

מקשטים או Decorators

הרעיון של מקשטים בפייתון הוא האפשרות לשנות התנהגות של פונקציה מבלי לשנות פיזית את הקוד שלה.

זה מאפשר גמישות של קוד מקסימלית וזה מתאפשר בגלל שבפייתון אפשר להעביר פונקציה כפרמטר לפונקציה אחרת וגם להחזיר ערך שהוא פונקציה מתוך פונקציה אחרת.

נראה דוגמה פשוטה:

נניח שכתבנו פונקציה שמחלקת 2 מספרים ומחזירה תוצאה. לא בדקנו אם המחלק שווה לאפס ולכן יכולה להיווצר שגיאה:

```
def div(a,b):  
    return a/b
```

כעת נכתוב פונקציה ש 'תקשט' לנו את הפונקציה div בכך שהיא תבדוק אם המחלק שווה לאפס ותתריע על כך. נקרא לה: check

```
def check(function1):          # receives a function as a parameter  
    def inside_func(a,b):     # This can be called a wrapper function  
        if b == 0:  
            print ("cannot divide by zero")  
            return  
        function1(a,b)        # Calling our original function if all is fine  
    return inside_func         # Returns the wrapper to use the mechanism
```

הפונקציה check (המקשטת) למעשה מקבלת כפרמטר את הפונקציה הדורשת קישוט. הפונקציה הפנימית inside_func (העוטפת) היא זו שמוסיפה את הפונקציונליות ומפעילה את הפונקציה המקורית (שצריכה קישוט)

לפייתון ישנו תחביר מיוחד להוסיף את ה 'קישוט' לפונקציה על ידי הסימן @

כך:

```
@check  
def div(a,b):  
    return a/b
```

כעת אם נקרא ל div עם מחלק שהוא אפס, נקבל הודעת שגיאה.

הקוד כולו:

```
def check(function1):
    def inside_func(a,b):
        if b == 0:
            print ('cannot divide by zero')
            return
        function1(a,b)
    return inside_func
```

```
@check
def div(a,b):
    return a/b
```

```
def main():

    x = 10
    y = 0
    result = (div(x,y))
    if not result:
        print ('something went wrong')
```

function1 היא הפונקציה שצריכה קישוט. היא מועברת לפונקציה המקשטת שניקראת check כפרמטר. בתוך check ישנה פונקציה פנימית שניקראת inside_func (שמוכרת רק שם) שעושה את הבדיקה שרוצים (ההרחבה של הפונקציה שצריכה קישוט) ולכן inside_func גם מקבלת אותם פרמטרים שמקבלת function1 על מנת לקרוא ל function1 עם פרמטרים אלה. לסיום מחזירה check את inside_func כתוצאה שלה.

הסימן @ אומר להוסיף את השם של פונקצית הקישוט (שצמוד לסימן @) לפונקציה שמופיעה מייד בשורה לאחר מכן div.

על ידי השמת השורה @check בהערה, מנטרלים את המנגנון ואת ההתערבות שלנו בפעולת div.