

```
from selectors import *
```

```
@selectors({'get': ['data','size'], 'set':['data', 'label']})
```

```
@private('data', 'size')
```

```
class E:
```

```
    def __init__(self, label, start):
```

```
        self.label = label
```

```
        self.data = start
```

```
        self.size = len(self.data)
```

```
@selectors({'get': ['data','size'], 'set':['data', 'label']})
```

```
@private('data', 'size', 'label')
```

```
class D:
```

```
    def __init__(self, label, start):
```

```
        self.label = label
```

```
        self.data = start
```

```
        self.size = len(self.data)
```

```
    def double(self):
```

```
        for i in range(self.size):
```

```
            self.data[i] = self.data[i] * 2
```

```
    def display(self):
```

```
        print('{0} => {1}'.format(self.label, self.data))
```

```
if __name__ == "__main__":
```

```
    try:
```

```
        Y = E('Wrong Label', ['a', 'b'])
```

```
    except Exception as e: print("{0}: {1}".format(type(e).__name__, e))
```

```
    X = D('X is', [1, 2, 3])
```

```
    X.display(); X.double(); X.display()
```

```
    try:
```

```
        print(X.size)
```

```
    except Exception as e: print("{0}: {1}".format(type(e).__name__, e))
```

```
    try:
```

```
        print(X.data)
```

```
    except Exception as e: print("{0}: {1}".format(type(e).__name__, e))
```

```
    print(X.getData())
```

```
    X.setData([25])
```

```
    print(X.getSize())
```

```
    try:
```

```
        X.setSize(25)
```

```
    except Exception as e: print("{0}: {1}".format(type(e).__name__, e))
```

```
    try:
```

```
        print(X.getLabel())
```

```
    except Exception as e: print("{0}: {1}".format(type(e).__name__, e))
```

```
    X.setLabel('It has worked!!!')
```

```
    X.display()
```