

Factory Method Pattern

하위 클래스에서 인스턴스 작성

- 상위 클래스에서 인스턴스 작성법의 뼈대를 세우고, 구체적인 작성은 하위 클래스에서 실행.

[Architecture](#), [Modeling](#), [DesignPattern](#), [Creational](#)

인스턴스 생성을 위한 **Framework**(골격) 측

```
package framework;

/* Product */
public abstract class Product {
    public abstract void use();
}
```

```
package framework;

/* Creator */
public abstract class Factory {
    public final Product create(String owner) {
        Product p = createProduct(owner);
        registerProduct(p);
        return p;
    }

    protected abstract Product createProduct(String owner);
    protected abstract void registerProduct(Product product);
}
```

구체적인 구현하고 있는 측

```
package product;

import framework.Product;

/* ConcreteProduct */
public class IDCard extends Product {

    private String owner;
    public IDCard(String owner) {
        this.owner = owner;
    }
}
```

```
}  
@Override  
public void use() {  
    System.out.println(owner + "의 카드 사용");  
}  
public String getOwner() {  
    return this.owner;  
}  
}
```

```
package product;  
  
import java.util.ArrayList;  
import java.util.List;  
  
import framework.Factory;  
import framework.Product;  
  
/* ConcreteCreator */  
public class IDCardFactory extends Factory {  
  
    private List<String> owners = new ArrayList<>();  
    @Override  
    protected Product createProduct(String owner) {  
        return new IDCard(owner);  
    }  
  
    @Override  
    protected void registerProduct(Product product) {  
        if(!(product instanceof IDCard))  
            return;  
        owners.add(((IDCard) product).getOwner());  
    }  
}
```

실행

```
import framework.*;  
import product.*;  
  
public class Main {  
    public static void main(String[] args) {  
        Factory factory = new IDCardFactory();  
    }  
}
```

```
        Product card1 = factory.create("철수");
        Product card2 = factory.create("영희");
        card1.use();
        card2.use();
    }
}

/*
철수의 카드 사용
영희의 카드 사용
*/
```

From:

<http://ledyx.xyz/> - **Ledyx WIKI**

Permanent link:

http://ledyx.xyz/doku.php?id=design_pattern:factory_method_pattern&rev=1509105332

Last update: **2021/02/07 03:15**

