

SWIFT Factory

Bill Kim(김정훈) | ibillkim@gmail.com

목차

Factory

Implementation

References

Factory

Factory(팩토리) 디자인 패턴은 객체 생성과 관련된 디자인 패턴으로서 사용자에게 **객체 생성 로직을 노출하지 않은채 객체를 생성** 하기 위해서 제공되는 패턴입니다.

사용자는 단순히 필요한 객체의 생성 로직을 신경쓰지 않고 특정 객체에 대해서 가져올 수 있습니다.

Implementation

구체적인 구현에 대해서 소스 코드를 통하여 살펴봅니다.

```
class Factory {  
    func makePen() -> Pen {  
        return Pen()  
    }  
  
    func makePencil() -> Pencil {  
        return Pencil()  
    }  
}  
  
class Pen {  
    init() {  
        print("Pen init")  
    }  
  
    func make() {  
        print("Make a pen.")  
    }  
}  
  
class Pencil {  
    init() {  
        print("Pencil init")  
    }  
  
    func make() {  
        print("Make a pencil.")  
    }  
}
```

Implementation

```
let factory = Factory()
```

```
let pen = factory.makePen() // Pen init  
pen.make() // Make a pen.
```

```
let pencil = factory.makePencil() // Pencil init  
pencil.make() // Make a pencil.
```

References

[1] Design Pattern - Factory : <https://ehdrjsdlzzzz.github.io/2019/04/03/Design-Pattern-Factory/>

[2] Factory Pattern(swift) : <https://linsaeng.tistory.com/5>

[3] Design Pattern - Factory : <https://ehdrjsdlzzzz.github.io/2019/04/03/Design-Pattern-Factory/>

[4] Swift : Factory Pattern : <https://medium.com/@mayooresan/swift-factory-pattern-6f0d7556d862>

[5] Design Pattern) 팩토리 패턴 : <https://o-o-wl.tistory.com/18>

References

[6] Objective-c로 구현한 Factory Pattern (Protocol을 이용한 추상 팩토리) : <https://thdev.net/321>

[7] Design Patterns — Creational Patterns — Factory Pattern in Swift : <https://medium.com/swift-programming/design-patterns-creational-patterns-factory-pattern-in-swift-d049af54235b>

[8] 18일차 [디자인 패턴] 팩토리 패턴 : <https://woovictory.github.io/2019/02/07/Design-Pattern-Factory-Pattern/>

[9] 디자인 패턴 : 추상 팩토리 vs 팩토리 메소드 : <https://www.it-swarm.dev/ko/design-patterns/디자인-패턴-추상-팩토리-vs-팩토리-메소드/970371311/>

[10] 디자인패턴 - 팩토리 패턴 (factory pattern) : <https://jusungpark.tistory.com/14>

Thank you!