MonkeyPatch: Debugging + Production Hotfix

Safwan Rahman
Github: @safwanrahman

DO NOT TRY THIS AT ~~HOME~~ PRODUCTION!

CAUTION!

Have you used it in your code? (Without tests)

Have You Pushed MonkeyPatched Code into Production?

Example!

```python
class Dog:  2 usages
    def bark(self):  3 usages
        return "Woof!"

def new_bark(self):  1 usage
    return "Meow!"

animal = Dog()
print(animal.bark())
print("Monkey patching...")
Dog.bark = new_bark
print(animal.bark())
```

OUTPUT

```
> python ex1.py
Woof!
Monkey patching...
Meow!
```

```python
from types import MethodType
class Dog:  1 usage
    def __init__(self, name):
        self.name = name
    def get_name(self):  3 usages
        return self.name

def patched_get_name(self):  1 usage
    return f"My name is {self.name}"


animal = Dog(name="Tike")
print(animal.get_name())
print("Monkey patching...")
animal.get_name = MethodType(patched_get_name, animal)
print(animal.get_name())
```

OUTPUT

```
safwan@safwans-MacBook-Pro ~/t/f/example> python3 ex2.py
Tike
Monkey patching...
My name is Tike
```

Debugging!

```python
# calc_module.py

def calculate_sum(a, b):
    return a - b
```

```python
# debug_1.py
from calc_module import (calculate_sum
                         as wrong_calculate_sum)
# Lets Monkey patch to debug the calculate_sum function
def patched_calculate_sum(a, b):  1 usage
    breakpoint()
    return wrong_calculate_sum(a, b)

import calc_module
calc_module.calculate_sum = patched_calculate_sum

from calc_module import calculate_sum
print(calculate_sum( a: 100,  b: 30))
```

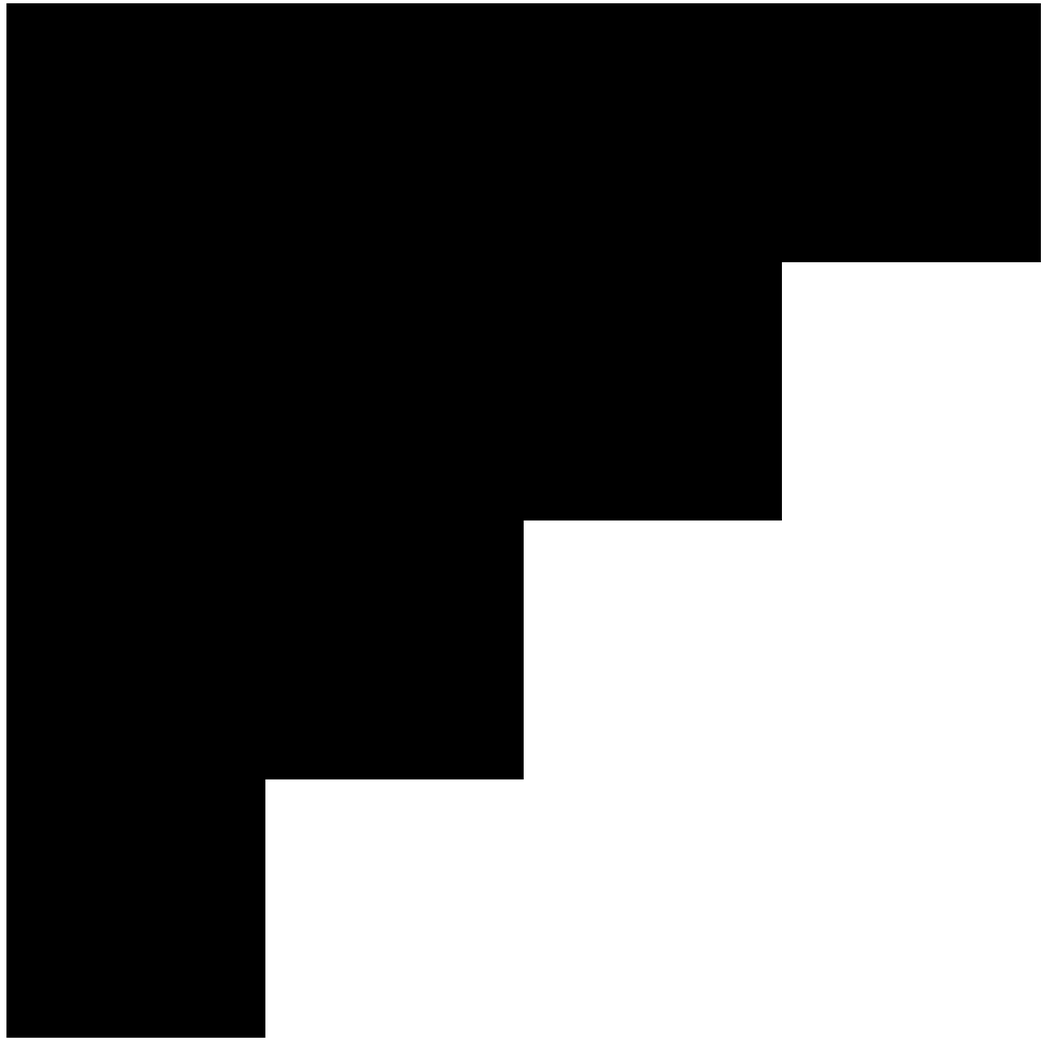
OUTPUT

```
safwan@safwans-MacBook-Pro ~/t/fosdem> python debugging/debug_1.py
> /Users/safwan/test/fosdem/debugging/debug_1.py(6)patched_calculate_sum()
-> return calculate_sum(a, b)
(Pdb)
```

```python
# debug_2.py
from calc_module import calculate_sum

print(calculate_sum( a: 100,  b: 30))
# Output: 70

# Lets Monkey patch the calculate_sum function
def patched_calculate_sum(a, b):  1 usage
    return a + b

import calc_module
calc_module.calculate_sum = patched_calculate_sum

from calc_module import calculate_sum
print(calculate_sum( a: 100,  b: 30))
# Output: 130
```

```python
# debug_3.py
import inspect
from calc_module import add_two

print(add_two(44))
# Output: 42

def patched_calculate_sum(a, b):  1 usage
    calling_function_name = inspect.stack()[1][3]
    calling_line = inspect.stack()[1][2]
    print(f"Calling function name: {calling_function_name}")
    print(f"Calling line: {calling_line}")
    # Fixed the calculate_sum function
    return a + b

import calc_module
calc_module.calculate_sum = patched_calculate_sum

from calc_module import add_two
def check_add_two():  1 usage
    print(add_two(44))
    # Output: 46
```

Hotfix!



See Example Codes

Safwan Rahman

Thank you!

safwan.rahman15@gmail.com

CAUTION!