# 实验三实验报告

**简单工厂方法**

## 1.1 UML设计图



## 1.2 核心代码

|  |
| --- |
| **public** **class** MeatFactory {  **public** MeatFactory() {    }  **public** **static** Meat getMeat(String meatTypeString) {  Meat meat = **null**;  **switch** (meatTypeString) {  **case** "beef":  meat = **new** Beef();  **break**;  **case** "pork":  meat = **new** Pork();  **break**;  **case** "chicken":  meat = **new** Chicken();  **break**;  }  **return** meat;  } |

## 1.3 运行效果



### 工厂模式

#### 2.1 UML图



#### 2.2 代码

|  |
| --- |
| **package** cn.edu.scau.cmi.lianzongsheng.factory;  **import** cn.edu.scau.cmi.lianzongsheng.simpleFactoryMethod.Meat;  **public** **abstract** **class** MeatFactory {  **public** **abstract** Meat getMeat();  **public** **static** MeatFactory getFactory(String meatFactoryType) {  MeatFactory meat\_factory =**null**;  **switch** (meatFactoryType) {  **case** "BeefMeatFactory":  meat\_factory = **new** BeefMeatFactory();  **break**;  **case** "PorkMeatFactory":  meat\_factory = **new** PorkMeatFactory();  **break**;  **case** "ChickenMeatFactory":  meat\_factory = **new** ChickenMeatFactory();  **break**;  }  **return** meat\_factory;    }  } |

#### **2.3 效果**



**抽象工厂模式**

**3.1 UML图**



#### 3.2 代码

|  |
| --- |
| package cn.edu.scau.cmi.lianzongsheng.abstractFactory;  import cn.edu.scau.cmi.lianzongsheng.simpleFactoryMethod.Beef;  import cn.edu.scau.cmi.lianzongsheng.simpleFactoryMethod.Chicken;  import cn.edu.scau.cmi.lianzongsheng.simpleFactoryMethod.Meat;  import cn.edu.scau.cmi.lianzongsheng.simpleFactoryMethod.Pork;  public abstract class Brand {  public static Brand getBrand(String brandType) {  Brand brand = null;  switch (brandType) {  case "Wens":  brand = new Wens();  break;  case "Tianbang":  brand = new Tianbang();  break;  case "Aonong":  brand = new Aonong();  break;  }  return brand;  }  public abstract String getBrandName();  public abstract Pork getPork();  public abstract Beef getBeef();  public abstract Chicken getChicken();  } |

#### **3.3 效果**

