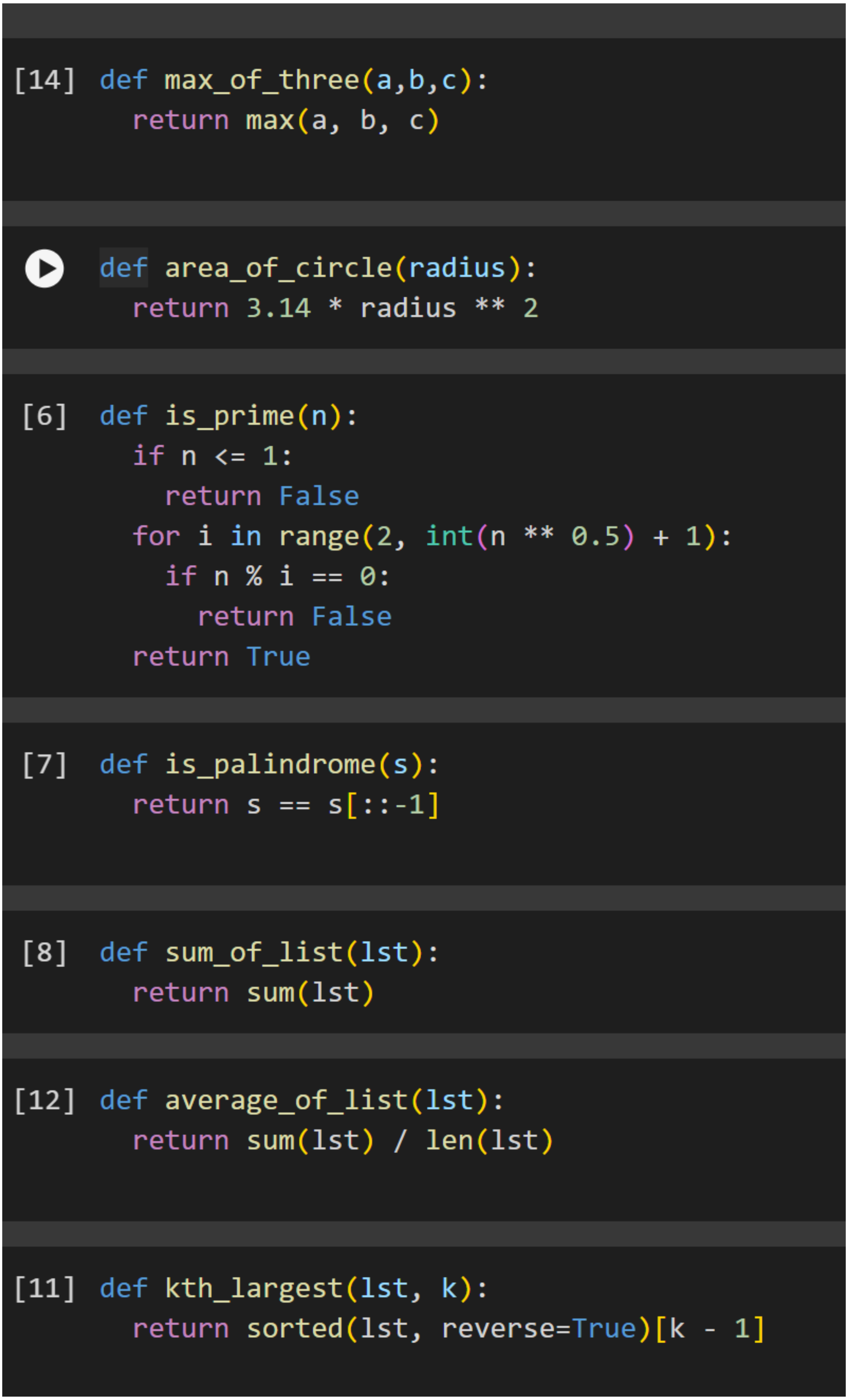
Daily_Task_Function_Numpy

Jenin L



[14]

```
def max_of_three(a,b,c):
  return max(a, b, c)
```

```
def area_of_circle(radius):
  return 3.14 * radius ** 2
```

[6]

```
def is_prime(n):
  if n <= 1:
    return False
  for i in range(2, int(n ** 0.5) + 1):
    if n % i == 0:
      return False
  return True
```

[7]

```
def is_palindrome(s):
  return s == s[::-1]
```

[8]

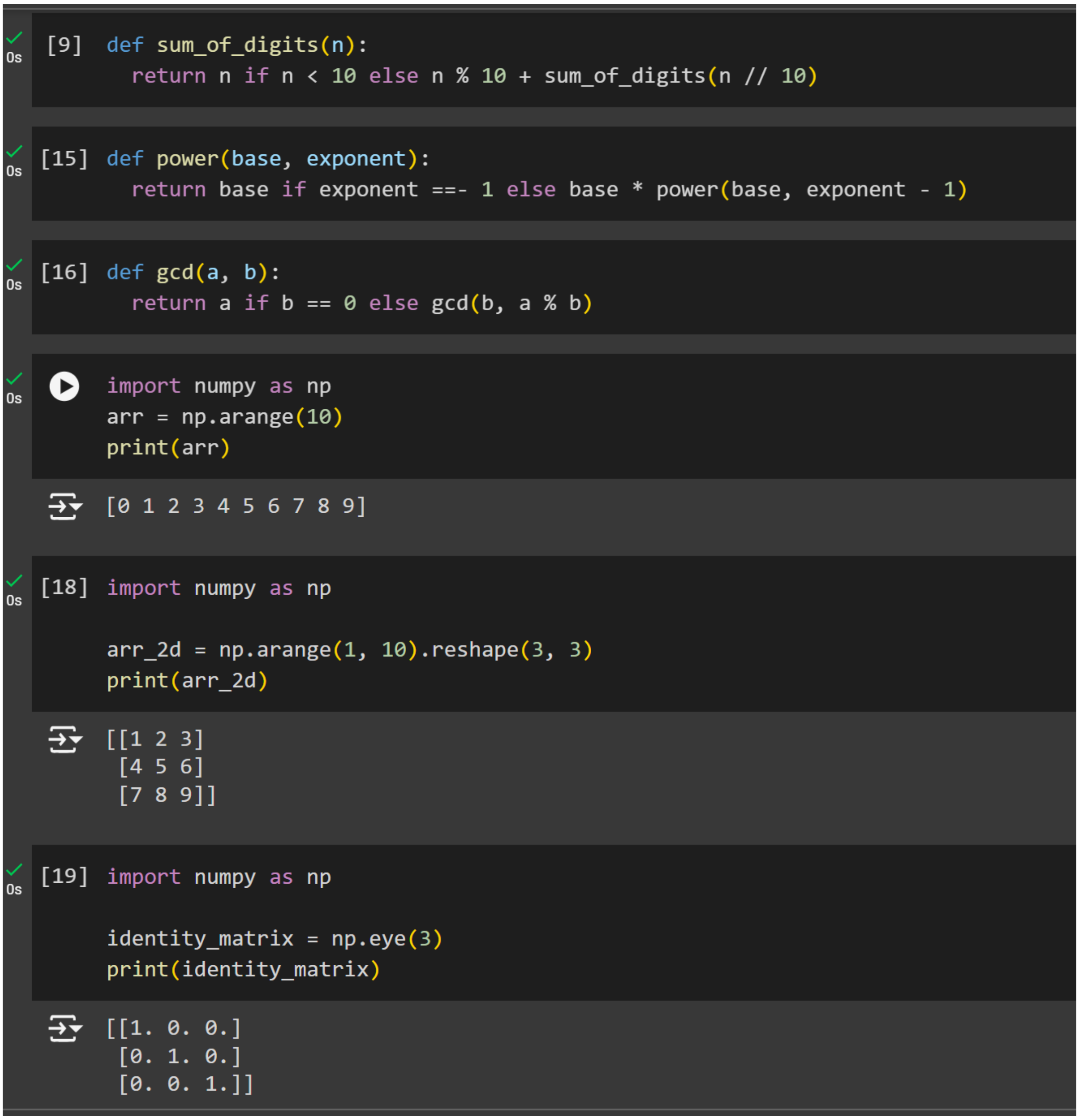
```
def sum_of_list(lst):
  return sum(lst)
```

[12]

```
def average_of_list(lst):
  return sum(lst) / len(lst)
```

[11]

```
def kth_largest(lst, k):
  return sorted(lst, reverse=True)[k - 1]
```



```
def sum_of_digits(n):
  return n if n < 10 else n % 10 + sum_of_digits(n // 10)
```

```
def power(base, exponent):
  return base if exponent ==- 1 else base * power(base, exponent - 1)
```

```
def gcd(a, b):
  return a if b == 0 else gcd(b, a % b)
```

```
import numpy as np
arr = np.arange(10)
print(arr)
```

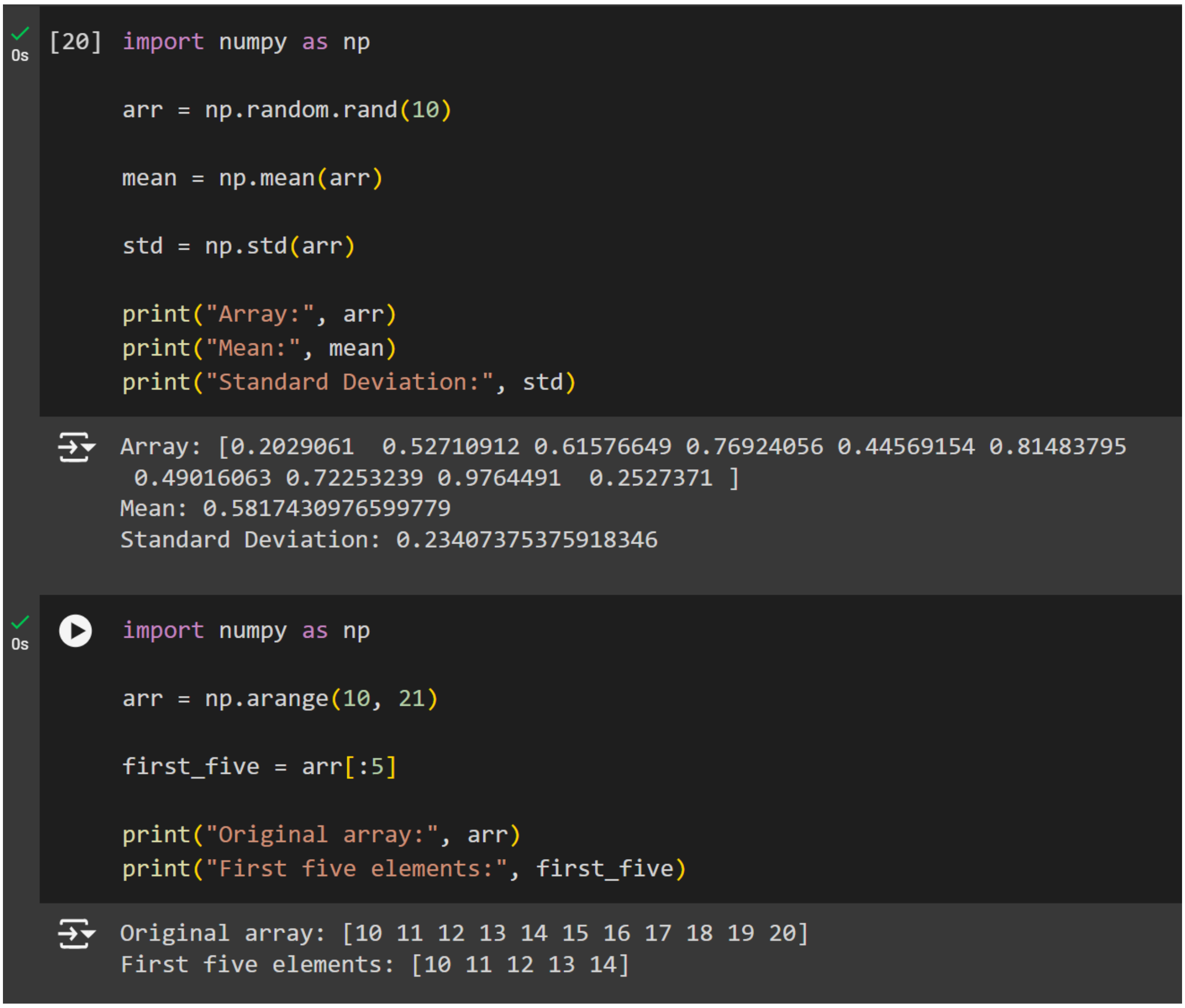
```
[0 1 2 3 4 5 6 7 8 9]
```

```
import numpy as np

arr_2d = np.arange(1, 10).reshape(3, 3)
print(arr_2d)
```

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

```
import numpy as np

identity_matrix = np.eye(3)
print(identity_matrix)
```

```
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
```

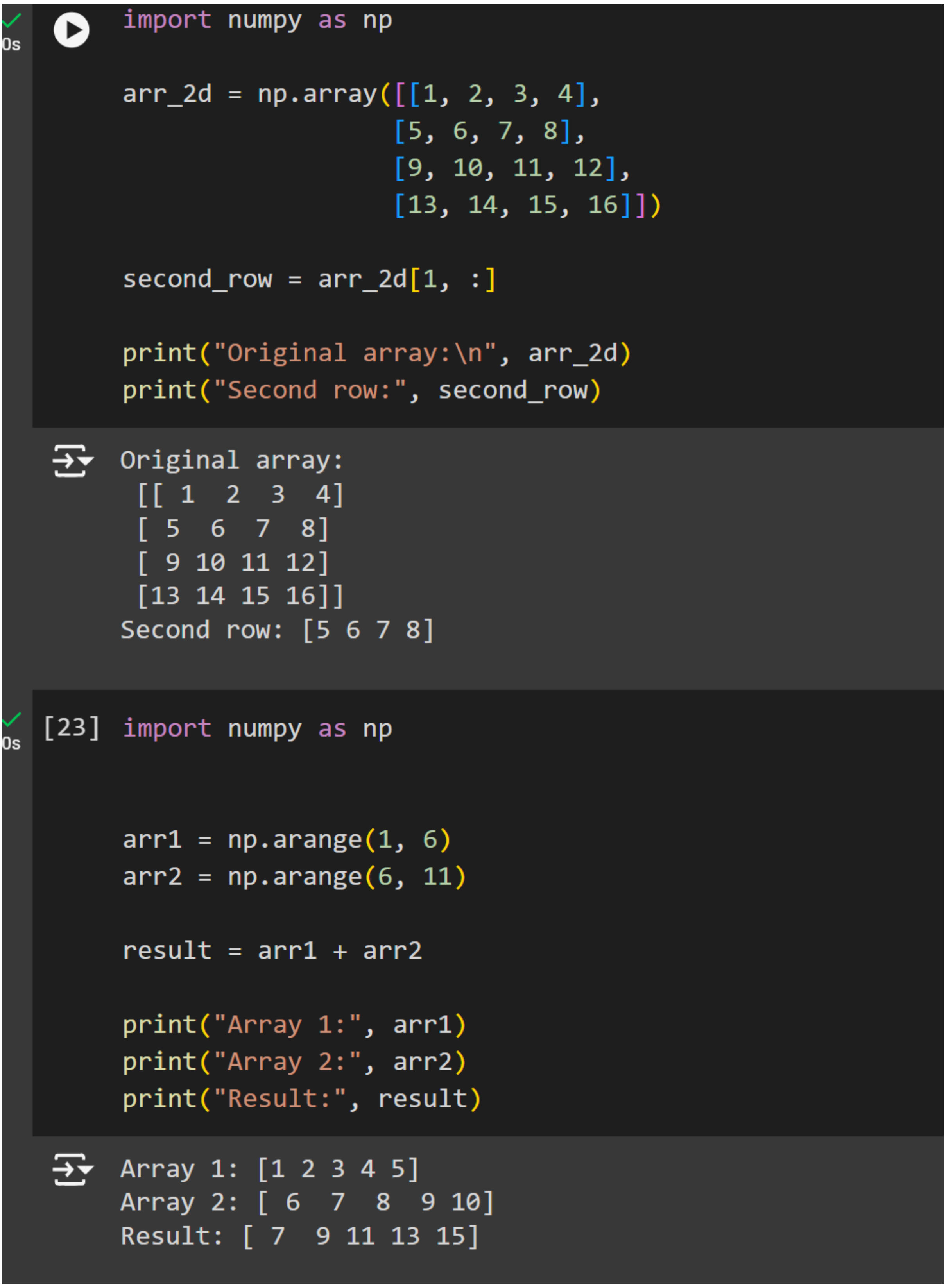


```python
import numpy as np

arr = np.random.rand(10)

mean = np.mean(arr)

std = np.std(arr)

print("Array:", arr)
print("Mean:", mean)
print("Standard Deviation:", std)
```

```
Array: [0.2029061  0.52710912 0.61576649 0.76924056 0.44569154 0.81483795
 0.49016063 0.72253239 0.9764491  0.2527371 ]
Mean: 0.5817430976599779
Standard Deviation: 0.23407375375918346
```

```python
import numpy as np

arr = np.arange(10, 21)

first_five = arr[:5]

print("Original array:", arr)
print("First five elements:", first_five)
```

```
Original array: [10 11 12 13 14 15 16 17 18 19 20]
First five elements: [10 11 12 13 14]
```



```python
import numpy as np

arr_2d = np.array([[1, 2, 3, 4],
                   [5, 6, 7, 8],
                   [9, 10, 11, 12],
                   [13, 14, 15, 16]])

second_row = arr_2d[1, :]

print("Original array:\n", arr_2d)
print("Second row:", second_row)
```

```
Original array:
 [[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]
 [13 14 15 16]]
Second row: [5 6 7 8]
```

```python
import numpy as np


arr1 = np.arange(1, 6)
arr2 = np.arange(6, 11)

result = arr1 + arr2

print("Array 1:", arr1)
print("Array 2:", arr2)
print("Result:", result)
```

```
Array 1: [1 2 3 4 5]
Array 2: [ 6  7  8  9 10]
Result: [ 7  9 11 13 15]
```