

# Кори Альтхофф

//Основано на реальном опыте!

```
import sys
```

```
class CannotResolve(Exception):  
    pass
```

```
class Resolver(object):  
    emess = "Can't found appropriate signature of func %s() for call with" + \  
        " params %r"  
    def __init__(self, name):  
        self.function_map = {}
```

# #САМ СЕБЕ ПРОГРАММИСТ

Как научиться программировать и устроиться в eBay?

```
        self.default = None  
        self.name = name  
    def __call__(self, *dt):  
  
        cls = tuple(map(type, dt))  
  
        try:  
            x = self.function_map[cls]  
        except KeyError:  
            if self.default is not None:  
                x = self.default  
            else:  
                raise CannotResolve(self.emess % (self.name, cls))  
        return x(*dt)  
    def overload(*dt):  
  
    def closure(func):  
        name = func.__name__  
        fr = sys._getframe(1).f_locals.get(name, Resolver(name))  
  
        fr.function_map[dt] = func  
  
        return fr  
    return closure  
    def overdef(func):  
        name = func.__name__
```

*{на примере Python}*