



제10장

생성자와 변수의 초기화

1. 생성자(constructor)란?

■ 생성자의 개념

- 몇 가지 조건만 제외하면, 근본적으로 메서드와 같다.
 - 인스턴스가 생성될 때마다, 호출되는 ‘인스턴스 초기화 메서드’ 라고 생각하자.
 - 인스턴스 변수의 초기화 또는 인스턴스 생성 시 미리 수행될 코드를 작성함.
(초기화란 생성자 호출시 매개변수 값으로 인스턴스 변수의 값으로 대입함.)
- “ 모든 클래스에는 반드시 하나 이상의 생성자가 있어야 한다.”**
(기본적으로 생성자가 존재하지 않을 때, 컴파일러가 컴파일시 생성해 줌)

```
Student s1 = new Student();
```

new는 연산자임(new가 인스턴스를 생성하는 것임.
생성자가 인스턴스를 생성하는 것이 아님.)

1. new 연산자에 의해서 힙에 Student클래스의 인스턴스(붕어빵)이 만들어진단다.
2. 생성자 Student()가 호출되어 수행코드를 실행한다.
3. 연산자 new의 결과로, Student인스턴스의 주소가 반환되어 참조변수 s1에 저장된다.

2. 생성자의 조건

■ 생성자의 조건

- 생성자의 이름은 반드시 클래스의 이름과 동일해야 한다.
- 생성자는 리턴 값이 없다. 그렇다고 void를 입력하지 않는다.(중요)

```
public class Student {  
  
    String name;  
    int age;  
  
    public Student() {  
        return; (생성자에도 return;을 쓸 수가 있  
                다.)  
    }  
  
    public Student(String name,int age) {  
        this.name = name;  
        this.age = age;  
    }  
}
```

1. 하나의 클래스에 여러 개의 생성자 사용할 수 있음.(오버로딩 개념)
2. 하여, 오버로딩의 규칙을 따라야 함.

3. 기본 생성자(default constructor)-1

■ 기본 생성자란?

- 지금까지 생성자를 따로 작성하지 않았음에도 불구하고, 인스턴스가 생성되었다.
(이유는 클래스의 생성자가 하나라도 없으면, 컴파일러가 기본 생성자를 추가해 주어 모든 클래스는 생성자가 하나 이상의 생성자가 반드시 있어야 한다는 법칙을 지켜준다.)
- 단, 클래스에 생성자가 하나라도 존재한다면, 컴파일러는 기본 생성자를 추가해주지 않는다.
- 기본 생성자는 매개변수가 없는 것을 말한다.

```
public Student() {  
}
```

(중요) “모든 클래스에는 반드시

하나 이상의 생성자가 반드시 있어야 한다.”

3. 기본 생성자(default constructor)-2

■ 예제

```
class A {  
    int value;  
}  
class B {  
    int value;  
  
    public B(int x) {  
        this.value = x;  
    }  
}  
public class ConstructorExample {  
  
    public static void main(String[] args) {  
  
        A a = new A();  
        B b = new B();  
  
    }  
}
```

예외가 발생치 않게 하기 위해서는 2가지
방법이 있음.

1. public B() 기본 생성자 추가.
2. 생성자 호출 시 매개변수로 int값 대입.

B클래스는 매개변수가 있는 생성자가 있으므로 컴파일러가 기본 생성자를 추가 해주지 않는다. 하여, 예외가 발생함

4. 매개변수가 있는 생성자

- 생성자에 매개변수가 하나 이상 존재하는 경우

```
public class Student {  
    String name;  
    int age;
```

```
    public Student() {  
    }  
}
```

```
    public Student(String name, int age) {  
        this.name = name;  
        this.age = age;  
    }  
}
```

호출

```
Student s1 = new Student();  
s1.name = "이영준";  
s1.age = 55;
```

코드 절약

```
Student s1 = new Student("이영준", 55);
```

호출

감사합니다.

