8. MyPizzaFactory 결과



9. NYyPizzaFactory, ChicagoyPizzaFactory 결과





**PizzaFactory.java**

package Mid\_Term;  
  
public abstract class PizzaFactory {  
  
 public abstract Pizza getClamPizza();  
 public abstract Pizza getCheesePizza();  
 public abstract Pizza getPepperoniPizza();  
 public abstract Pizza getVeggiePizza();  
  
}

**NYPizzaFactory.java**

package Mid\_Term;  
  
public class NYPizzaFactory extends PizzaFactory{  
 public Pizza getClamPizza(){return new NYStyleClamPizza();}  
 public Pizza getCheesePizza(){return new NYStyleCheesePizza();}  
 public Pizza getPepperoniPizza(){return new NYStylePepperoniPizza();}  
 public Pizza getVeggiePizza(){return new NYStyleVeggiePizza();}  
}

**ChicagoPizzaFactory.java**

package Mid\_Term;  
  
public class ChicagoPizzaFactory extends PizzaFactory{  
 public Pizza getClamPizza(){return new ChicagoStyleClamPizza();}  
 public Pizza getCheesePizza(){return new ChicagoStyleCheesePizza();}  
 public Pizza getPepperoniPizza(){return new ChicagoStylePepperoniPizza();}  
 public Pizza getVeggiePizza(){return new ChicagoStyleVeggiePizza();}  
  
}

**MyPizzaStore.java**

package Mid\_Term;  
public class MyPizzaStore {  
 PizzaFactory pizzaFactory;  
  
 public MyPizzaStore(PizzaFactory pizzaFactory){  
 this.pizzaFactory = new MyPizzaFactory(); //5번 내용에 대하여 남겨둠  
 this.pizzaFactory = pizzaFactory;  
 }  
 public Pizza createPizza(String type) {  
 if (type.equals("cheese")) {  
 return pizzaFactory.getCheesePizza();  
 } else if (type.equals("pepperoni")) {  
 return pizzaFactory.getPepperoniPizza();  
 } else if (type.equals("clam")) {  
 return pizzaFactory.getClamPizza();  
 } else if (type.equals("veggie")) {  
 return pizzaFactory.getVeggiePizza();  
 }  
 return null;  
 }  
  
 public Pizza orderPizza(String type) {  
 Pizza pizza = createPizza(type);  
 System.*out*.println("--- Making a " + pizza.getName() + " ---");  
 pizza.prepare();  
 pizza.bake();  
 pizza.cut();  
 pizza.box();  
 return pizza;  
 }  
}