

인스턴스 생성 및 소멸

이니셜라이저

새로운 인스턴스를 사용하기 위해 저장 프로퍼티의 초기값을 초기화해준다.

키워드 `init`

프로퍼티 기본값

- 인스턴스의 옵셔널 저장 프로퍼티를 제외한 모든 저장프로퍼티에 기본값을 할당해줘야 한다.
- 저장 프로퍼티에 기본값을 설정해두면 인스턴스 생성시 초기값을 가지고 있다.

```
1 class BasicInformation {
2     let height : Float = 0.0
3     var weight : Float = 0.0
4 }
5
6 var kimInfo : BasicInformation = BasicInformation()
7
8 print(kimInfo.weight)// 0.0
9 print(kimInfo.weight)// 0.0
10
```

이니셜라이저

이니셜라이저를 정의할 때 함수를 정의할 때와 마찬가지로 매개변수를 가질 수 있다.

전달인자 레이블이 필요 없을 시 와일드카드 식별자(`_`) 사용가능

```
class BasicInformation {
    let height : Float
    var weight : Float

    init(_ height: Float, KG weight: Float) {
        self.height = height
        self.weight = weight
    }
}

var kimInfo : BasicInformation = BasicInformation(118.0, KG: 99.0)

print(kimInfo.height)
print(kimInfo.weight)
```

클래스 인스턴스의 상수 프로퍼티는 정의된 클래스에서만 초기화 가능

해당 클래스를 상속받은 자식 클래스의 이니셜라이저에서 상수 프로퍼티 값을 초기화 할 수 없음

옵셔널 프로퍼티 타입

인스턴스가 사용되는 동안에 값을 꼭 갖지 않아도 되는 저장 프로퍼티는 옵셔널 타입으로 지정
값이 없을때 자동으로 nil이 할당

```
1 class Person {
2     var name : String
3     var age : Int?
4
5     init(name: String) {
6         self.name = name
7     }
8 }
9
10 let kim : Person = Person(name: "kim")
11 |
```

실패 가능한 이니셜라이저

반환 타입은 옵셔널 타입

이니셜라이저 매개변수로 전달되는 초기값이 잘못된 경우 인스턴스 생성에 실패할 수 있습니다.

```
1 class Person {
2     var name : String
3     var age : Int?
4
5     init?(name: String) {
6         if name.isEmpty {
7             return nil
8         }
9         self.name = name
10    }
11
12    init?(name: String, age: Int) {
13        if name.isEmpty || age < 0 {
14            return nil
15        }
16        self.name = name
17        self.age = age
18    }
19 }
20
21 let kim : Person? = Person(name: "kim") // name : kim age : nil
22 let who : Person? = Person(name: "") // name : nil age : nil
23 let yam : Person? = Person(name: "yam", age: -1) // name : yam age : nil
24
```