

5주차(3/3)

# 객체지향 프로그래밍

파이썬으로 배우는 기계학습

한동대학교  
김영섭 교수

# 객체 지향 프로그래밍

---

- 학습 목표
  - 객체의 특성을 이해한다.
  - 객체와 클래스가 무엇인지 이해한다.
  - 생성자가 무엇인지 이해한다.
  - 매개변수 **self**의 역할이 무엇인지 이해한다.
- 학습 내용
  - 객체
  - 클래스와 객체
  - 생성자
  - 매개변수 **self**

# 1. 용어설명: 객체지향프로그래밍

---

- Object Oriented Programming  
→ OOP

## 2. OOP 핵심: 실제 세계 표현

---

- 세상에 있는 모든 것은 객체 즉 **object**임
- **OOP**를 통해 세상의 객체 모델링 가능

### 3. 실제 세계 표현: 속성(데이터)과 기능(함수)

---



- 속성(데이터)
  - 차 길이 : 4m
  - 차 무게 : 1380kg
- 기능(함수)
  - 전진
  - 후진

### 3. 실제 세계 표현: 객체 간 상호작용

---



### 3. 실제 세계 표현: 객체의 독립성

---

**Speed UP!!**



**I don't care~**



## 4. 객체: 객체의 특성 정리

---

- 1) 데이터와 함수로 구성됨
- 2) 객체 간 상호작용 함
- 3) 객체 간 독립적으로 존재



## 4. 객체: 예제 코딩(기능)



- 속성(데이터)
  - 차 길이 : 4 m
  - 차 무게 : 1380 kg
- 기능(함수)
  - 전진
  - 후진

```
mycar = Car(4, 1380)
```

```
print(mycar.length)  
print(mycar.weight)
```

```
4  
1380
```

```
mycar.forward()  
mycar.backward()
```

```
Move Forward!  
Move Backward!
```

## 4. 객체: 예제 코딩(상호작용)



```
mycar = Car(4, 1380)  
owner = Person()
```

```
owner.start(mycar)
```

Whoom!!

```
print(owner.pos)  
mycar.move(owner)  
print(owner.pos)
```

```
0  
10
```

## 4. 객체: 예제 코딩(독립성)

**Speed UP!!**



**I don't care~**



```
mycar = Car(4, 1380)  
yourcar = Car(4, 1500)
```

```
mycar.speedUP()  
print(mycar.speed)  
print(yourcar.speed)
```

10

0

## 5. 클래스: 정의

- 객체를 만들어 내는 설계도



### 자동차

#### [데이터]

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

#### [함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 5. 클래스: 이름 짓기

```
1 class Car:
2     def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```

### 자동차

#### [데이터]

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

#### [함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 5. 클래스: 이름 짓기

```
1 class Car:
2     def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```

자동차

[데이터]

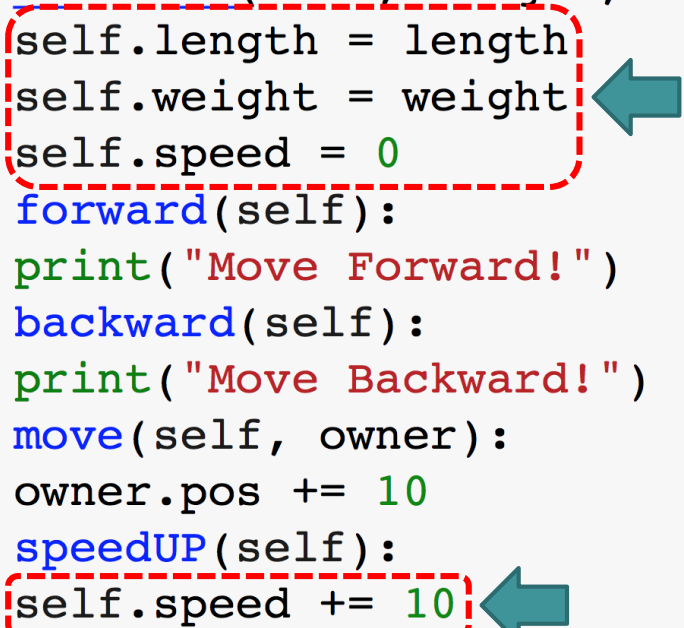
- 길이 (length)
- 무게 (weight)
- 속도 (speed)

[함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 5. 클래스: self.로 시작하는 변수

```
1 class Car:
2     def init (self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```



### 자동차

#### [데이터]

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

#### [함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)



## 5. 클래스: 매개변수/지역변수

```
1 class Car:
2     def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```

### 자동차

#### [데이터]

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

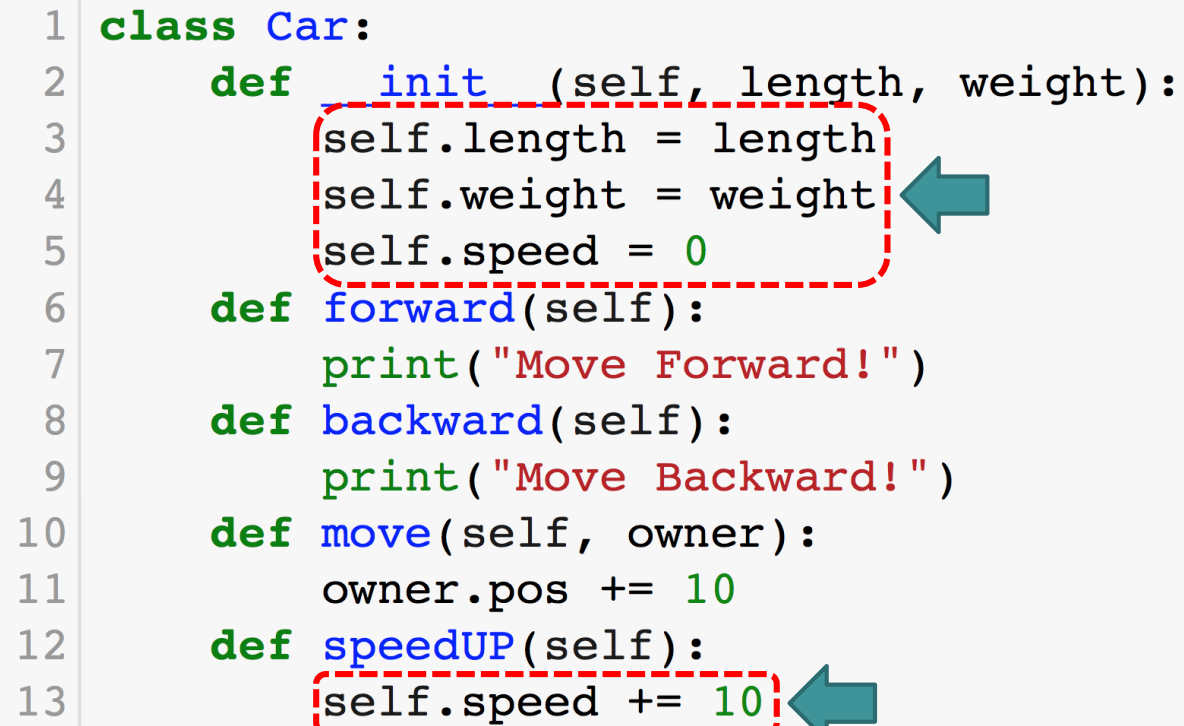
#### [함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)



## 5. 클래스: 인스턴스 변수

```
1 class Car:
2     def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```



### 자동차

#### [데이터]

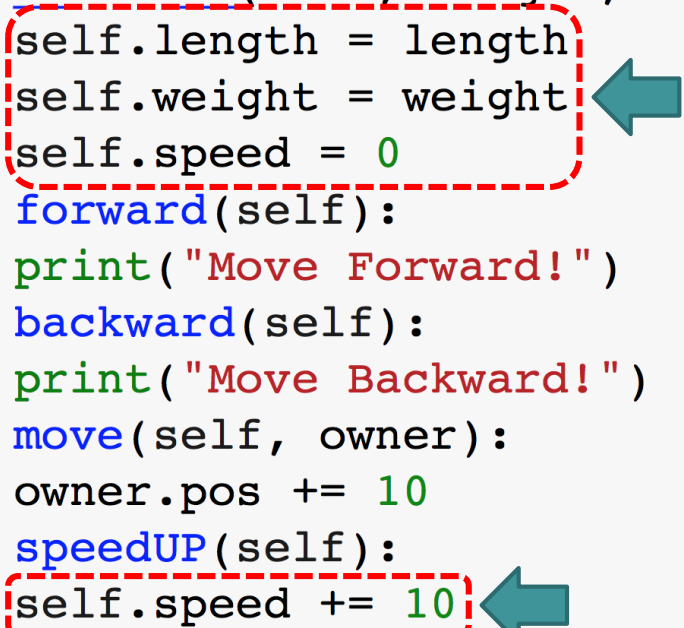
- 길이 (length)
- 무게 (weight)
- 속도 (speed)

#### [함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 5. 클래스: 인스턴스 변수

```
1 class Car:
2     def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```



자동차

**[인스턴스 변수]**

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

**[함수]**

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 5. 클래스: 객체가 사용할 함수

```
1 class Car:
2     ➡ def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     ➡ def forward(self):
7         print("Move Forward!")
8     ➡ def backward(self):
9         print("Move Backward!")
10    ➡ def move(self, owner):
11        owner.pos += 10
12    ➡ def speedUP(self):
13        self.speed += 10
```

### 자동차

#### [인스턴스 변수]

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

#### [함수]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 5. 클래스: 메소드

```
1 class Car:
2     def __init__(self, length, weight):
3         self.length = length
4         self.weight = weight
5         self.speed = 0
6     def forward(self):
7         print("Move Forward!")
8     def backward(self):
9         print("Move Backward!")
10    def move(self, owner):
11        owner.pos += 10
12    def speedUP(self):
13        self.speed += 10
```

### 자동차

#### [인스턴스 변수]

- 길이 (length)
- 무게 (weight)
- 속도 (speed)

#### [메소드]

- 앞으로 (forward)
- 뒤로 (backward)
- 이동 (move)
- 속도증가 (speedUP)

## 6. 클래스 메소드: 생성자 (\_\_init\_\_)

- 클래스로부터 객체 생성
- 객체 초기화

```
mycar = Car(4, 1380)
```




```
def __init__(self, length, weight):  
    self.length = length  
    self.weight = weight  
    self.speed = 0
```

## 6. 클래스 메소드: 매개변수 self

---

```
mycar = Car(4, 1380)
```



```
def __init__(self, length, weight):  
    self.length = length  
    self.weight = weight  
    self.speed = 0
```

## 6. 클래스 메소드: 매개변수 self

- 객체 자기 자신

```
def change(self, length, weight):  
    self.length = length  
    self.weight = weight
```

```
mycar.change(3.5, 1200)
```



# 객체 지향 프로그래밍

---

- 학습 정리
  - 객체 지향 프로그래밍의 개념
  - 객체와 클래스 및 생성자
  - 인스턴스 변수와 메소드
  - **self**와 매개변수