

4주차 2차시

프로그래밍 언어(2)

인간과 컴퓨터의 대화_진화하는 소통·진화하는 컴퓨터

- 1 절차 지향 언어의 프로그래밍
- 2 객체 지향 언어의 프로그래밍
- 3 교육용 프로그래밍 언어



프로그래밍 언어(2)

◆ 학습목표

- 절차 지향 언어와 객체 지향 언어의 특성과 사용 절차를 알아본다.
- 교육용 프로그래밍 언어의 특성과 사용 절차를 알아본다.

1. 절차 지향 언어의 프로그래밍

1) 절차 지향 언어의 개념

◆ 절차 지향 언어의 개념

- 프로그램 코드가 순서대로 실행되는 언어
- 파스칼, 코볼, 포트란, 베이직, C언어 등

1. 냉장고 문을 연다.
2. 소고기를 넣는다.
3. 냉장고 문을 닫는다.

(a) 냉장고에 소고기를 넣는 과정

1. open 냉장고
2. insert 소고기
3. close 냉장고

(b) 냉장고에 소고기를 넣는 프로그램

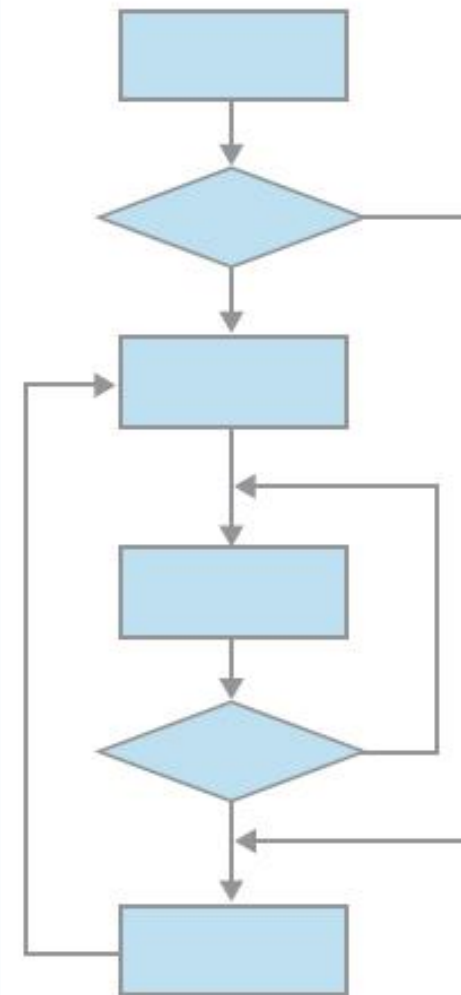
그림 4-18 절차 지향 언어의 프로그래밍 개념

1. 절차 지향 언어의 프로그래밍

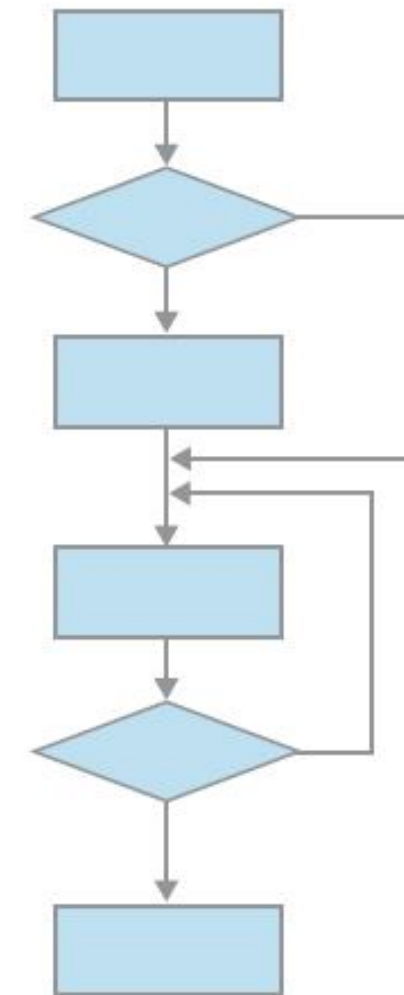
2) 구조적 프로그래밍의 이해

◆ 구조적 프로그래밍 등장 배경

- goto문의 무분별한 분기 구조를 개선하고 모든 명령문의 처리를 블록으로 모듈화시키기 위해 등장



(a) 나쁜 프로그램



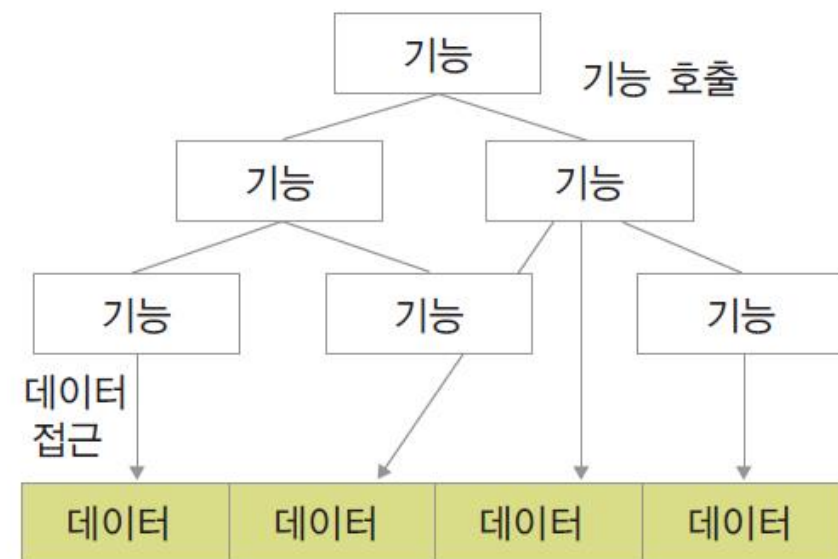
(b) 좋은 프로그램

그림 4-19 구조적 프로그래밍의 예

2. 객체 지향 언어의 프로그래밍

1) 절차 지향 언어와 객체 지향 언어의 차이점

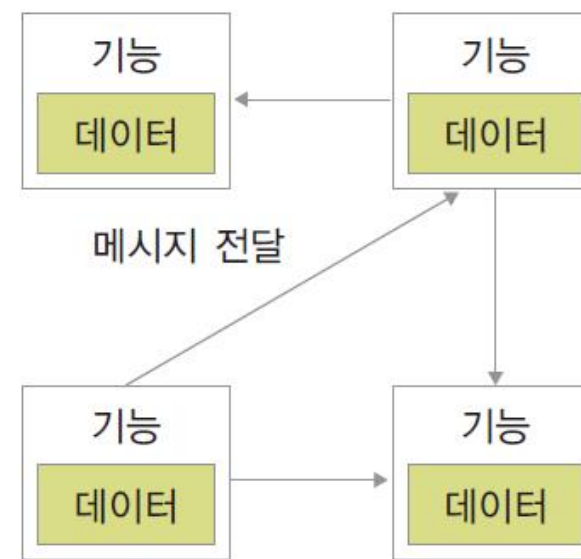
- ◆ 절차 지향 언어 : 데이터와 데이터를 처리하는 기능이 별도로 관리
- ◆ 객체 지향 언어 : 데이터와 기능을 묶어 캡슐화시킨 후 메시지를 전달하여 일을 처리



기능과 데이터의 불일치

빈약한 유지 보수성

(a) 절차 지향 언어



기능과 데이터 캡슐화

개선된 유지 보수성

(b) 객체 지향 언어

그림 4-20 절차 지향 언어와 객체 지향 언어의 차이점

2. 객체 지향 언어의 프로그래밍

2) 객체 지향 언어의 주요 개념

◆ 클래스

- 다른 사물과 구분되는 속성을 가진 객체가 모여, 일반화된 범주로 묶인 것

◆ 객체

- 개별적으로 식별되는 사물을 지칭
- 속성과 기능을 캡슐화 함

◆ 상속

- 하위 클래스는 상위 클래스가 가지는 속성과 기능을 모두 이어받을 수 있는데 이를 상속이라고 함

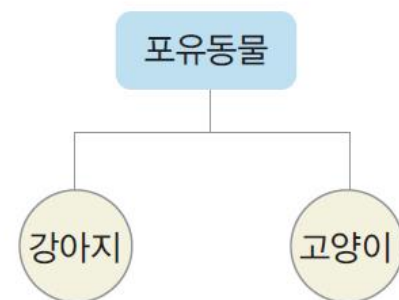


그림 4-22 상속의 예

2. 객체 지향 언어의 프로그래밍

2) 객체 지향 언어의 주요 개념

◆ 메시지

- 객체 간에 전달되는 명령 단위



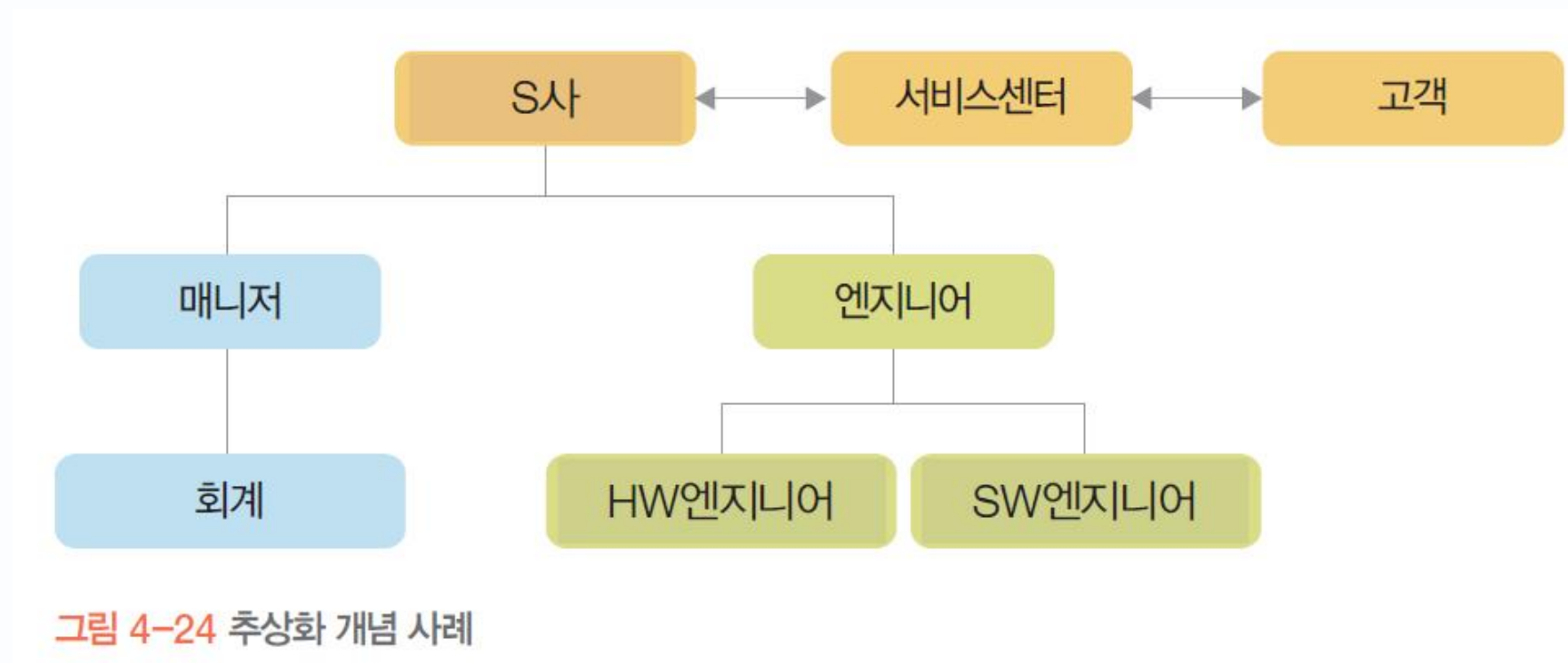
그림 4-23 객체 간의 메시지 송수신

2. 객체 지향 언어의 프로그래밍

2) 객체 지향 언어의 주요 개념

◆ 추상화

- 어떤 객체가 상대하는 다른 객체에 대해, 꼭 필요한 부분만 알고 나머지 세부적인 사항은 감추는 것



◆ 캡슐화

- 객체에 속성과 기능을 포함하면서 추상화 개념을 통해 객체의 세부내용은 사용자로부터 은폐하는 것

2. 객체 지향 언어의 프로그래밍

2) 객체 지향 언어의 주요 개념

◆ 다형성

- 일반화된 클래스는 어떤 특정화된 클래스 객체를 지칭할 수 있기 때문에 같은 동작을 함, 하지만 각각 특정화된 클래스는 다른 성질을 가질 수 있음
- 이런 성질을 다형성이라 함

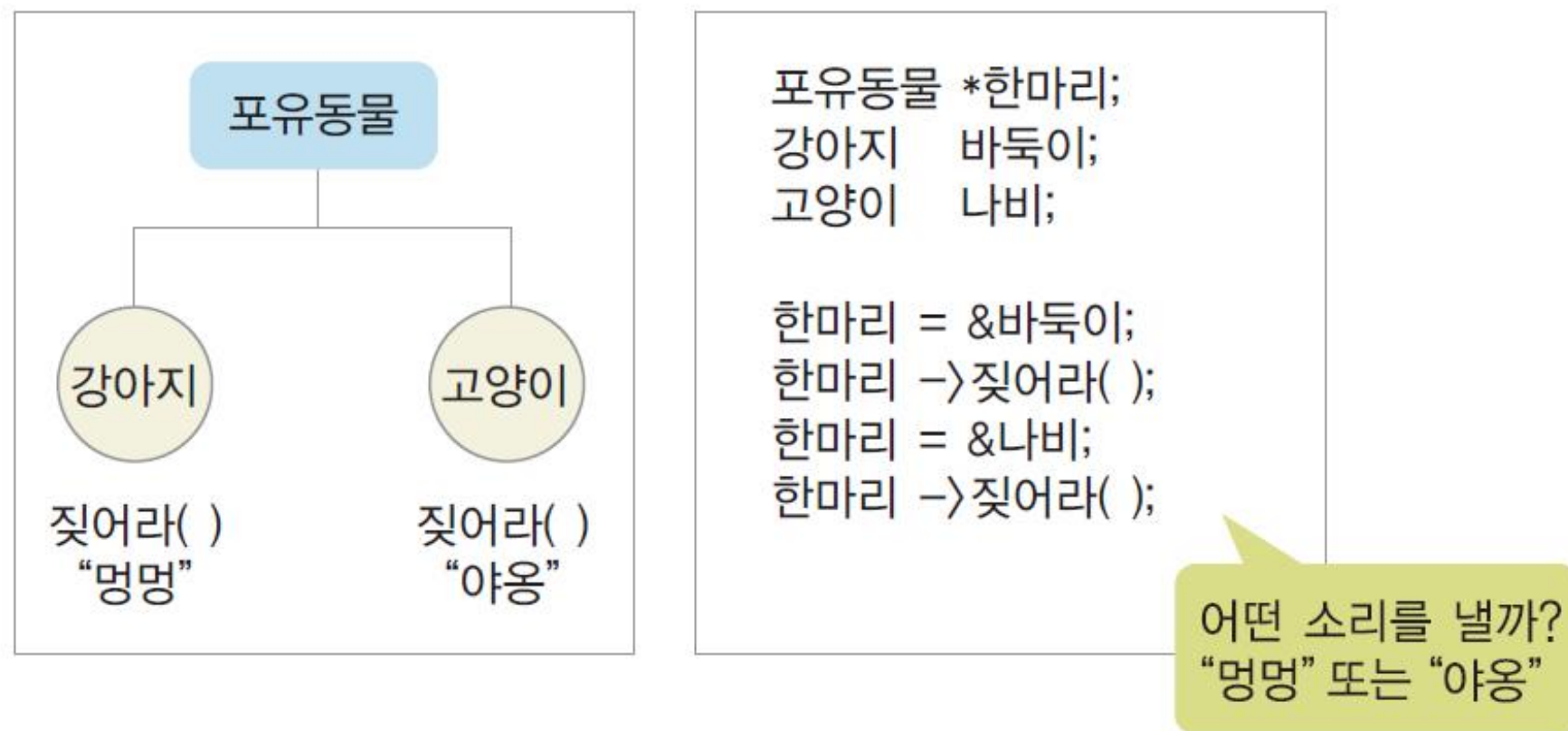


그림 4-25 다형성의 예

3. 교육용 프로그래밍 언어

1) 교육용 프로그래밍 언어의 종류

- ◆ 스크래치
- ◆ 파이썬
- ◆ 프로그래민
- ◆ 스몰베이직
- ◆ 스킵
- ◆ 이토이즈
- ◆ 엔트리

3. 교육용 프로그래밍 언어

2) 스크래치 역사

- ◆ 2005년에 미국의 MIT의 미디어랩(Media Lab.) 소속의 킨더카튼그룹(Lifelong Kindergarten Group)에서 발표한 교육용 프로그래밍 언어
- ◆ 이미 1.4와 2.0버전을 거쳐서 현재 3.0 버전이 발표되어 사용
- ◆ 초등학생이나 중등학생 등과 같은 낮은 연령대를 대상으로 프로그래밍(코딩)을 교육하기 위한 도구



3. 교육용 프로그래밍 언어

3) 스크래치 특징

- ◆ 블록 명령어
- ◆ 통합 개발 환경
- ◆ 풍부한 멀티미디어 콘텐츠 제공
- ◆ 스프라이트 중심의 스크립트
- ◆ 교육용 시스템

3. 교육용 프로그래밍 언어

3) 스크래치 특징

- ◆ 복잡한 규모가 큰 소프트웨어 제작에는 적합하지 않음
- ◆ 문자 중심의 입출력 기능이 약하다
- ◆ 작업을 위한 데이터 저장에 필수적인 파일 처리 등의 기능이 없기 때문에
사용화된 응용 프로그램을 만들기 어려움

3. 교육용 프로그래밍 언어

4) 스크래치 실습

◆ 따라하기 : 스크래치로 간단한 프로그램 만들기

①



그림 4-26 스크래치 사이트 접속

②



그림 4-27 스크래치 프로그래밍 시작

3. 교육용 프로그래밍 언어

4) 스크래치 실습

◆ 따라하기 : 스크래치로 간단한 프로그램 만들기

③

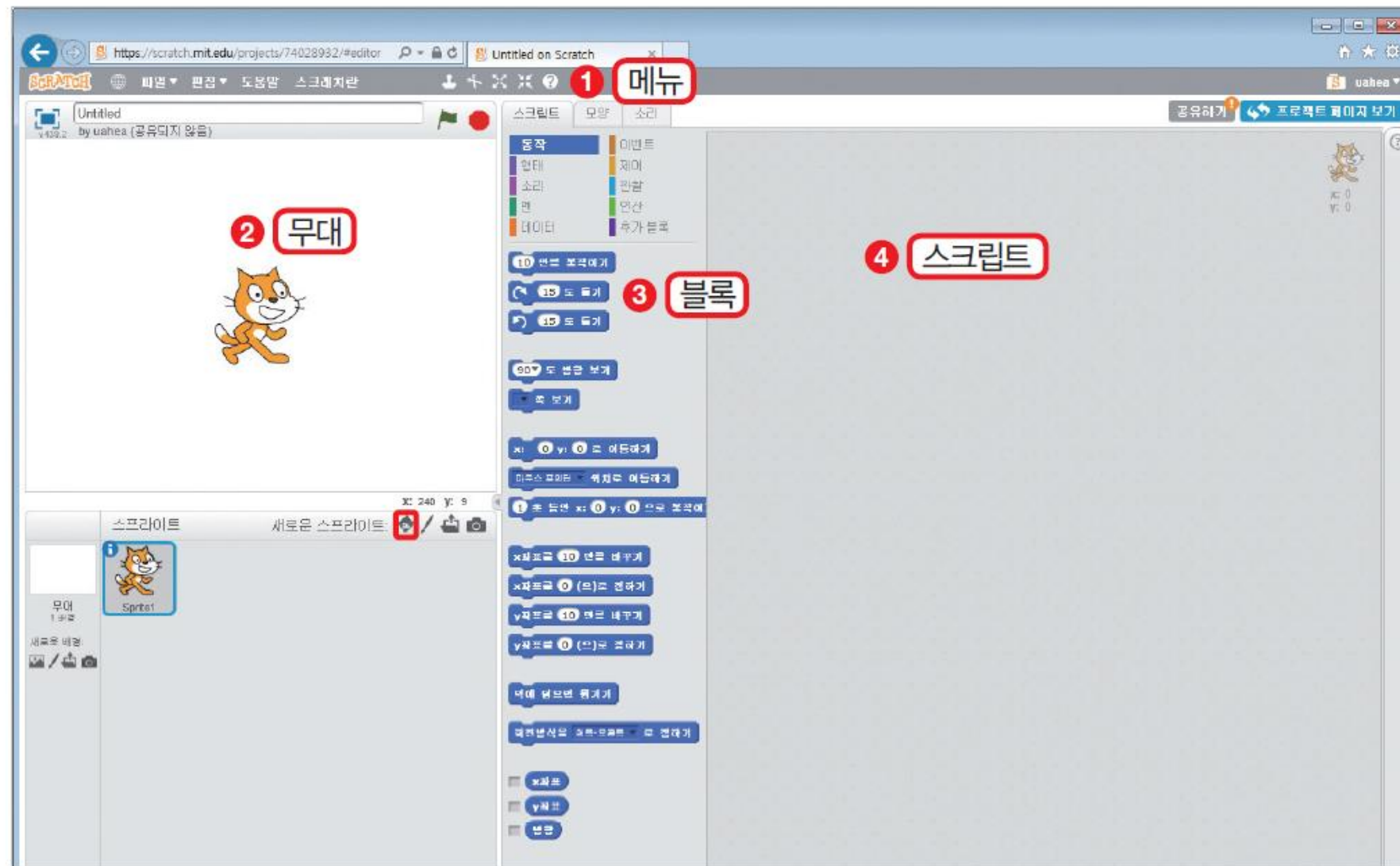


그림 4-28 스크래치 초기 화면

3. 교육용 프로그래밍 언어

4) 스크래치 실습

◆ 따라하기 : 스크래치로 간단한 프로그램 만들기

④

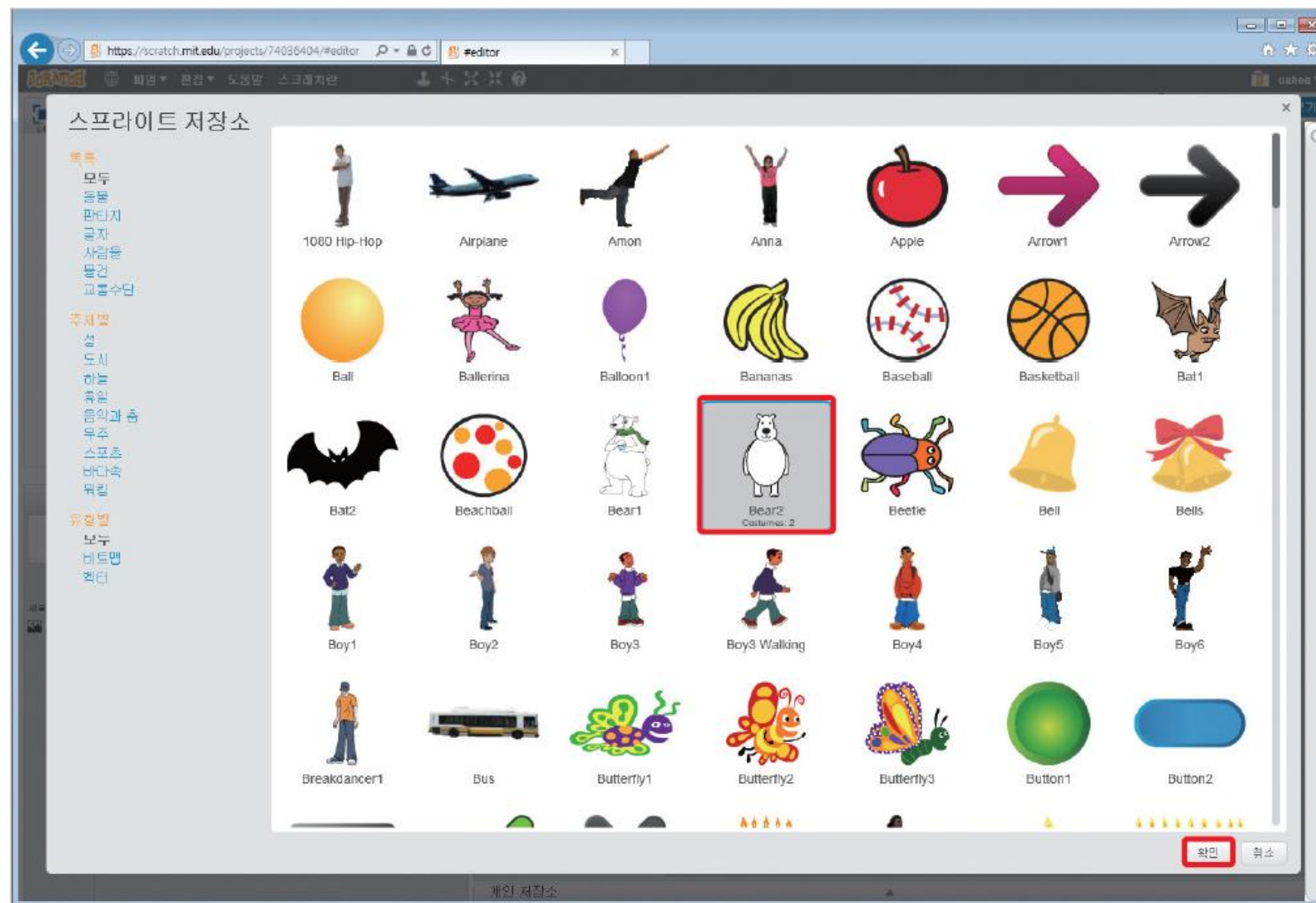


그림 4-29 새 스프라이트 추가

3. 교육용 프로그래밍 언어

4) 스크래치 실습

◆ 따라하기 : 스크래치로 간단한 프로그램 만들기

⑤



그림 4-30 스프라이트 삭제

⑥

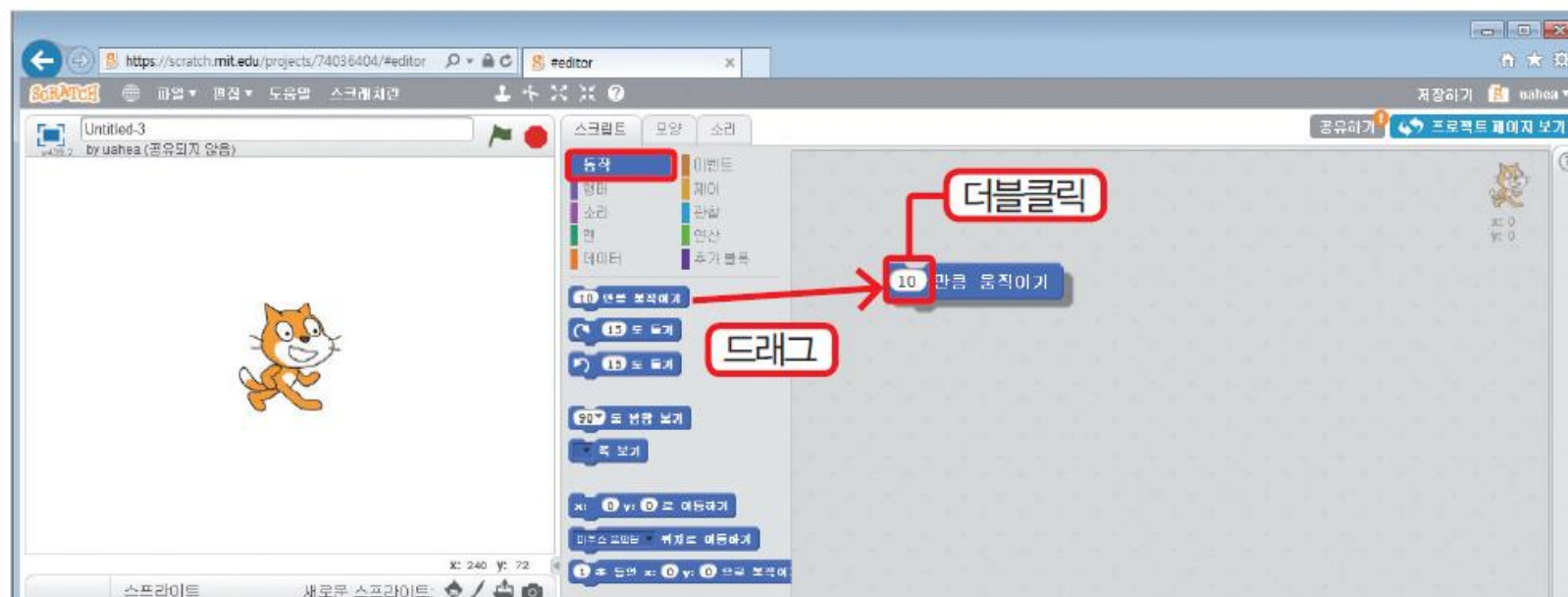


그림 4-31 동작 추가

3. 교육용 프로그래밍 언어

4) 스크래치 실습

◆ 따라하기 : 스크래치로 간단한 프로그램 만들기

⑦



그림 4-32 회전 동작 추가

⑧

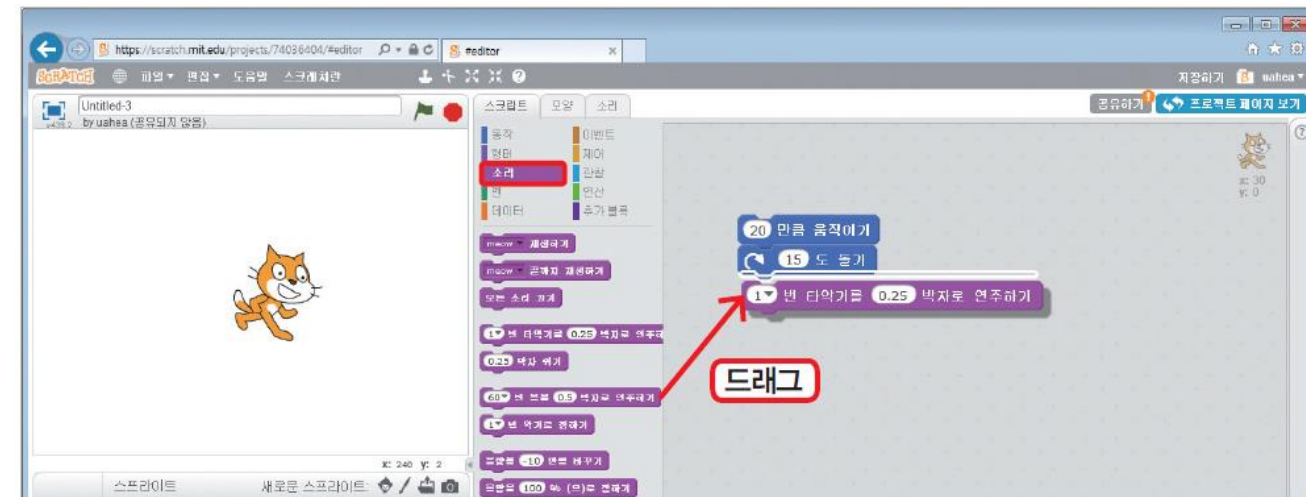


그림 4-33 소리 추가

3. 교육용 프로그래밍 언어

4) 스크래치 실습

◆ 따라하기 : 스크래치로 간단한 프로그램 만들기

⑨

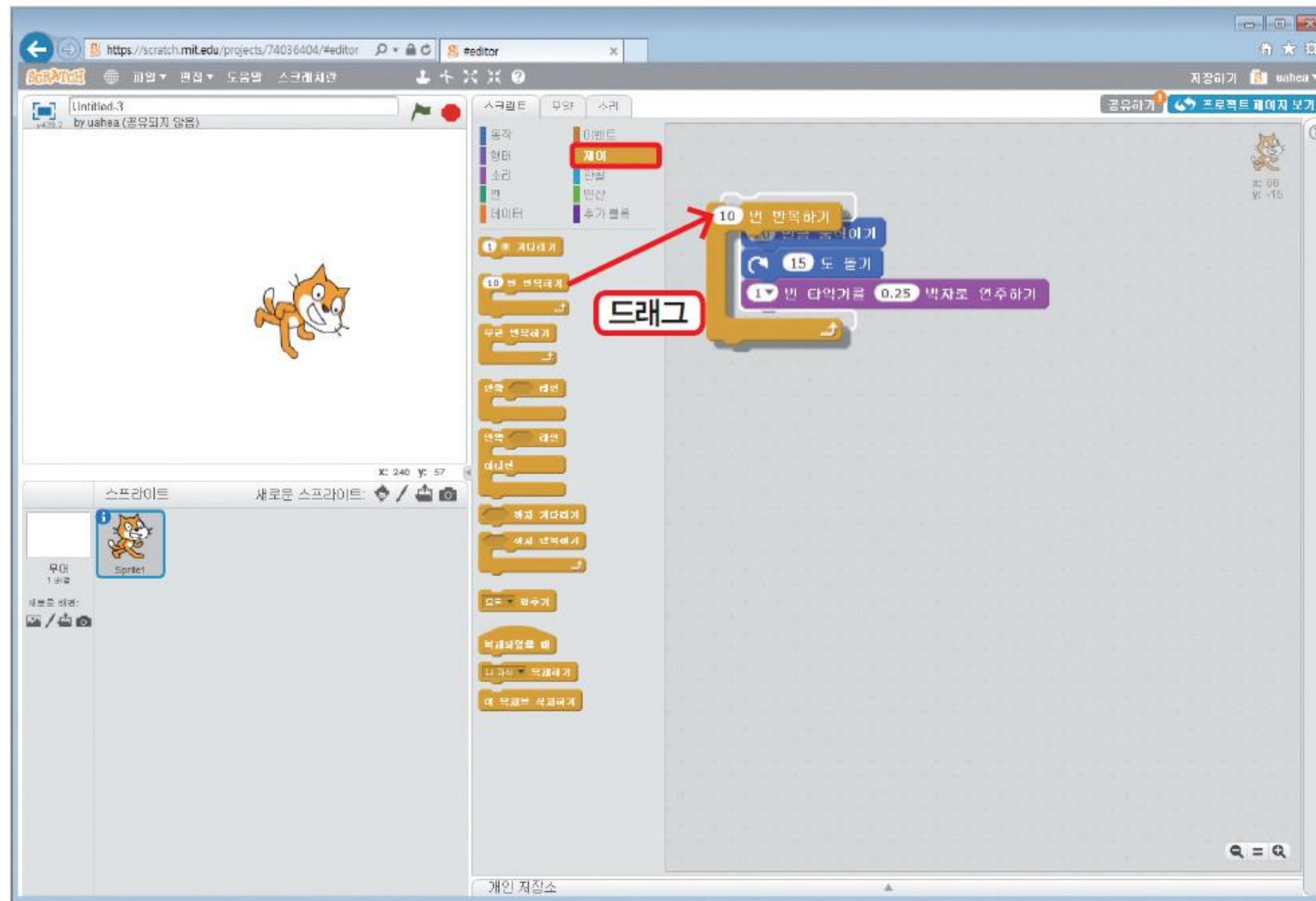


그림 4-34 반복 기능

3. 교육용 프로그래밍 언어

5) 파이썬

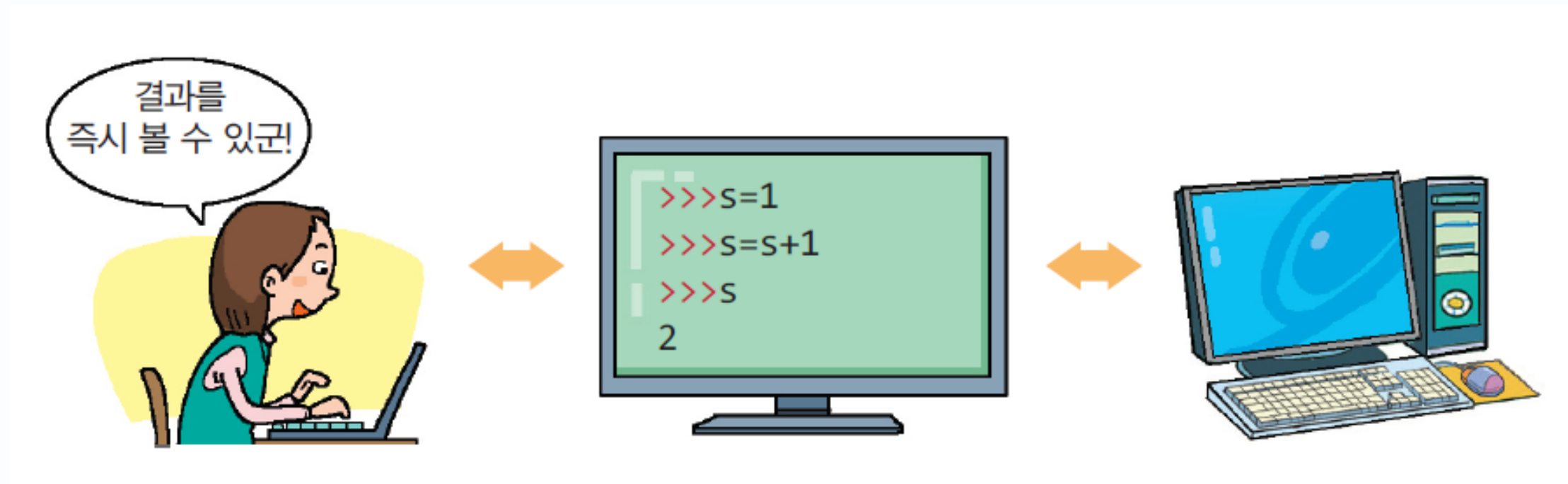
- ◆ 1991년에 귀도 반 로섬(Guido van Rossum)이 개발한 대화형 프로그래밍 언어



3. 교육용 프로그래밍 언어

6) 파이썬의 특징

- ◆ 생산성이 뛰어나다
- ◆ 초보자한테 좋은 언어 – 인터프리터 언어



3. 교육용 프로그래밍 언어

6) 파이썬의 특징

- ◆ 파이썬은 문법이 쉬워서 코드를 보면 직관적으로 알 수 있는 부분이 많다.

```
if "사과" in ["딸기", "바나나", "포도", "사과"]:  
    print("사과가 있습니다")
```

- ◆ 파이썬은 다양한 플랫폼에서 사용
- ◆ 라이브러리가 풍부
- ◆ 애니메이션이나 그래픽을 쉽게 사용

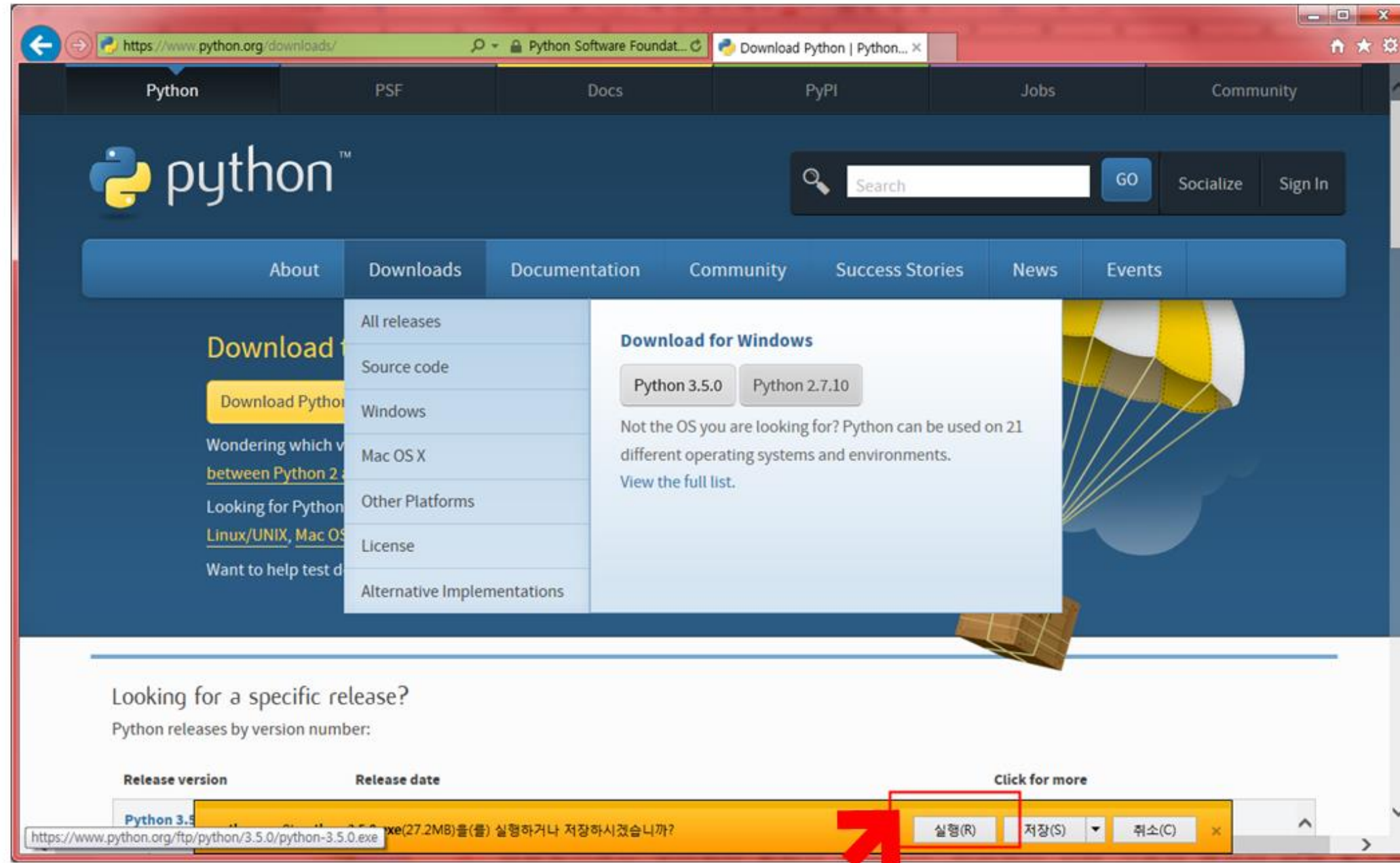
3. 교육용 프로그래밍 언어

7) 파이썬을 사용하고 있는 기업들



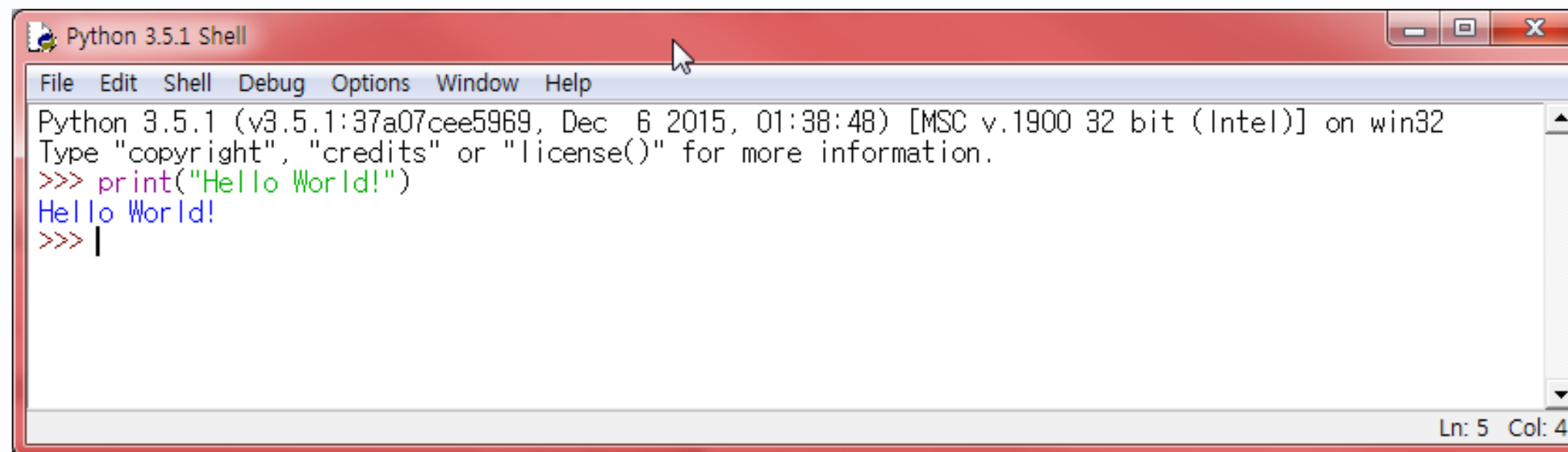
3. 교육용 프로그래밍 언어

8) 파이썬 연습



3. 교육용 프로그래밍 언어

8) 파이썬 연습



The image shows a screenshot of a Python 3.5.1 Shell window. The window has a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area displays the following content:

```
Python 3.5.1 (v3.5.1:37a07cee5969, Dec 6 2015, 01:38:48) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print("Hello World!")
Hello World!
>>> |
```

The status bar at the bottom right indicates 'Ln: 5 Col: 4'.





요약

◆ 프로그램 제어 흐름의 유형

- 순차구조: 프로그램 코드 순서대로 실행하는 구조이다.
- 선택구조: 조건문이라고도 하며 프로그램이 다음에 무엇을 해야 하는지를 결정하는 분기 구조이다.
- 반복구조: 루프 구조라고도 하며 조건이 만족하지 않을 때 까지 계속 반복한다.

◆ 절차지향 언어

- 프로그램이 순서대로 실행되는 언어이다. 함수를 이용해서 다른 곳으로 건너뛰었다 돌아와도 그 전체의 흐름은 순서대로 흘러가는 특징이 있다.

◆ 객체지향 언어

- 절차 지향언어의 제한점인 재사용성을 보완하기 위해 등장한 언어이다. 객체, 클래스, 상속, 메시지, 추상화, 캡슐화, 다형성 등의 개념이 추가됐다.