

Lösungen für Kapitel 3

3.1 ~~Es sollte natürlich der Besucher und nicht der Beobachter verändert werden;~~

Hier die Lösung für die PKWs in Python

```
#####
# encoding: iso-8859-15
# Entwurfsmuster Besucher (Besucher) in Python
# Besucher.py
#####

#Abstrakte Klassen
class AbstrBesucher(object):
    def besucheFahrrad(self, element): pass
    def besuchePKW(self, element): pass
    def besucheLKW(self, element): pass

class AbstrElement(object):
    def akzeptiereBesucher(self, Besucher): pass

#Besucher
class DruckBesucher(AbstrBesucher):
    def besuchePKW(self, element):
        print "PKW: " + element.name
    def besucheLKW(self, element):
        print "LKW " + element.name
    def besucheFahrrad(self, element):
        print "Fahrrad: " + element.name

class SommerpreisBesucher(AbstrBesucher):
    def besuchePKW(self, element):
        print element.name + " Preis: ",
        if element.name == "BMW":
            print 23000-element.alter*800
        elif element.name == "GO":
            print 5000 -element.alter*100
        elif element.name == "MB":
            print 30000-element.alter*1000
        else:
            print 13000-element.alter*500
    def besucheLKW(self, element):
        print element.name + " Preis: ",
        print 26000-element.alter*500 + element.last
    def besucheFahrrad(self, element):
        print element.name + " Preis: ",
        print 1200-element.alter*200

class WinterpreisBesucher(AbstrBesucher):
    def besuchePKW(self, element):
        print element.name + " Preis: ",
        if element.name == "BMW":
            print 20000-element.alter*800
        elif element.name == "GO":
            print 4000 -element.alter*100
        elif element.name == "MB":
            print 27000-element.alter*1000
        else:
            print 10000-element.alter*500
    def besucheLKW(self, element):
        print element.name + " Preis: ",
        print 30000-element.alter*500 + element.last - 6000
    def besucheFahrrad(self, element):
        print element.name + " Preis: ",
        print 1000-element.alter*200
```

```

#Elemente
class PKW(AbstrElement):
    def __init__(self, name, alter):
        self.name = name
        self.alter = alter
    def akzeptiereBesucher(self, Besucher):
        Besucher.besuchePKW(self)

class Fahrrad(AbstrElement):
    def __init__(self, name, alter):
        self.name = name
        self.alter = alter
    def akzeptiereBesucher(self, Besucher):
        Besucher.besucheFahrrad(self)

class LKW(AbstrElement):
    def __init__(self, name, alter, las):
        self.name = name
        self.alter = alter
        self.last = las
    def akzeptiereBesucher(self, Besucher):
        Besucher.besucheLKW(self)

#Einsatz des Besucher
def strukturDurchlauf(struktur, Besucher):
    for element in struktur:
        element.akzeptiereBesucher(Besucher)

if __name__=="__main__":
    struktur=[ PKW("BMW", 2), LKW("GO", 3, 9000), PKW("MB", 4),
              LKW("Mercedes", 1, 10), Fahrrad("Diamant", 3),]

    print "Druck vorhandener Fahrzeuge"
    strukturDurchlauf(struktur, DruckBesucher())
    print
    print "Sommerpreistabelle"
    strukturDurchlauf(struktur, SommerpreisBesucher())
    print
    print "Winterpreistabelle"
    strukturDurchlauf(struktur, WinterpreisBesucher())

```