

# 교수의 파이썬

01\_5 동적 타이핑 vs 정적 타이핑

창원대학교 정보통신공학과 교수 박동규

# 넌넌한 교수의 파이썬

01\_5 동적 타이핑 vs 정적 타이핑

창원대학교 정보통신공학과 교수 박동규

# 넌넌한 교수의 고급 파이썬

01\_5 동적 타이핑 vs 정적 타이핑

창원대학교 정보통신공학과 교수 박동규

# 넌넌한 교수의 고급 파이썬

01\_5 동적 타이핑 vs 정적 타이핑

창원대학교 정보통신공학과 교수 박동규

# 동적 타이핑

# 동적 타이핑

# 리스트 객체 생성, 변수 a가 이 객체를 참조함

# 동적 타이핑

# 리스트 객체 생성, 변수 a가 이 객체를 참조함

**a = [1, 2, 3, 4]**

# 동적 타이핑

# 리스트 객체 생성, 변수 a가 이 객체를 참조함

**a = [1, 2, 3, 4]**

# 변수 a가 문자열 객체를 참조함



# 동적 타이핑

# 리스트 객체 생성, 변수 a가 이 객체를 참조함

**a = [1, 2, 3, 4]**

# 변수 a가 문자열 객체를 참조함

**a = 'Hello'**

# 동적 타이핑

# 리스트 객체 생성, 변수 a가 이 객체를 참조함

**a = [1, 2, 3, 4]**

# 변수 a가 문자열 객체를 참조함

**a = 'Hello'**

- 파이썬 변수의 형은 실행시에 결정된다.
- 변수는 객체에 대해서 바운드되며 객체는 임의의 형이 될 수 있다.
  - 바인딩 : 변수에 변수와 관련된 속성을 연관시키는 것을 말함
- 만일 한 자료형의 객체에 변수 이름 a가 할당된다면, 나중에 이 이름 a는 다른 자료형의 객체를 참조하는데 사용될 수 있다.

# Lab

```
[>>> a = [1, 2, 3, 4]
[>>> type(a)
<class 'list'>
[>>> a = 'Hello'
[>>> type(a)
<class 'str'>
[>>> a = 100
[>>> type(a)
<class 'int'>
```

# 정적 타이핑

# 정적 타이핑

- C, C++, Java와 같은 프로그래밍 언어에서는 `char a ;` 과 같은 선언을 통해서 참조하는 형이 고정된다. 따라서 `a = 3.14;` 와 같이 할당 연산을 사용할 수 없다

# 정적 타이핑

- C, C++, Java와 같은 프로그래밍 언어에서는 `char a ;` 과 같은 선언을 통해서 참조하는 형이 고정된다. 따라서 `a = 3.14;` 와 같이 할당 연산을 사용할 수 없다
- 이러한 종류의 타이핑(자료형 결정방식)을 정적 타이핑이라 한다.

# 정적 타이핑

- C, C++, Java와 같은 프로그래밍 언어에서는 `char a ;` 과 같은 선언을 통해서 참조하는 형이 고정된다. 따라서 `a = 3.14;` 와 같이 할당 연산을 사용할 수 없다
- 이러한 종류의 타이핑(자료형 결정방식)을 정적 타이핑이라 한다.
- 정적 타이핑 언어는 변수의 자료형이 컴파일할 때 결정된다.

# 정적 타이핑

- 변수를 사용하기에 앞서 변수의 형을 선언해야 한다
- 변수의 형을 실행할 때 변경할 수 없다
- 변수는 사용기간동안 오직 한가지 자료형에 대한 참조만 가능하다

// C언어

```
int n = 100;
```

# 동적 타이핑

- 변수의 형 선언자체가 필요없다
- 변수의 형을 실행시에 변경할 수 있다
- 변수는 사용기간동안 다른 자료형을 참조하는 것이 가능하다

# 파이썬 언어

```
n = 100
```

```
...
```

```
n = 'Hello'
```





감사합니다.