

Inheritance(extends) I - Constructor

상속관계에 있는 인스턴스 생성의 예 - Keyword super

// 상위, 기초, 부모 Class

```
class Man {  
    public static void main(String[] args){  
        private String name;  
  
        public Man(String name) { this.name=name };  
  
        public void tellYourName(){ System.out.println("My Name is "+name); }  
    }  
}
```

// 하위, 유도, 자식 Class

```
class BusinessMan extends Man {  
    private String company;  
    private String position;
```

```
    Public BusinessMan(String name, String company, String position) {
```

```
        super(name);
```

```
        super : 상위, 부모 Class
```

```
        super() : 상위 Class의 생성자를 호출하겠다!
```

```
        this.company = company;  
        this.position = position;
```

★ 규칙 : 실행순서를 보면 상위 Class가 먼저 초기화 되고, 하위 Class를 초기화해야 한다.

```
    }
```

```
    public void tellYourInfo() {  
        System.out.println("My company is "+company);  
        System.out.println("My position is "+position);  
        tellYourName();  
    }
```

```
}
```

```
class ExtendsClass {  
    public static void main(String[] args) {  
        BusinessMan man1 = new BusinessMan("Mr.Kim", "Google", "Staff Eng.");  
        BusinessMan man2 = new BusinessMan("Mr.Lee", "Google", "Assist Eng.");  
  
        System.out.println("First Main Info..");  
        man1.tellYourInfo();  
        System.out.println("Second Main Info..");  
        man2.tellYourInfo();  
    }  
}
```

- 하위 Class의 생성자는 상속member Variable을 초기화 할 의무가 있다.
- 하위 Class에서 상위 Class의 Member를 초기화 할 때에는 상위 Class의 생성자를 super()를 통해 이용하는 것이 타당!

반드시 호출되어야 하는 상위 Class의 생성자

```
class AAA {
```

```
int num1;

// Default Constructor
AAA() {

}

}
```

```
class BBB extends AAA {
    int num2;
```

상속받은 하위 Class의 Default 생성자는 상위 Class의 생성자를 호출해야 할 의무
(상위 Class의 Member를 초기화 해야 할 의무)가 있으므로, super()로 상위 Class의 생성자가 자동삽입됨.

```
// Default Constructor
BBB() {
    super();
}
```

```
BBB(int n) {
    Super(); // 자동삽입
    this.num2 = n;
}

}
```