

LU02.L02 - Klassendiagramm "Bottle" umsetzen

bottle.py

```
"""
Implements a bottle
"""

class Bottle:
    """
    class representing a bottle

    Attributes
    -----
    _quantity_available: float
    _color: str
    _capacity: float

    """

    def __init__(self, color, capacity):
        """
        constructor with color and capacity
        :param color:
        :param capacity:
        """
        self._quantity_available = 0.0 # 0.0 da float
        self._color = color
        self._capacity = capacity

    @property
    def color(self):
        """
        Gets the color of the bottle.
        :return: color(str)
        """
        return self._color

    @property
    def quantity_available(self):
        """
        Get the available quantity.
        :return: quantity_available(float)
        """
        return self._quantity_available
```

```
@property
def capacity(self):
    """
    Gets the capacity.
    :return: capacity(float)
    """
    return self._capacity

def open_bottle(self):
    """
    Empty method.
    """
    return

def close_the_bottle(self):
    """
    Empty method.
    """
    return

def fill_bottle(self):
    """
    Fill the bottle to capacity
    """
    self._quantity_available = self._capacity

def drink_water(self, amount):
    """
    Returns the specified amount of liquid if available.
    Otherwise, returns the available quantity.

    :param: amount (float): the requested amount
    :return: amount(float)
    """
    if amount > self._quantity_available:
        amount = self._quantity_available
        self._quantity_available = 0.0
    else:
        self._quantity_available -= amount
    return amount
```

m320-LU02



René Probst, bearbeitet durch Marcel Suter

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

https://wiki.bzz.ch/modul/m320_2024/learningunits/lu02/loesungen/bottle

Last update: **2024/08/20 09:49**

