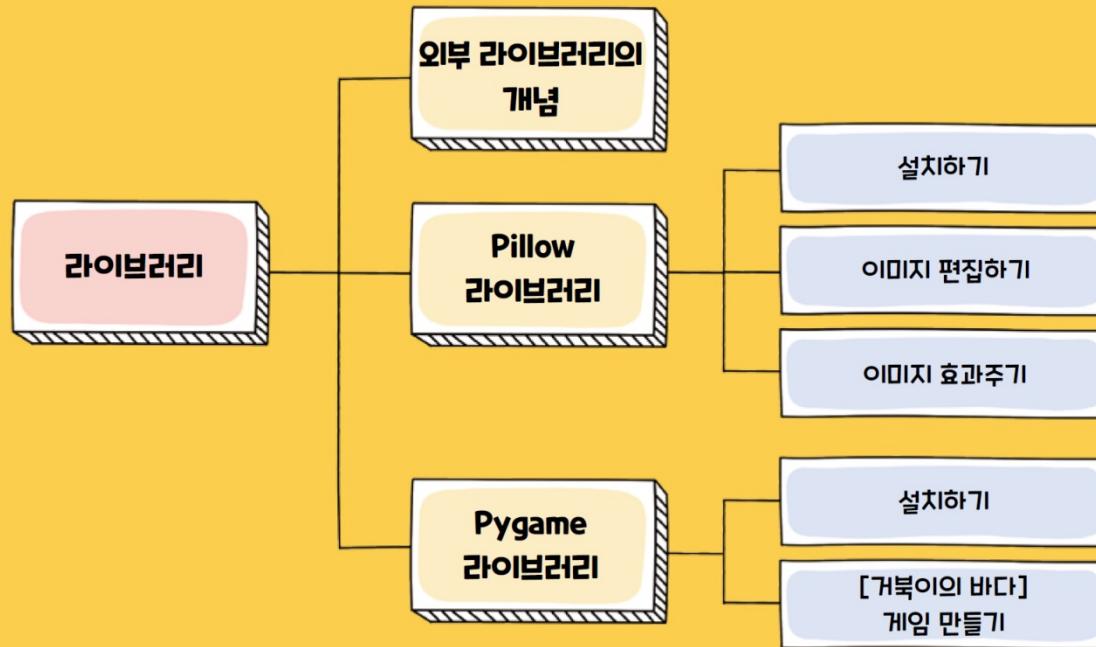


Chapter 12

Pillow and Pygame

임경태





학습목표

- 외부 라이브러리의 개념을 파악합니다.
- Pillow 라이브러리를 설치해서 이미지 처리 프로그램을 작성합니다.
- Pygame 라이브러리를 설치해서 간단한 게임을 만듭니다.
- 거북이 그래픽과 Pillow를 조합한 응용 프로그램을 작성합니다

Section 01

Pillow 라이브러리

외부 라이브러리의 개념



■ 외부 라이브러리란?

- 파이썬 자체에는 없는 기능이지만, 다른 개발자가 좋은 기능들을 만들어 제공하는 뮤음을 의미함
- 대부분의 외부 라이브러리는 완전히 무료로 사용할 수 있음

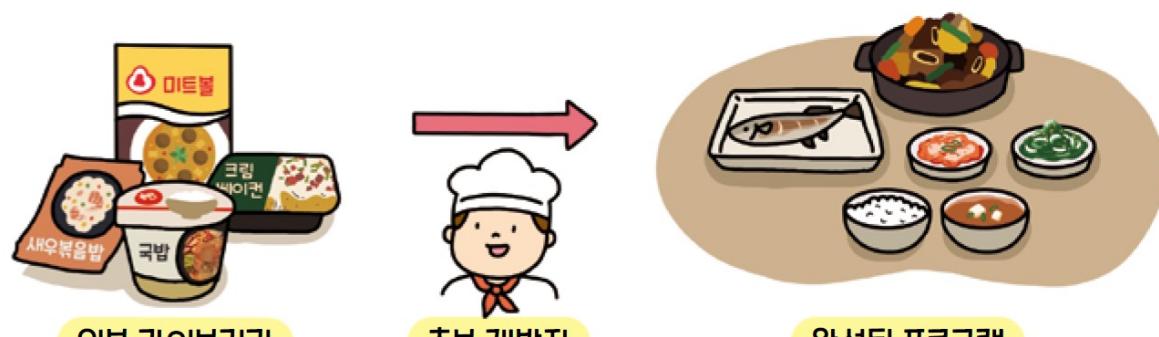


그림 11-2 외부 라이브러리와 요리의 비교

외부 라이브러리의 개념



■ 외부 라이브러리의 종류

- 파이썬 외부 라이브러리의 종류는 수십만 개 이상임
 - 게임/네트워크 프로그래밍/고급 GUI/이미지 처리/데이터베이스 연동 등
- 파이썬 외부 라이브러리는 <https://pypi.org/>에서 검색할 수 있음

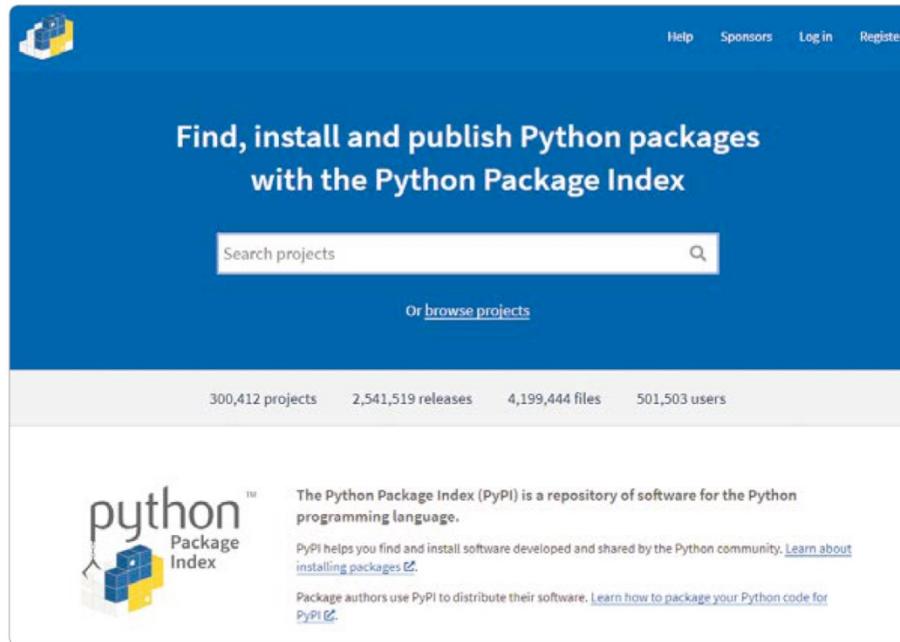


그림 11-3 pypi 사이트에서 라이브러리 검색

외부 라이브러리의 개념



- 외부 라이브러리로 간단한 이미지 처리 프로그램 만들기
 - 이미지를 처리하는 Pillow(필로우)라는 라이브러리를 사용함

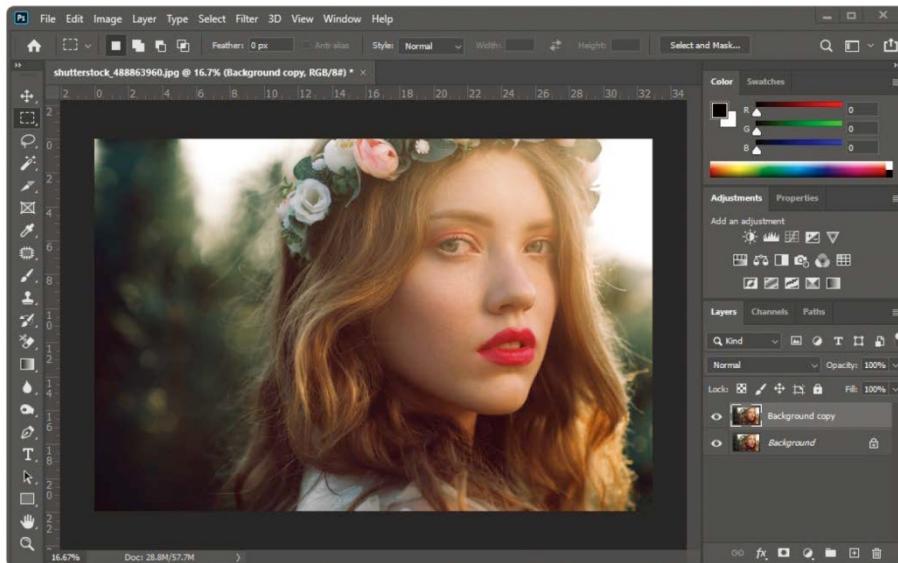


그림 11-1 포토샵 화면



Pillow(필로우)는 영상 처리(Image Processing)과 관련된 다양한 기능을 제공합니다. 영상 처리 기능을 제공하는 라이브러리로는 Pillow 외에도 OpenCV, ImageMagick 등이 있습니다.

외부 라이브러리의 개념



확인문제

다음 빈칸에 들어갈 단어를 채우시오.

파이썬 자체에는 없는 기능이지만, 훌륭한 기능들을 만들어서 제공하는 외부 기능들을 ([])라고 부른다.

정답

Click!

Pillow(필로우) 설치하기



■ 외부 라이브러리를 설치하는 편리한 방법

- ① 외부 라이브러리를 다운로드하여 설치하기
- ② 명령어로 바로 설치하기

직접 장보기



다운로드해서 설치

인터넷 쇼핑



명령어로 바로 설치

그림 11-4 설치 방법 비교

Pillow(필로우) 설치하기



■ Pillow(필로우) 설치

- '명령 프롬프트'를 실행한 후 다음 명령어를 <Enter> 누르기

```
pip install pillow
```

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\WinUser> pip install pillow
Collecting pillow
  Downloading Pillow-8.2.0-cp39-cp39-win_amd64.whl (2.2 MB)
    ██████████| 2.2 MB 2.2 MB/s
Installing collected packages: pillow
Successfully installed pillow-8.2.0
WARNING: You are using pip version 20.2.3; however, version 21.0.1 is available.
You should consider upgrading via the 'c:\Users\WinUser\AppData\Local\Programs\Python\Python39\python.exe -m pip install --upgrade pip' command.

C:\Users\WinUser>
```

그림 11-5 Pillow 라이브러리 설치

Pillow(필로우) 설치하기



설치 확인

- 파이썬 IDLE을 열고, import PIL을 입력하기
 - 아무런 메시지도 표시되지 않는다면 Pillow 라이브러리가 제대로 설치된 것임
 - 설치에 문제가 있다면 오류 메시지가 표시됨

A screenshot of the Python IDLE Shell window. The title bar says "IDLE Shell 3.9.2". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main window shows the Python interpreter prompt:

```
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
>>> import PIL
>>>
```

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

그림 11-6 Pillow 라이브러리 설치 확인

Pillow(필로우) 설치하기



하나 더 알기

파이썬과 외부 라이브러리의 버전이 맞지 않을 때 해결 방법

Pillow 및 Pygame은 파이썬과 잘 맞는 버전이 설치되지 않으면 책의 코드가 작동하지 않는 경우도 생길 수 있습니다. 만약 Pillow 또는 Pygame이 설치되지 않거나, 잘 작동하지 않는다면 다음과 같이 진행해 보세요.

- (1) Windows [시작]에서 마우스 오른쪽 버튼을 클릭해서, [앱 및 기능]을 실행하고 [Python 3.x.x]를 제거합니다.
- (2) 파일 탐색기에서 C:\사용자(또는 users)\현재사용자아이름\appdata\local\programs\python\ 폴더를 삭제합니다
(먼저 파일 탐색기 메뉴의 [보기]-[숨김 항목]을 선택해야 폴더가 보입니다).
- (3) 이 책의 자료실(<http://hanbit.co.kr/src/4552>)에서 교재와 동일한 Python 3.9.2(64-bit)용을 다운로드하고 설치합니다(꼭 1장을 참조해서 설치하세요).
- (4) 이 책의 자료실에서 Pillow-8.2.0-cp39-cp39-win_amd64.whl 및 pygame-2.0.1-cp39-cp39-win_amd64.whl 파일을 다운로드하고 적당한 폴더로 이동시킵니다(예를 들어 C:\temp\ 폴더를 만들어서 이동시키세요).
- (5) **Windows**+**R**을 누른 후, cmd를 입력해서 명령 프롬프트를 열고 다음 명령을 차례로 수행하세요.

Pillow(필로우) 설치하기



```
cd c:\temp  
pip install pillow Tab → Pillow-8.2.0-cp39-cp39-win_amd64.whl이 자동 완성됨  
pip install pygame Tab → pygame-2.0.1-cp39-cp39-win_amd64.whl이 자동 완성됨
```

```
명령 프롬프트  
Microsoft Windows [Version 10.0.19042.928]  
(c) Microsoft Corporation. All rights reserved.  
C:\Users\WinUser> cd c:\temp  
c:\temp> pip install Pillow-8.2.0-cp39-cp39-win_amd64.whl  
Processing .\Pillow-8.2.0-cp39-cp39-win_amd64.whl  
Installing collected packages: Pillow  
Successfully installed Pillow-8.2.0 ←  
WARNING: You are using pip version 20.2.3; however, version 21.1.1 is available.  
You should consider upgrading via the 'c:\Users\winuser\AppData\Local\Programs\Python\Python39\python.exe -m pip install --upgrade pip' command.  
c:\temp> pip install pygame-2.0.1-cp39-cp39-win_amd64.whl  
Processing .\pygame-2.0.1-cp39-cp39-win_amd64.whl  
Installing collected packages: pygame  
Successfully installed pygame-2.0.1 ←  
WARNING: You are using pip version 20.2.3; however, version 21.1.1 is available.  
You should consider upgrading via the 'c:\Users\winuser\AppData\Local\Programs\Python\Python39\python.exe -m pip install --upgrade pip' command.  
c:\temp>
```





사진 준비하기

■ 이미지 처리하기 전에 미리 사진 준비하기

- 1000x1000 픽셀 이하의 이미지 준비하기
 - tip) 책의 자료실(<http://hanbit.co.kr/src/4552>)에서 제공하는 사진 사용하기
- C 드라이브에 [photo] 폴더를 생성하고 이곳에 저장하기
- Pillow로 처리가 가능한 사진의 확장명
 - bmp, png, jpg, jpeg, tif, tiff 등

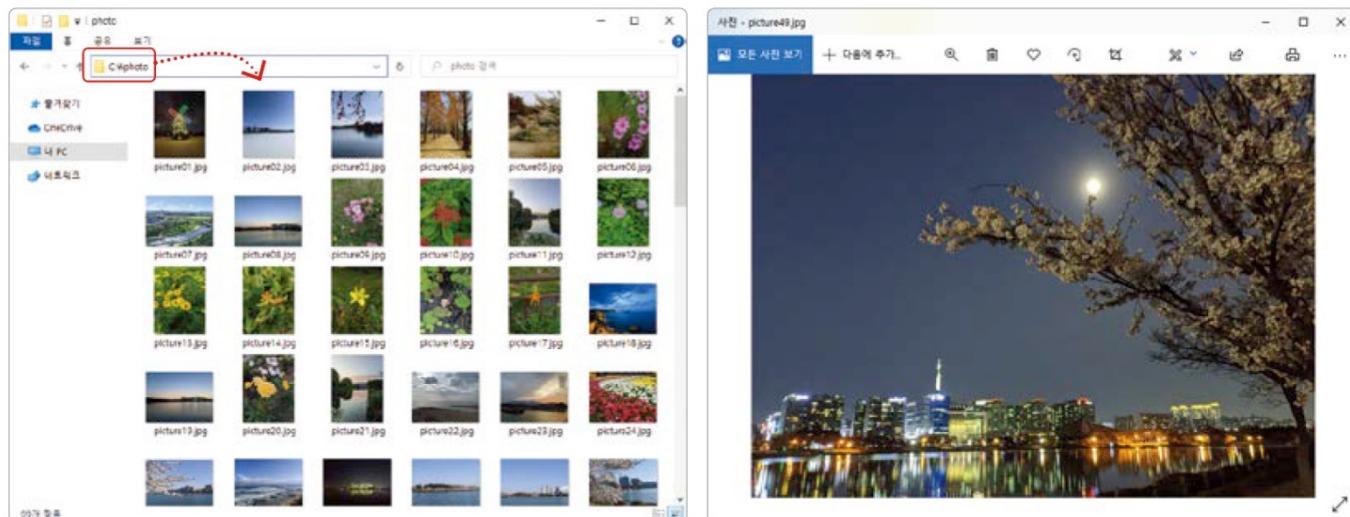


그림 11-7 처리할 사진의 준비

사진 준비하기



확인문제

1. 다음 빈칸에 들어갈 단어를 채우시오.

명령 프롬프트에서 Pillow 라이브러리를 설치하는 명령어는 다음과 같습니다.

`install pillow`

2. 다음 중 Pillow 라이브러리에서 처리가 가능한 확장명이 아닌 것을 고르시오.

- | | |
|-------|-------|
| ① JPG | ② BMP |
| ③ PNG | ④ HWP |

정답

Click!

Pillow(필로우)의 기초 다지기



■ PIL 라이브러리 임포트

- PIL이란?
 - Pillow 라이브러리는 전체 이름을 의미함
- PIL 안에는 Image, ImageFilter, ImageEnhance, ImageOps 등의 분류로 나눠 기능을 제공함

```
from PIL import Image, ImageFilter, ImageEnhance, ImageOps
```

Pillow(필로우)의 기초 다지기



■ 이미지 열고 저장하기

- 이미지 열기 : `Image.open(파일경로)`

```
>>> img = Image.open("c:/photo/picture01.jpg")
>>> img.show()
```



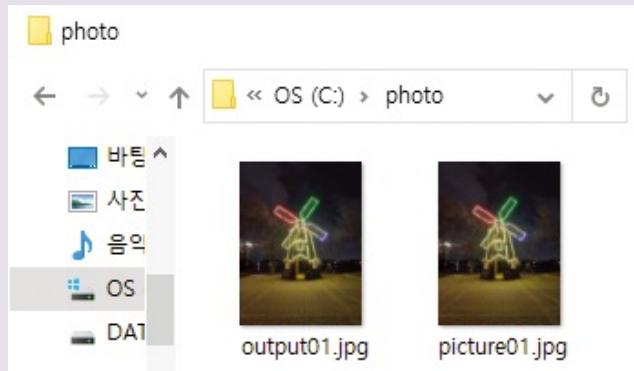
Pillow(필로우)의 기초 다지기



■ 이미지 열고 저장하기

- 이미지 저장하기 : 변수.save(파일경로)
 - 파일 탐색기의 C:/photo 폴더에서 output01.jpg 파일을 확인할 수 있음

```
>>> img.save("c:/photo/output01.jpg")
```



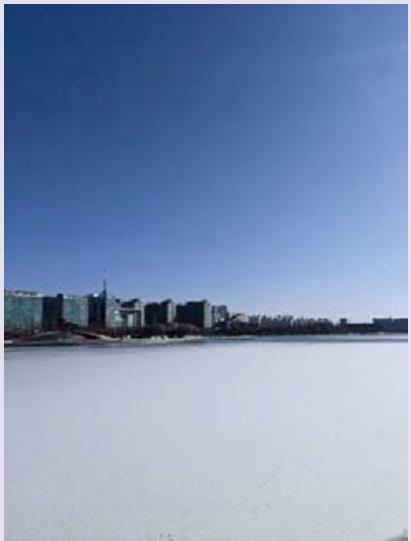
Pillow(필로우) 활용하기



■ 좌우 반전

- transpose(Image.FLIP_LEFT_RIGHT) 함수
 - 이미지의 왼쪽과 오른쪽을 반전시키고 반전 결과를 반환함

```
>>> img = Image.open("c:/photo/picture02.jpg")
>>> img.show()
>>> img = img.transpose(Image.FLIP_LEFT_RIGHT)
>>> img.show()
```



Pillow(필로우) 활용하기



■ 상하 반전

- transpose(Image.FLIP_TOP_BOTTOM) 함수
 - 이미지의 위쪽과 아래쪽을 반전시키고 반전 결과를 반환함

```
>>> img = Image.open("c:/photo/picture03.jpg")
>>> img.show()
>>> img = img.transpose(Image.FLIP_TOP_BOTTOM)
>>> img.show()
```



Pillow(필로우) 활용하기



■ 회전

- rotate(각도, expand=True) 함수
 - 각도만큼 이미지를 회전시키고 회전 결과를 반환함
 - expand=True : 회전할 때 화면이 확대되도록 함

```
>>> img = Image.open("c:/photo/picture05.jpg")
>>> img.show()
>>> img = img.rotate(45, expand=True)
>>> img.show()
```



Pillow(필로우) 활용하기



■ 이미지 잘라내기

- `crop((x1, y1, x2, y2))` 함수
 - (x1, y1) 좌표에서 (x2, y2) 좌표까지 이미지를 잘라내고 잘라낸 결과를 반환함

```
>>> img = Image.open("c:/photo/picture52.jpg")
>>> img.show()
>>> img = img.crop((100,100,600,600))
>>> img.show()
```



Pillow(필로우) 활용하기



■ 밝게 하기

- `ImageEnhance.Brightness(img).enhance(3.0)` 함수
 - 이미지(img)를 3.0만큼 밝게 하고, 밝게 한 결과를 반환함
 - 밝게 할 값은 1.0~5.0 사이의 값을 지정함

```
>>> img = Image.open("c:/photo/picture06.jpg")
>>> img.show()
>>> img = ImageEnhance.Brightness(img).enhance(3.0)
>>> img.show()
```



Pillow(필로우) 활용하기



■ 어둡게 하기

- `ImageEnhance.Brightness(img).enhance(0.4)` 함수
 - 이미지(img)를 0.4만큼 어둡게 하고, 어둡게 한 결과를 반환함
 - 밝게 하기와 반대로 어둡게 할 값은 0.0~1.0 사이의 값을 지정함

```
>>> img = Image.open("c:/photo/picture07.jpg")
>>> img.show()
>>> img = ImageEnhance.Brightness(img).enhance(0.4)
>>> img.show()
```



Pillow(필로우) 활용하기



■ 흑백 사진으로 만들기

- `ImageOps.grayscale(img)` 함수
 - 이미지(img)를 흑백으로 전환하고, 흑백으로 전환된 결과를 반환함

```
>>> img = Image.open("c:/photo/picture55.jpg")
>>> img.show()
>>> img = ImageOps.grayscale(img)
>>> img.show()
```



Pillow(필로우) 활용하기



■ 엠보싱 효과 넣기

- filter(ImageFilter.EMBOSS) 함수
 - 이미지(img)에 엠보싱 효과를 넣고, 엠보싱 효과가 들어간 결과를 반환함

```
>>> img = Image.open("c:/photo/picture55.jpg")
>>> img.show()
>>> img = img.filter(ImageFilter.EMBOSS)
>>> img.show()
```



Pillow(필로우) 활용하기

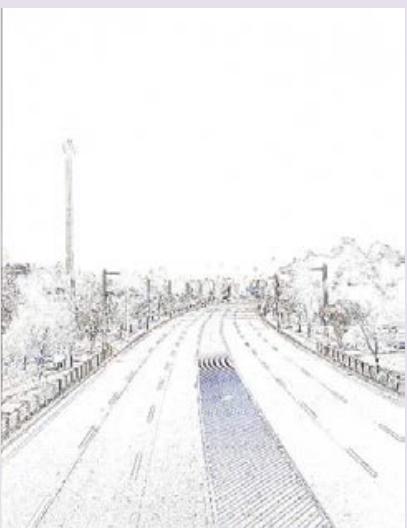
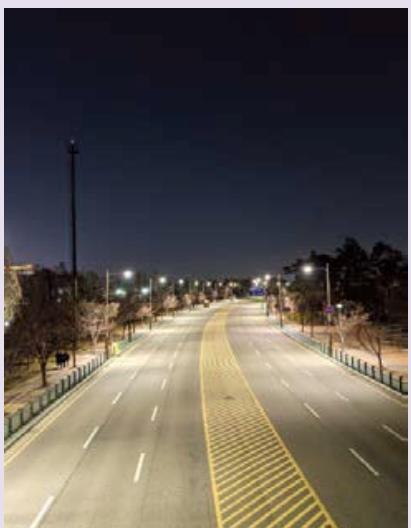


■ 연필 스케치 효과 넣기

■ filter(ImageFilter.CONTOUR) 함수

- 이미지(img)에 연필 스케치 효과를 넣고, 연필 스케치 효과가 들어간 결과를 반환함

```
>>> img = Image.open("c:/photo/picture73.jpg")
>>> img.show()
>>> img = img.filter(ImageFilter.CONTOUR)
>>> img.show()
```





■ 경계선 추출하기

- filter(ImageFilter.FIND_EDGES) 함수
 - 이미지(img)에서 경계선을 추출하고, 경계선이 추출된 결과를 반환함

```
>>> img = Image.open("c:/photo/picture24.jpg")
>>> img.show()
>>> img = img.filter(ImageFilter.FIND_EDGES)
>>> img.show()
```



Pillow(필로우) 활용하기



확인문제

각 영상 처리 코드의 효과를 다음 보기 중에서 고르시오.

[보기]

회전

엠보싱 효과

좌우 반전

흑백 만들기

일부 잘라내기

경계선 추출

밝게 또는 어둡게 하기

상하 반전

① `img.transpose(Image.FLIP_LEFT_RIGHT)` ---

② `img.transpose(Image.FLIP_TOP_BOTTOM)` ---

③ `img.rotate(각도, expand=True)` ---

④ `img.crop((x1, y1, x2, y2))` ---

⑤ `ImageEnhance.Brightness(img).enhance(숫자)` ---

⑥ `ImageOps.grayscale(img)` ---

⑦ `img.filter(ImageFilter.EMBOSS)` ---

⑧ `img.filter(ImageFilter.FIND_EDGES)` ---

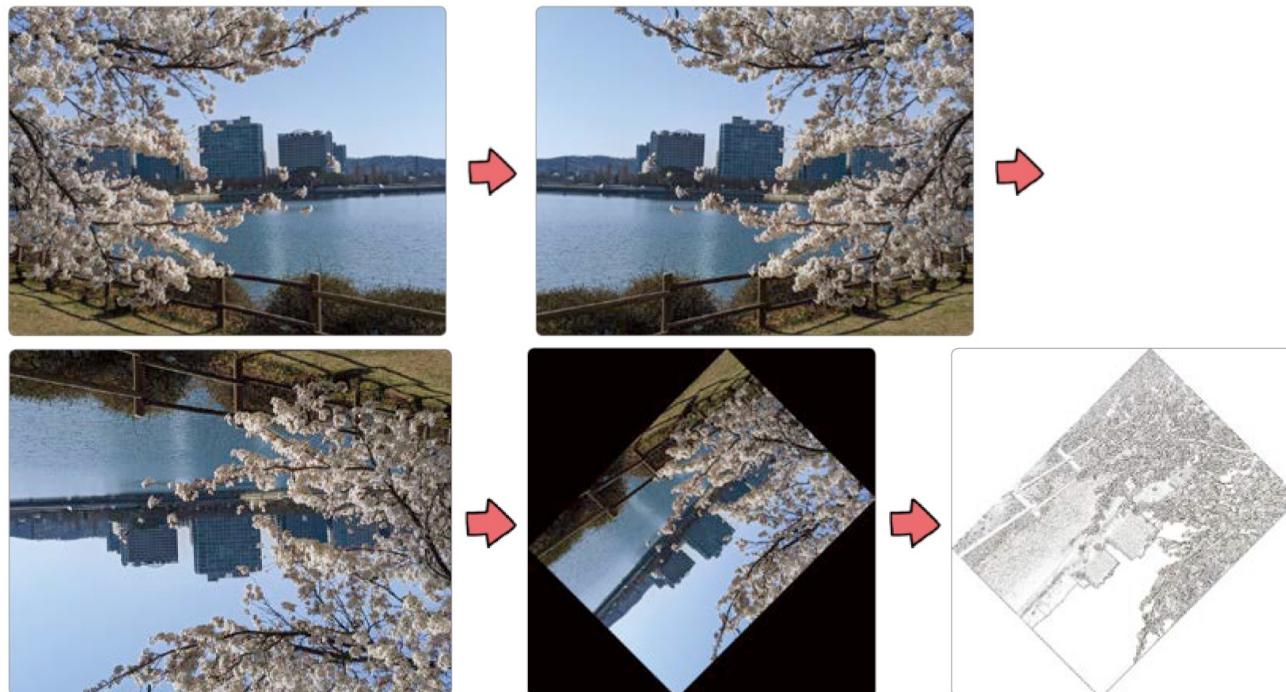
정답

Click!

사진에 효과 누적하기

99개의 사진 중 1개를 랜덤하게 뽑은 후에 좌우 반전, 상하 반전, 45도 회전, 연필 스케치 효과를 누적해서 적용해 봅시다.

효과가 누적된 최종 이미지는 뽑힌 사진의 번호를 output 뒤에 붙여 output00.jpg 형태로 저장합니다.



사진에 효과 누적하기

1. lab11-01.py 파일을 만들고, Pillow를 사용하기 위해 임포트하기

- 99개의 사진 중 1개를 뽑기 위해 random도 임포트하기

```
from PIL import Image, ImageFilter, ImageEnhance, ImageOps  
import random
```

2. 랜덤하게 파일을 추출하기

- 우선 0~99사이의 숫자를 뽑고, 10 미만의 한 자리 숫자가 뽑힐 경우에는 앞에 '0'을 붙여서 두 자리의 문자로 만들기
- 그리고 만들어진 글자와 파일명을 합쳐서 랜덤한 파일 이름을 추출하기

```
# 랜덤한 파일을 추출  
number = random.randint(1, 99)  
if number < 10 :  
    number = '0' + str(number)  
else :  
    number = str(number)  
  
filename = "C:/photo/picture" + number + ".jpg"
```

3. 파일을 열고 화면에 출력하고 앞에서 배운 Pillow 효과를 좌우 반전, 상하 반전, 45도 회전, 연필 스케치 효과 순서대로 적용한 후 화면에 출력하기

```
img = Image.open(filename)
img.show()
img = img.transpose(Image.FLIP_LEFT_RIGHT)
img.show()
img = img.transpose(Image.FLIP_TOP_BOTTOM)
img.show()
img = img.rotate(45, expand=True)
img.show()
img = img.filter(ImageFilter.CONTOUR)
img.show()
```

4. img 변수가 계속 이어지기 때문에 이미지 효과가 계속 누적됨

- 파일 이름을 저장하기 위해 랜덤으로 뽑힌 숫자(number)를 붙여 output00.jpg 형태로 파일을 저장하기

```
filename = "C:/photo/output" + number + ".jpg"  
img.save(filename)
```

5. <Ctrl>+<S>를 눌러서 변경된 내용을 저장하고, <F5>를 눌러 실행 결과 확인하기

Section 02

Pygame 라이브러리

Pygame 라이브러리 설치하기



■ Pygame(파이게임) 라이브러리

- 게임을 개발하기 위해서 사용하는 외부 라이브러리
- Pygame은 간단한 게임을 구현하는 데 편리하고 다양한 기능을 제공함
 - 특히 비행기 슈팅 게임 같은 것을 개발하기에 적합함

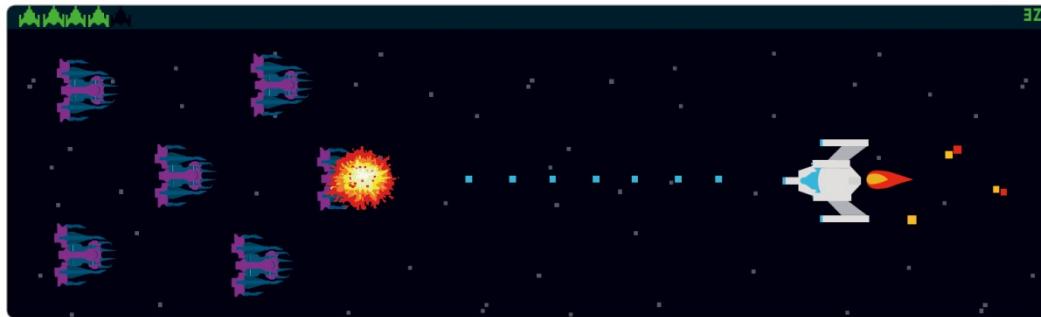


그림 11-8 비행기 슈팅 게임

Pygame 라이브러리 설치하기



■ Pygame 설치

- '윈도우 명령 프롬프트'에서 다음 명령을 입력하고 <Enter> 누르기

```
pip install pygame
```

A screenshot of a Windows Command Prompt window titled 'cmd C:\Windows\system32\cmd.exe'. The command 'pip install pygame' is entered at the prompt. The output shows the package being downloaded and installed. A red arrow points to the text 'Successfully installed pygame==2.0.1'. Another red arrow points to a warning message at the bottom about using pip version 20.2.3 while version 21.0.1 is available.

```
C:\Users\WinUser> pip install pygame
Collecting pygame
  Downloading pygame-2.0.1-cp39-cp39-win_amd64.whl (5.2 MB)
    100% |████████████████████████████████| 5.2 MB 2.2 MB/s
Installing collected packages: pygame
Successfully installed pygame==2.0.1
WARNING: You are using pip version 20.2.3; however, version 21.0.1 is available.
You should consider upgrading via the 'c:\Users\WinUser\AppData\Local\Programs\Python\Python39\python.exe -m pip install --upgrade pip' command.

C:\Users\WinUser>
```

그림 11-9 pygame 라이브러리 설치

Pygame 라이브러리 설치하기



설치 확인

- 파이썬 IDLE을 열고, import pygame을 입력하기
 - 아무런 메시지도 표시되지 않거나, 파란색 'Hello~~' 글자가 보이면 제대로 설치된 것임
 - 설치에 문제가 있다면 오류 메시지가 표시됨

```
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
>>> import pygame
pygame 2.0.1 (SDL 2.0.14, Python 3.9.2)
Hello from the pygame community. https://www.pygame.org/contribute.html
>>> |
```

그림 11-10 Pygame 라이브러리 설치 확인

[거북이의 바다] 게임 만들기



■ [거북이의 바다] 게임

- 사용자가 키보드의 화살표를 이용해서 움직이는 대로 거북이 한 마리가 바다(화면)를 돌아다니는 간단한 게임



그림 11-11 [거북이의 바다] 게임 완성 화면

이번에 작성하는 [거북이의 바다]와 거북이 그래픽은 전혀 상관 없는 코드입니다.

[거북이의 바다] 게임 만들기



■ 기본 화면 구성 익히기

- 배경 화면의 색상은 프로그램을 실행할 때마다 랜덤하게 변함
- [코드 11-1]을 실행하면 기본 화면이 나오고, IDLE Shell에 #이 무한 반복됨
 - 프로그램을 종료하려면 IDLE Shell 창을 닫아야 함

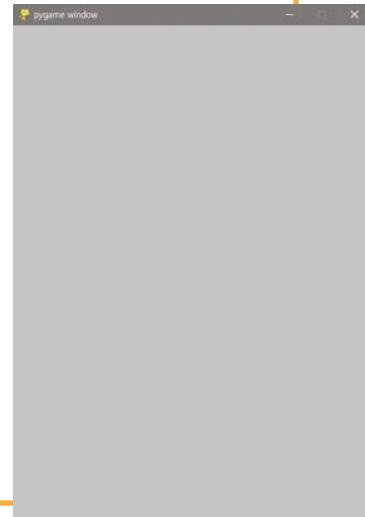
[코드 11-1]

```
import pygame
import random
import sys

### 전역변수부
monitor = None
colorList = ["red", "green", "blue", "black", "magenta",
            "orange", "gray"]

### 메인 코드부
pygame.init()
monitor = pygame.display.set_mode((500, 700))
color = random.choice(colorList)

while True :
    monitor.fill(color)
    pygame.display.update()
    print('#', end='')
```



[거북이의 바다] 게임 만들기



■ 화면에 거북이 나타내기

- [거북이의 바다]의 메인 캐릭터인 거북이의 크기는 100×100 내외가 적절함



그림 11-12 [거북이의 바다] 게임의 캐릭터(turtle.png 파일)
그림 파일은 아무거나 상관없습니다. 자신의 사진을 100×100 정도로 작게 저장한 그림파일도 괜찮습니다.

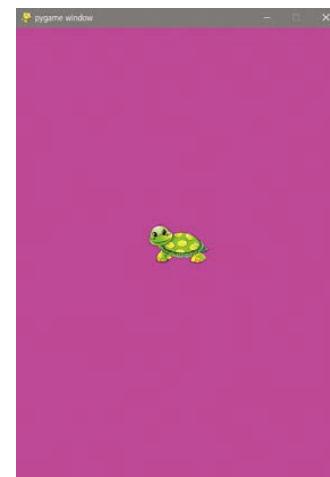
[코드 11-2]

...생략([코드 11-1]의 1~7행)...

메인 코드부

```
pygame.init()
monitor = pygame.display.set_mode((500, 700))
color = random.choice(colorList)
turtle = pygame.image.load('turtle.png')

while True :
    monitor.fill(color)
    monitor.blit(turtle, (200,300))
    pygame.display.update()
```



[거북이의 바다] 게임 만들기



확인문제

다음 빈칸에 들어갈 단어를 채우시오.

이미지를 Pygame에 로딩하기 위해서는 pygame.image. (“그림파일”) 함수를 사용해야 한다. 그리고 모니터에 출력하려면 monitor. (그림, (X위치, Y위치))를 사용한다.

정답

Click!

[거북이의 바다] 게임 만들기



■ 마우스 이벤트 처리하기

- 이벤트(Event)란?
 - 윈도우 화면에서 일어나는 작은 사건들을 의미함
 - 마우스 클릭하기, 이동하기 등의 마우스 작동과 키보드 누르기 등
- Pygame 화면에 마우스나 키보드의 이벤트를 잡아내는 코드의 형태

```
for e in pygame.event.get() :  
    if e.type in [이벤트 종류] :  
        # 작동할 코드
```

[거북이의 바다] 게임 만들기



■ [거북이의 바다] 게임 종료시키기

- 이제까지는 IDLE Shell 창을 닫아야 Pygame이 종료되었음
- 이번에는 오른쪽 위 버튼을 누르면 게임이 종료되도록 수정하기

[코드 11-3]

...생략([코드 11-1]의 1~7행)...

```
### 메인 코드부
pygame.init()
monitor = pygame.display.set_mode((500, 700))
color = random.choice(colorList)
turtle = pygame.image.load('turtle.png')

while True :
    monitor.fill(color)
    monitor.blit(turtle, (200,300))
    pygame.display.update()

    for e in pygame.event.get() :
        if e.type in [pygame.QUIT] :
            pygame.quit()
            sys.exit()
```



그림 11-13 프로그램 종료 방법 구현

[거북이의 바다] 게임 만들기



■ 키보드 이벤트 처리하기

- 키보드의 화살표 키를 누르면, 거북이가 이동함
- for문 안에서 키보드를 눌렀는지 체크하고, 키보드 중에서 화살표키 4개를 눌렀는지 구분하여 이동시킴
 - 외부 if문 : [pygame.KEYDOWN]을 사용하여 키보드를 누른 작동은 모두 참이 되어 내부 if문이 실행됨
 - 내부 if문 : 누른 키가 어떤 것이냐에 따라 거북이의 좌표를 증가 또는 감소함

```
for e in pygame.event.get() :  
    if e.type in [키보드 이벤트 종류] :  
        if e.key == 왼쪽 화살표 :  
            # 거북이의 x좌표를 감소(=왼쪽 이동)  
        if e.key == 오른쪽 화살표 :  
            # 거북이의 x좌표를 증가(=오른쪽 이동)  
        if e.key == 위쪽 화살표 :  
            # 거북이의 y좌표를 감소(=위쪽 이동)  
        if e.key == 아래쪽 화살표 :  
            # 거북이의 y좌표를 증가(=아래쪽 이동)
```

[거북이의 바다] 게임 만들기



■ 키보드 이벤트 처리하기

[코드 11-4]

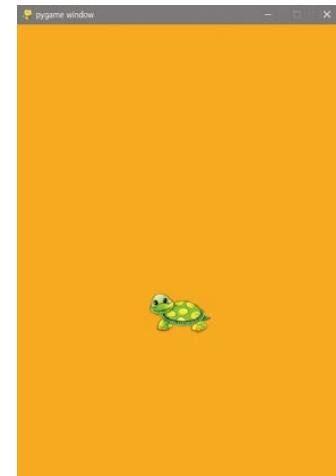
...생략([코드 11-1]의 1~7행)...

```
### 메인 코드부
pygame.init()
monitor = pygame.display.set_mode((500, 700))
color = random.choice(colorList)
turtle = pygame.image.load('turtle.png')
tx, ty = 200, 300

while True :
    monitor.fill(color)
    monitor.blit(turtle, (tx,ty))
    pygame.display.update()

    for e in pygame.event.get() :
        if e.type in [pygame.QUIT] :
            pygame.quit()
            sys.exit()

        if e.type in [pygame.KEYDOWN] :
            if e.key == pygame.K_LEFT : tx -= 10
            elif e.key == pygame.K_RIGHT : tx += 10
            elif e.key == pygame.K_UP : ty -= 10
            elif e.key == pygame.K_DOWN : ty += 10
```



[거북이의 바다] 게임 만들기



확인문제

다음 중 잘못된 것을 고르시오.

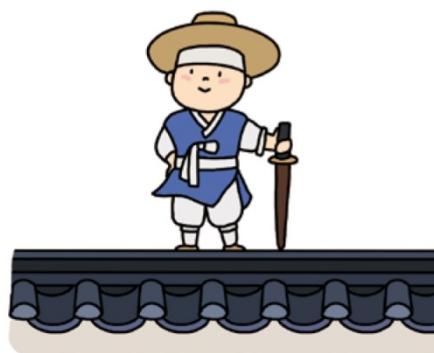
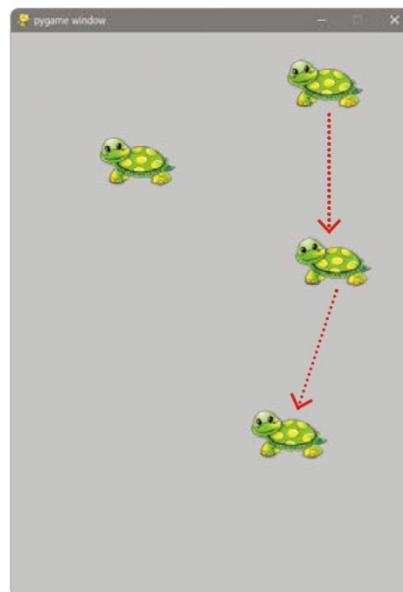
- ① Pygame에서 마우스, 키보드 등의 작은 사건을 이벤트라 부른다.
- ② 키보드를 누른 이벤트를 잡기 위해 pygame.KEYDOWN을 사용한다.
- ③ 모든 키보드를 누른 이벤트가 pygame.KEYDOWN에 속하는 것은 아니다.
- ④ Pygame 화면의 **X**를 누르는 이벤트 종류는 [pygame.QUIT]이다.

정답

Click!

동에 번쩍 서에 번쩍 홍길동 거북이

'홍길동'이라 함은 동에 번쩍 서에 번쩍 하는 사람을 일컫는 대명사로 사용하기도 합니다. 이번에는 Spacebar를 누를 때마다, Pygame의 거북이가 홍길동처럼 이곳 저곳을 번쩍번쩍 랜덤하게 이동하는 코드를 작성해 봅시다.



1. lab11-02.py 파일을 만들고, [거북이의 바다] 게임과 동일한 기본 화면을 구성하기

```
import pygame
import random
import sys

### 전역 변수부
monitor = None
colorList = ["red", "green", "blue", "black", "magenta",
"orange", "gray"]

### 메인 코드부
pygame.init()
monitor = pygame.display.set_mode((500, 700))
color = random.choice(colorList)
turtle = pygame.image.load('turtle.png')
```

동에 번쩍 서에 번쩍 홍길동 거북이

2. 누른 키가 <Spacebar>라면 랜덤하게 tx와 ty를 추출하고, 거북이의 위치를 변경시킴

```
tx, ty = 200, 300
while True :
    monitor.fill(color)
    monitor.blit(turtle, (tx,ty))
    pygame.display.update()

    for e in pygame.event.get() :
        if e.type in [pygame.QUIT] :
            pygame.quit()
            sys.exit()

        if e.type in [pygame.KEYDOWN] :
            if e.key == pygame.K_SPACE :
                tx = random.randint(0, 500)
                ty = random.randint(0, 700)
```

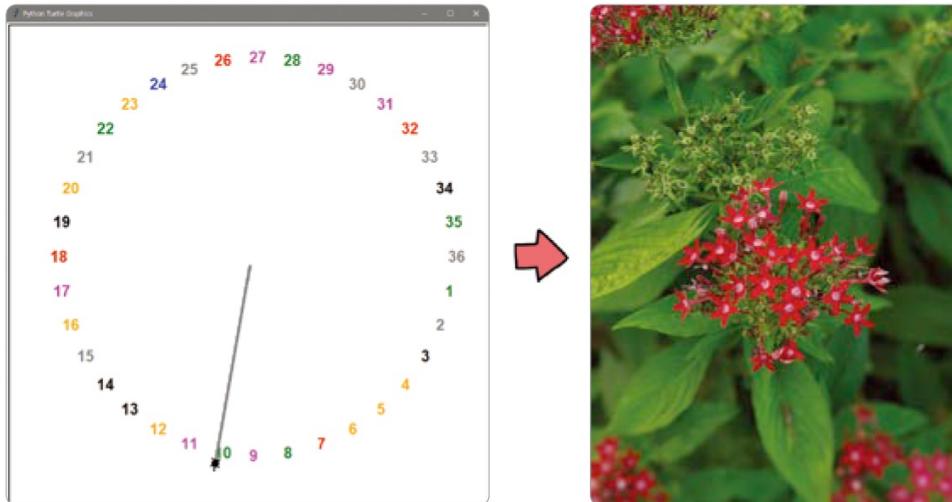
3. <Ctrl>+<S>를 눌러서 변경된 내용을 저장하고, <F5>를 눌러 실행 결과 확인하기

[실전 예제] 룰렛하는 거북이

실전 예제 룰렛하는 거북이

[문제]

- 거북이 그래픽에서 360도의 룰렛판을 10도 단위로 그리고, 룰렛 중에서 하나를 골라서 해당 번호에 해당하는 사진을 화면에 출력하기
- 거북이 그래픽과 Pillow를 혼합한 응용 프로그램



실전 예제 룰렛하는 거북이

[해결]

```
import turtle
import random
import time
from PIL import Image, ImageFilter, ImageEnhance, ImageOps

colorList = ["red", "green", "blue", "black", "magenta", "orange", "gray"]
turtle.shape('turtle')
turtle.setup(850, 850)
turtle.screensize(800, 800)
turtle.penup()

num = 1
for _ in range(36) :
    turtle.goto(0, 0)
    turtle.right(10)
    turtle.forward(350)
    turtle.pencolor(random.choice(colorList))
    turtle.write(str(num), font=('맑은고딕', 20, 'bold'))
    num += 1

    turtle.goto(0,0)
    time.sleep(5)
    turtle.pendown()
    turtle.pensize(5)
```

실전 예제 룰렛하는 거북이

[해결]

```
angle = random.randint(10,360) // 10 * 10
turtle.right(angle)
turtle.forward(350)

number = angle // 10
if number < 10 :
    number = '0' + str(number)
else :
    number = str(number)

filename = "C:/photo/picture" + number + ".jpg"
img = Image.open(filename)
img.show( )

turtle.done( )
```

Preview

감사합니다 :)