

در درس های قبلی در مورد ارسال مقدار به تابع، ارسال تابع به تابع و همچنین نحوه ایجاد توابع داخلی توضیح دادیم. حال می خواهیم این سه مبحث را در این درس و در قالب مبحث جدیدی به نام Decorator به کار ببریم. فرض کنید می خواهیم یک تابع را ایجاد کنیم که هر عددی که به آن دادیم را به توان 2 برساند و نتیجه را برگشت دهد. روش تعریف همچین تابعی به صورت زیر است:

```
def powEvenValue(number):  
    return number * number;
```

اما اگر بخواهیم فقط اعداد زوج را به توان دو برساند به دو روش می توانیم این کار را انجام دهیم. یا باید بدنه تابع را دستکاری کنیم :

```
def powEvenValue(number):  
    if ((number % 2) == 0):  
        return number * number  
    else:  
        number;
```

و اگر نخواهید بدنه تابع را دستکاری کنید می توانید از روش دوم استفاده کنید و آن استفاده از یک تابع داخلی است. بدین صورت که شما هر تغییری را که می خواهید در تابع اصلی اعمال کنید به یک تابع داخلی می دهید. کد بالا را به صورت زیر اصلاح می کنیم :

```
1 def getFunction(function):  
2  
3     def checkValue(num):  
4         if ((num % 2) == 0):  
5             return function(num)  
6         else:  
7             return num  
8  
9     return checkValue;  
10  
11 def powEvenValue(number):  
12     return number * number;  
13  
14  
15 result = getFunction(powEvenValue);  
16  
17 print(result(10));
```

100

در کد بالا با تابع اصلی کاری نداریم (خطوط 11-12). یک تابع تعریف می کنیم، که یک تابع دریافت می کند (خطوط 9-1). در داخل این تابع یک تابع دیگر تعریف می کنیم (خطوط 3-7)، که همان تابع داخلی بوده و همان تغییراتی را که قرار است در تابع اصلی بدهیم، از این تابع می خواهیم. این تابع یک پارامتر قبول می کند که همان عددی است که قرار است به توان برسد. در داخل بدنه این تابع در خط 4 چک می کنیم که اگر باقیمانده تقسیم عدد گرفته شده بر 2 برابر با 0 بود، عدد را به تابع اصلی بدهد تا آن را به توان 2 برساند در غیر اینصورت خود عدد را نشان دهد (خط 7). در خط 9 هم همین تابع داخلی را به وسیله تابع خارجی برگشت می دهیم.

در خط 15، تابع خارجی را صدا می زنیم. این تابع، یک تابع را به عنوان پارامتر دریافت می کند، و ما هم تابع اصلی (powEvenValue()) را به آن ارسال می کنیم و نتیجه را در یک متغیر با نام result می ریزیم. از آنجاییکه خروجی تابع خارجی یعنی تابع getFunction()، تابع داخلی checkValue() است، در نتیجه، متغیر result همان تابع checkValue() است و کافیت در خط 17 یک پرانتزها را در جلوی این متغیر قرار داده و یک عدد به آن بدهیم تا چک کند که آیا زوج است یا فرد؟

اگر بخواهید همان تابع اصلی خود را صدا زده و خروجی مورد نظر را دریافت کنید، کافیت که از مفهوم Decorator استفاده کنید. خطوط 13-17 کد بالا را حذف کرده و خط 11 کد زیر را قبل از تابع اصلی بنویسید. خط 11 همان کار خطوط 15 و 17 کد بالا را انجام می دهد و مفهوم Decorator هم همین است. در این خط علامت @ و سپس نام تابع خارجی را نوشته ایم:

```
1 def getFunction(function):  
2  
3     def checkValue(num):  
4         if ((num % 2) == 0):  
5             return function(num)  
6         else:  
7             return num  
8  
9     return checkValue;  
10  
11 @getFunction  
12 def powEvenValue(number):  
13     return number * number;  
14  
15 print(powEvenValue(10))
```

100

همانطور که در کد بالا مشاهده می کنید می توان تابع اصلی را صدا کرد و یک عدد به آن داد، تا مجذورش را محاسبه کند.