# 02 - Architektur

## **Datenkapselung**

Wir vermeiden direkte Zugriffe von ausserhalb auf die Attribute einer Klasse. Dadurch können wir das Prinzip der Datenkapselung sicherstellen.



Jedes Attribut einer Klasse wird mit @property gekennzeichnet.



Setter-Funktionen werden mittels @name1).setter gekennzeichnet.

#### Beispiel

```
class Celsius:
    def __init__(self, temperature=0):
        self._temperature = temperature

    @property
    def temperature(self):
        print('Getting value...')
        return self._temperature

    @temperature.setter
    def temperature(self, value):
        print('Setting value...')
        if value < -273.15:
            raise ValueError('Temperature below -273 is not possible')
        self._temperature = value</pre>
```

### Main

Zum Testen und zur Vermeidung von Fehlern kann ein Pythonskript unterscheiden, ob es direkt ausgeführt wurde oder als Import von einem anderen Skript aufgerufen wurde.



Jedes Pythonskript prüft, ob es direkt ausgeführt wird.

#### Beispiel

```
def hello():
    print('Hello World')

if __name__ == '__main__':
```

Last update: 2024/03/28 14:07

hello()

1)

«name» ist der Name der Property; z.B. @temperature.setter

From:

https://wiki.bzz.ch/ - BZZ - Modulwiki

Permanent link:

https://wiki.bzz.ch/howto/codingstandards/02\_architektur

Last update: 2024/03/28 14:07



https://wiki.bzz.ch/ Printed on 2025/11/13 09:45