

工厂模式:

实现了创建者和调用者的分离, 使专门的工厂类将选择实现类、创建对象进行统一管理和控制

【实现】实现工厂类

```
class CarFactory:
    def createCar(self, brand):
        if brand == "奔驰":
            return Benz()
        elif brand == "宝马":
            return BMW()
        else:
            return BYD()
class Benz:
    pass
class BMW:
    pass
class BYD:
    pass
```

```
car = CarFactory()
benchi = car.createCar("奔驰")
baoma = car.createCar("宝马")
print(isinstance(benchi, Benz))
```

【效果】类似于

```
vbox = Box.createBox(verticalBox)
```