

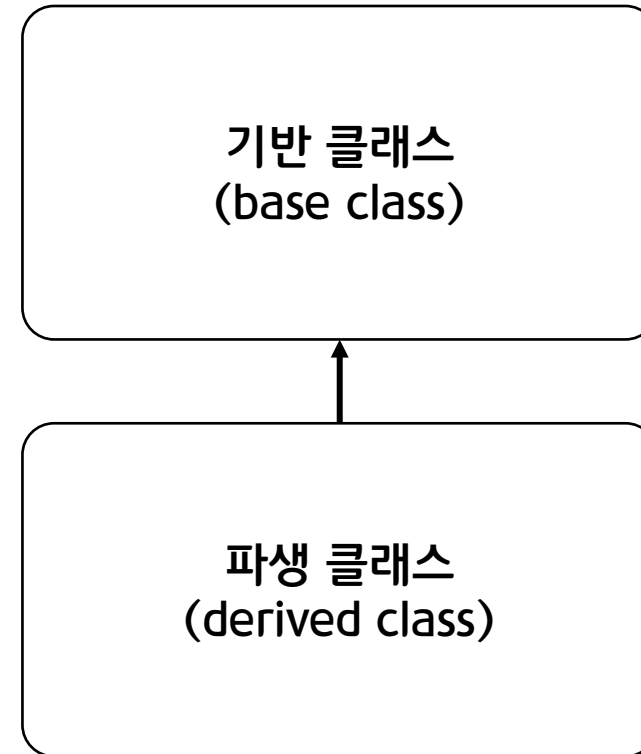
10강

클래스 상속

10-1 클래스 상속 개요

- 클래스 상속

- ✓ 클래스 상속은 기존에 있는 기능을 재사용할 때 사용한다.
- ✓ 또한 기존 기능을 이어 받고 더불어 클래스에 다른 기능을 추가할 때 사용하는 기능이다.
- ✓ 기능을 물려주는 클래스를 부모 클래스 (parent class), 상속을 받아 새롭게 만드는 클래스를 자식 클래스(child class)라고 한다.



10-1 클래스 상속

● 클래스 상속

- ✓ 아래 코드는 Person 클래스를 상속받아 Student 클래스를 새롭게 구현하였다.
- ✓ Student 클래스는 Person 클래스를 상속 받았기 때문에 Person 클래스의 기능을 사용할 수 있다.

```
class Person:
    def insa(self):
        print('안녕하세요')

class Student(Person):
    def study(self):
        print('공부하는 중...')

obj1 = Student()
obj1.study()
obj1.insa()
```

[결과]

공부하는 중...

안녕하세요

10-1 클래스 상속

● 클래스 상속

- ✓ 왼쪽 코드를 진행하면 에러가 발생한다.
- ✓ 파이썬은 상속 받은 클래스의 인스턴스 속성을 사용하기 위해서는 부모 클래스의 `__init__()`를 호출해야 한다.
- ✓ 부모 클래스의 `__init__()`를 호출하기 위해서는 `super()`를 사용한다.

```
class AAA:
    def __init__(self):
        print('AAA init method')
        self.message = 'hi everyone'
```

```
class BBB(AAA):
    def __init__(self):
        print('BBB init method')
        print(self.message)
```

```
obj1 = BBB()
```

[결과]

AttributeError: 'BBB' object has no attribute 'message'

```
class AAA:
    def __init__(self):
        print('AAA init method')
        self.message = 'hi everyone'
```

```
class BBB(AAA):
    def __init__(self):
        super().__init__()
        print('BBB init method')
        print(self.message)
```

```
obj1 = BBB()
```

[결과]

AAA init method
BBB init method
hi everyone

10-1 클래스 상속

● 오버라이딩

- ✓ 오버라이딩(overriding)은 우선하다 라는 뜻을 가지고 있는데 상속에서 이 기능은 부모 클래스의 특정 메서드를 무시하고 자식 클래스에서 새롭게 특정 메서드를 정의해서 사용하는 방법이다.

```
class Person:
    def insa(self):
        print('안녕하세요')

class Student(Person):
    def insa(self):
        super().insa()
        print('저는 파이썬 초등학교 학생입니다.')

obj1 = Student()
obj1.insa()
```

[결과]

안녕하세요

저는 파이썬 초등학교 학생입니다.

10-2 추상 클래스

● 추상 클래스

- ✓ 추상클래스는 미완성된 메서드 즉 추상 메소드를 한 개 이상 가지고 있는 클래스를 말한다.
- ✓ 추상 클래스를 사용하기 위해서는 상속을 이용한다. 즉 자식클래스에서 해당 부모 클래스의 추상 메소드를 구현하여 사용하면 된다.
- ✓ 추상클래스를 만들기 위해서는 abc모듈을 import 해야 한다.

```
from abc import *
```

```
class StudentBase(metaclass=ABCMeta):
```

```
    @abstractmethod
```

```
    def study(self):
```

```
        pass
```

```
    @abstractmethod
```

```
    def gotoSchool(self):
```

```
        pass
```

10-2 추상 클래스

● 추상 클래스

- ✓ 추상 클래스는 상속 받는 클래스의 메서드를 구현을 강제하기 사용한다.
- ✓ 추상 클래스는 특정 메소드가 상속 받아서 사용할 때 자식 클래스의 성격에 따라 메서드 구현이 다르게 구현해야 할 때 사용한다.

```
from abc import *  
  
class StudentBase(metaclass=ABCMeta):  
    @abstractmethod  
    def study(self):  
        pass  
  
    @abstractmethod  
    def gotoSchool(self):  
        pass  
  
class Student(StudentBase):  
    def study(self):  
        print('공부 중...')  
  
    def gotoSchool(self):  
        print('학교 가는 중...')  
  
obj1 = Student()  
obj1.study()  
obj1.gotoSchool()
```

[결과]

공부 중...

학교 가는 중...