

# Uygulamalı Laboratuvar: Python Paketi Oluşturma



## Python Paketi Oluşturma

Gerekli tahmini süre: **30** dakika

### Hedefler

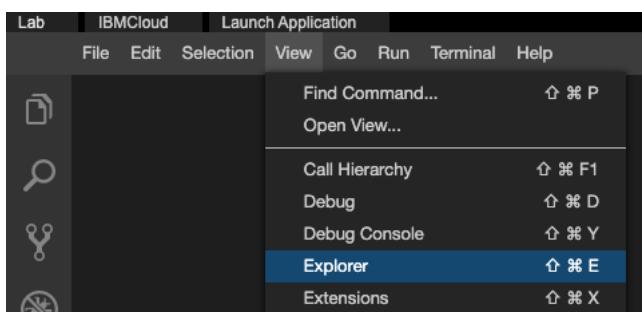
Bu laboratuvar çalışmasında:

- basic adında bir modül oluşturun
- basic modülüne iki fonksiyon ekleyin
- stats adında bir modül oluşturun
- stats modülüne iki fonksiyon ekleyin
- mymath adında bir python paketi oluşturun
- paketin çalıştığını doğrulayın

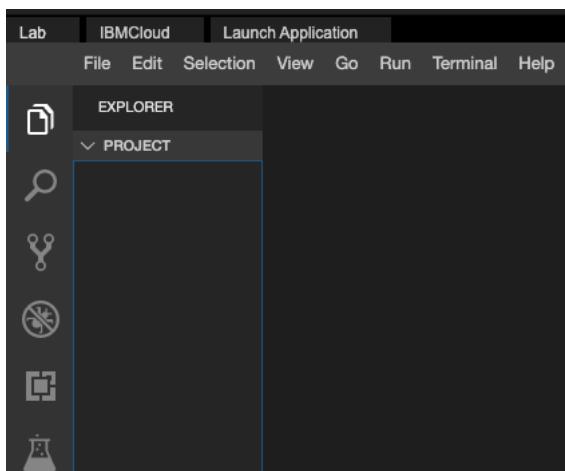
## Laboratuvar

### Paket Oluştur

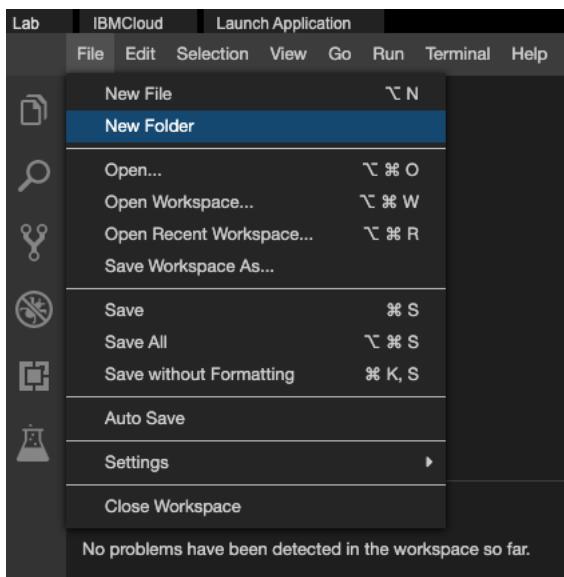
- Sağdaki pencerede **Görünüm** menüsüne tıklayın ve aşağıdaki görüntüde gösterildiği gibi **Gezgin** seçeneğini seçin.



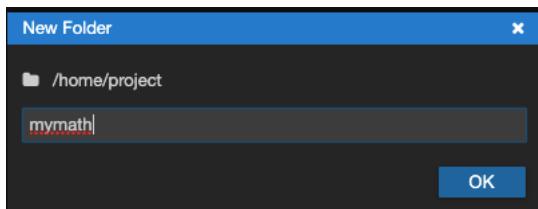
- Artık IDE'niz aşağıdaki görüntüye benzemelidir.



- Sağdaki pencerede **Dosya** menüsüne tıklayın ve aşağıdaki görüntüde gösterildiği gibi **Yeni Klasör** seçeneğini seçin.



- **mymath** yazın ve aşağıdaki görüntüde gösterildiği gibi Tamam'a tıklayın.



## İlk Modülü Oluştur

- basic adında bir python modülü oluşturun

**basic.py** adında bir dosya oluşturun.

Aşağıdaki kodu basic.py dosyasına kopyalayın ve yapıştırın.

```
def square(number):
    """
    This function returns the square of a given number
    """
    return number ** 2
def double(number):
    """
    This function returns twice the value of a given number
    """
    return number * 2
def add(a, b):
    """
    This function returns the sum of given numbers
    """
    return a + b
```

Artık böyle bir ekran görmelisiniz.

```
basic.py ●

1  def square(number):
2      """
3          This function returns the square of a given number
4      """
5      return number ** 2
6
7
8  def double(number):
9      """
10         This function returns twice the value of a given number
11     """
12     return number * 2
13
14 def add(a, b):
15     """
16         This function returns the sum of given numbers
17     """
18     return a + b
19
```

Dosyayı **basic.py** olarak kaydedin.

## İkinci Modülü Oluştur

- stats adında bir modül oluştur

**stats.py** adında bir dosya oluştur.

Aşağıdaki kodu stats.py dosyasına kopyalayıp yapıştırın.

```
def mean(numbers):
    """
    This function returns the mean of the given list of numbers.
    The mean is calculated as the sum of all numbers divided by the count of numbers.
    """
    # Calculate the mean by summing all the numbers and dividing by the length of the list.
    # 'sum(numbers)' computes the total of all numbers in the list.
    # 'len(numbers)' gives the count of numbers in the list.
    # The mean (or average) is the total sum divided by the number of elements.
    return sum(numbers) / len(numbers) # Return the mean value.

def median(numbers):
    """
    This function returns the median of the given list of numbers.
    The median is the middle value when the numbers are sorted.
    If there is an even number of observations, it returns the average of the two middle numbers.
    """
    # Sort the list of numbers in ascending order.
    # Sorting is necessary to find the median, as the median depends on the order of values.
    numbers.sort()
    # Check if the number of elements in the list is even.
    # len(numbers) % 2 == 0 evaluates to True if the list has an even number of elements.
    if len(numbers) % 2 == 0:
        # If the list has an even number of elements, find the two middle numbers.
        # 'len(numbers) // 2' computes the index of the second middle element in a zero-indexed list.
        # 'len(numbers) // 2 - 1' computes the index of the first middle element.
        median1 = numbers[len(numbers) // 2] # The higher index of the two middle values.
        median2 = numbers[len(numbers) // 2 - 1] # The lower index of the two middle values.

        # Calculate the median by taking the average of the two middle numbers.
        # This is done by adding the two middle values and dividing by 2.
        mymedian = (median1 + median2) / 2 # Average of the two middle values.
    else:
        # If the list has an odd number of elements, return the middle number.
        # 'len(numbers) // 2' computes the index of the middle element.
```

```
mymedian = numbers[len(numbers) // 2] # The middle value for lists with an odd number of elements.  
# Return the calculated median value.  
return mymedian
```

Artık böyle bir ekran görmelisiniz.

## stats.py ●

```
1  def mean(numbers):  
2      """  
3          This function returns the mean of the given list of numbers.  
4      """  
5      return sum(numbers)/len(numbers)  
6  
7  
8  def median(numbers):  
9      """  
10         This function returns median of the given list of numbers.  
11     """  
12     numbers.sort()  
13  
14     if len(numbers) % 2 == 0:  
15         median1 = numbers[len(numbers) // 2]  
16         median2 = numbers[len(numbers) // 2 - 1]  
17         mymedian = (median1 + median2) / 2  
18     else:  
19         mymedian = numbers[len(numbers) // 2]  
20     return mymedian
```

Dosyayı stats.py olarak kaydedin.

## Create init.py

- \_\_init\_\_.py dosyasını oluşturun

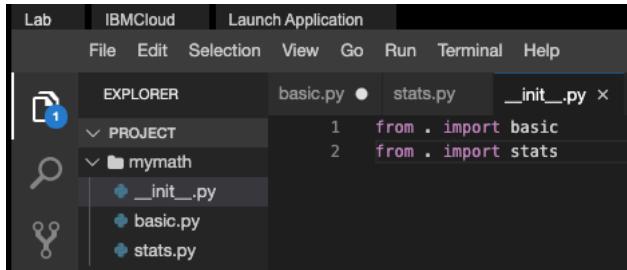
Aşağıdaki kodu \_\_init\_\_.py dosyasına kopyalayın ve yapıştırın.

```
from . import basic  
from . import stats
```

Dosyayı \_\_init\_\_.py olarak kaydedin.

Artık dizin yapınız şu şekilde görünmelidir:

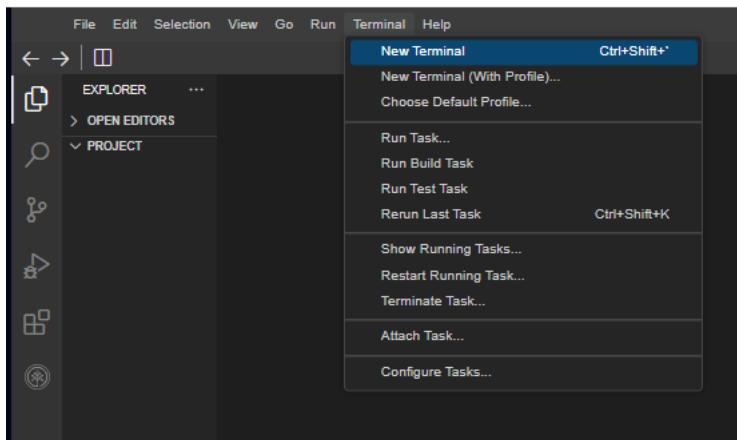
```
mymath
mymath/__init__.py
mymath/basic.py
mymath/statistics.py
```



Bir paket oluşturmayı tamamladınız.

## Paketi Doğrula

- Sağdaki pencerede, **Terminal** menüsüne tıklayın ve aşağıdaki resimde gösterildiği gibi **Yeni Terminal** seçeneğini seçin.



- Ekranın altında aşağıdaki resimdeki gibi bir terminal açıldığını göreceksiniz.

```
theia@theiadocker-rsannareddy: /home/project ×
theia@theiadocker-rsannareddy:/home/project$ █
```

- Terminalde **python3** yazarak python yorumlayıcısını başlatın.
- Python yorumlayıcıyı yüklen dikten sonra.
- Python isteminde **import mymath** yazın.
- Yukarıdaki komut hatasız çalışıysa, bu mymath paketinin başarıyla yükleniğinin bir göstergesidir.
- Python isteminde **mymath.basic.add(3,4)** yazın.
- Ekranda 7 çıktısını görmelisiniz.
- Python isteminde **mymath.stats.mean([3,4,5])** yazın.
- Ekranda 4.0 çıktısını görmelisiniz.
- Python yorumlayıcısından çıkmak için **exit()** yazın.

```
theia@theiadocker-rsannareddy: /home/project ×
theia@theiadocker-rsannareddy:/home/project$ python3
Python 3.6.9 (default, Oct  8 2020, 12:12:24)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import mymath
>>> mymath.basic.add(3,4)
7
>>> mymath.stats.mean([3,4,5])
4.0
>>> exit()
theia@theiadocker-rsannareddy:/home/project$ █
```

## Pratik Alıştırma

geometry adında yeni bir modül oluşturun ve mymath paketine ekleyin.

- geometry adında bir modül oluşturun
- Uzunluk ve genişliği girdi olarak alan ve bir dikdörtgenin alanını döndüren `area_of_rectangle` adında bir fonksiyon ekleyin.
- Yarıçapı girdi olarak alan ve bir dairenin alanını döndüren `area_of_circle` adında bir fonksiyon ekleyin.
- Bu modülü içerecek şekilde `__init__.py` dosyasını düzenleyin.
- Python terminalinden `area_of_circle` fonksiyonunu içe aktarın ve test edin.

## Yazarlar

Ramesh Sannareddy

## Diger Katkida Bulunanlar

Rav Ahuja

© IBM Corporation. Tüm hakları saklıdır.

Bu defter ve kaynak kodu [MIT Lisansı](#).