
[[ 2  3  4]
 [ 5  6  7]
 [ 8  9 10]]
```

## 5) code:

```python
import numpy as np

x = np.zeros(10)

print(x)

print("Update sixth value to 11")

x[6] = 11

print(x)
```

## output:

```
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
Update sixth value to 11
[ 0.  0.  0.  0.  0.  0. 11.  0.  0.  0.]
```

## 6) code:

```python
import numpy as np

array1 = np.arange(12,38)

print("initial array", str(array1))

res = array1[ : :-1]

print("final array", str(res))
```

## output:

```
initial array [12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
31 32 33 34 35
 36 37]
final array [37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18
17 16 15 14
 13 12]
```

7) code:

import numpy as np

x = [1, 2, 3]

print("Original array:")

print(x)

x = np.append(x, [[4, 5, 6], [7, 8, 9]])

print("After append values to the end of the array:")

print(x)

output:

```
Original array:
[1, 2, 3]
After append values to the end of the array:
[1 2 3 4 5 6 7 8 9]
```

8) code:

import numpy as np

list1 = [1, 2, 3, 4, 5, 6, 7, 8]

print("List to array: ")

print(np.asarray(list1))

tuple1 = ([5, 4, 3], [2, 1, 0], [-1, -2, -3])

print("Tuple to array: ")

print(np.asarray(tuple1))

output:

```
List to array:
[1 2 3 4 5 6 7 8]
Tuple to array:
[[ 5  4  3]
```

```
[ 2   1   0]
[-1  -2  -3]]
```

9) code:

import numpy as np

out_arr = np.random.randint(low = 4, size =(3, 3))

print ("Output 3D Array filled with random integers:     ", out_arr)

output:

```
Output 3D Array filled with random integers:
[[1 2 0]
 [0 3 3]
 [0 1 1]]
```