

MRO&pizza

February 16, 2016

1 16.02.16 - MRO/Ausgabe/OOP allgemein - Laura

1) MRO.

Betrachten Sie die folgende Klassendefinitionen:

```
In [ ]: class A:
        pass
        class B:
            pass
        class C:
            pass
        class D:
            pass
        class E(A, B):
            pass
        class F(B, C, D):
            pass
        class G(E, F):
            pass
```

Wenden Sie auf diese Klassenhierarchie den C3-Algorithmus an, um die Method-Resolution-Order für die Klasse G zu bestimmen. Geben Sie alle Zwischenschritte an.

2) Instanzmethoden/Statische Methoden/Klassenmethoden.

Betrachten Sie folgendes Beispiel:

```
In [ ]: class Pizza:
        pizzacount = 0
        def __init__(self, cheese, meat):
            self.cheese = cheese
            self.meat = meat
            Pizza.pizzacount += 1

        @staticmethod
        def print_pizzacount():
            print("%s pizzas" % Pizza.pizzacount)

        class Calzone(Pizza):
            calzonecount = 0
            def __init__(self, cheese, meat, size):
                self.size = size
                Calzone.calzonecount += 1
```

```
        super().__init__(cheese, meat)

p1 = Pizza("mozzarella", "salami")
p2 = Pizza("gouda", "ham")
c1 = Calzone("mozzarella", "veg", "XL")
p1.print_pizzacount()
```

- a) Was wird ausgegeben?
- b) Was passiert wenn wir '@staticmethod' entfernen? Erklären Sie kurz den Unterschied zwischen Instanzenmethoden, statische Methoden und Klassenmethoden.