

Python 3 – Debugging: Best Practices

PyCharm Debugger: <https://www.jetbrains.com/help/pycharm/part-1-debugging-python-code.html>, Erişim: Haziran 2022.

PyCharm geniş debug araç yelpazesine sahiptir. Yukarıdaki bağlantıdan örneğe ulaşabilirsiniz.

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

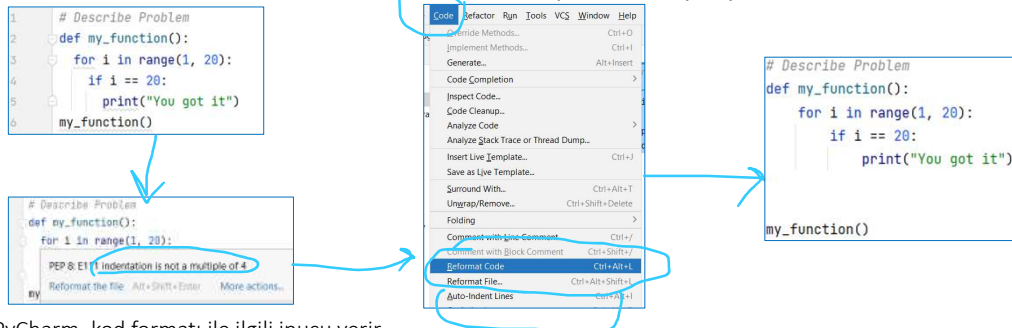
224

224

Python 3 – Debugging: Best Practices

PyCharm ile debug işlemi gerçekleştirilirken izlenmesini önerdiğim adımları sıralıyorum:

1. **Reformat Code:** Problemi tespit etmeye yardımcı.



PyCharm, kod formatı ile ilgili ipucu verir.

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

225

225

Python 3 – Debugging: Best Practices

PyCharm ile debug işlemi gerçekleştirilirken izlenmesini önerdiğim adımları sıralıyorum:

2. **Describe:** Beklenen ve gerçek çıktıyı karşılaştırın.


```
# Describe Problem
def my_function():
    for i in range(1, 20):
        if i == 20:
            print("You got it")

my_function()
```

Beklenen

Değişkenler:	Output
i=1	
i=2	
...	
i=20	You got it

Gerçek



```
.../pyclassroom2022/day1/debug-problems/problem1.py
Process finished with exit code 0
```

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

226

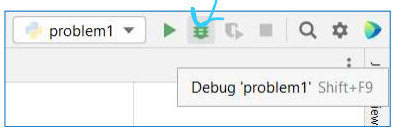
226

Python 3 – Debugging: Best Practices

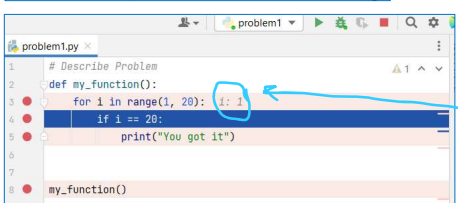
PyCharm ile debug işlemi gerçekleştirilirken izlenmesini önerdiğim adımları sıralıyorum:

3. **Break-points & run debug mode:** Kritik satırları işaretleyin ve programı debug modunda yürütün.

```
1 # Describe Problem
2 def my_function():
3     for i in range(1, 20):
4         if i == 20:
5             print("You got it")
6
7
8 my_function()
```



Debug 'problem1' Shift+F9



F9: breakpoint'ler arası geçiş.
Değişken değeri döngü boyunca takip edilir.

29.06.2022

227

227

Python 3 – Debugging: Best Practices

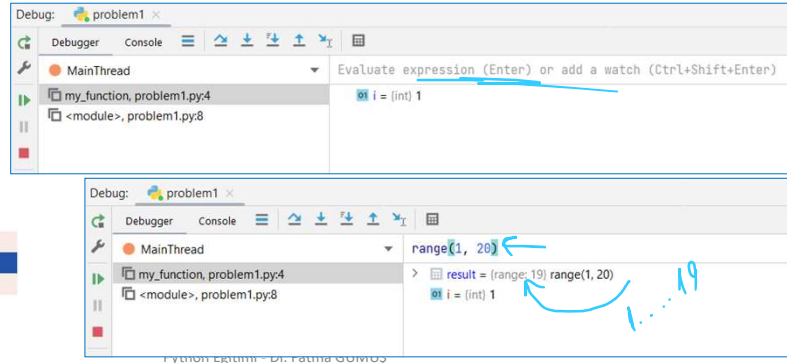
PyCharm ile debug işlemi gerçekleştirilirken izlenmesini önerdiğim adımları sıralıyorum:

- 3. Add Watch:** Gerek duyduğunuzda değerini izlemek istediğiniz ifadeleri girin:

i == 20 her zaman False

```
for i in range(1, 20): i: 1
    if i == 20:
        print("You got it")
```

29.06.2022



228

228

Python 3 – Debugging: Best Practices

PyCharm ile debug işlemi gerçekleştirilirken izlenmesini önerdiğim adımları sıralıyorum:

- 4. Print:** Hatadan şüphelendiğiniz değerleri ekrana yazdırın.

```
def my_function():
    for i in range(1, 20):
        print("i = ", i)
        if i == 20:
            print("You got it")
my_function()
```

i == 20 her zaman False

```
i = 1
i = 2
i = 3
i = 4
i = 5
i = 6
i = 7
i = 8
i = 9
i = 10
i = 11
i = 12
i = 13
i = 14
i = 15
i = 16
i = 17
i = 18
i = 19
```

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

229

229

Python 3 – Debugging: Best Practices

PyCharm ile debug işlemi gerçekleştirilirken izlenmesini önerdiğim adımları sıralıyorum:

5. **Fix the errors:** Hata oluştuysa hata mesajını çevrimiçi aratın.

Özellikle random değerler söz konusu olduğunda;

6. **Reproduce:** Daha basit bir örnekle program mantığını kontrol edin. Hatayı yeniden üretebildiniz mi?

7. **Play computer:** Bazen doğru, bazen yanlış mı çalışıyor? Yanlış çalışmaya neden olan bölümleri kalem-kağıt ile yazarak çalışıp izole edin, hataya neden olan durum hakkında işlem yapın.

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

230

230

Python 3 – Debugging: Best Practices

Alıştırma Ödevi

1. **problem2.py, ..., problem6.py** dosyalarındaki debug işlemlerini gerçekleştirin.
2. Aşağıdaki programların debug işlemlerini kâğıt kalem kullanarak gerçekleştirin.

```
# even or odd
number = int(input("Which number do you want to check?"))

if number % 2 == 0:
    print("This is an even number.")
else:
    print("This is an odd number.")
```

```
# leap year
year = input("Which year do you want to check?")

if year % 4 == 0:
    if year % 100 == 0:
        if year % 400 == 0:
            print("Leap year.")
        else:
            print("Not leap year.")
    else:
        print("Leap year.")
else:
    print("Not leap year.")
```

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

231

231

Python 3 – Debugging: Best Practices

Alıştırma Ödevi

3. Gereksinimleri verilen aşağıdaki programın birden fazla bug'ı vardır. Debug işlemlerini kâğıt kalem kullanarak gerçekleştirin.

Your program should print each number from 1 to 100 in turn.

When the number is divisible by 3 then instead of printing the number it should print "Fizz".

'When the number is divisible by 5, then instead of printing the number it should print "Buzz".'

'And if the number is divisible by both 3 and 5 e.g. 15 then instead of the number it should print "FizzBuzz"'

```
# fizz/buzz
for number in range(1, 101):
    if number % 3 == 0 or number % 5 == 0:
        print("FizzBuzz")
    if number % 3 == 0:
        print("Fizz")
    if number % 5 == 0:
        print("Buzz")
    else:
        print([number])
```

29.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

232

232