

Python 5강

5강 - PYGAME 시작

스케줄

5월 4일 - Sprite, 충돌 이벤트 등

과제 : 갤로그

5월 11일(자습) - (cont.) 갤로그 만들기

과제: 게임 구상 (5월 18일 전까지 나한테 컨펌 받기)

5월 18일 - 구상한 게임 만들기 시작

과제: 5월 25일까지 게임 완성하기

5월 25일 - 축제

과제: 게임구상 (6월1일 전까지 나한테 컨펌 받기)

6월 1일 - 구상한 게임 만들기 시작

파이게임 시작 - 스크린 만들기

```
>>> import pygame  
>>> screen = pygame.display.set_mode((1024,768))  
>>> screen = pygame.display.set_mode((1024,768), FULLSCREEN)
```

`set_mode` 함수를 불러와서 게임 창 크기를 정할 수 있고, `FULLSCREEN`을 통해 전체화면 모드로도 불러올 수 있다.

* 아직 게임 종료 기능을 구현하지 않았으니 파일을 실행할 수 없다. 실행하면 게임이 응답없음 상태에서 빠져 나오지 못한다.

파이게임 시작 - 자동차 그리기

```
>>> car = pygame.image.load('car.png')  
>>> screen.blit(car, (50,100))  
>>> pygame.display.flip()
```

지금 작성하는 파일의 디렉토리와 같은 디렉토리에 **car.png** 를 불러온다. 그리고, **BLIT(Block Image Transfer)**라는 함수를 통해, 이미지를 **screen**의 **X,Y** 좌표 위치에 복사한다.

마지막으로, **flip()** 함수로 화면에 띄워준다.

파이게임 시작 - 자동차 회전하기

이렇게 하고 나면, 자동차가 화면의 50,100 위치에 나타날 것이다.

```
>>> import math
```

```
>>> rotated = pygame.transform.rotate(car, 45 * math.pi / 180)
```

```
>>> screen.blit(car, (50,100))
```

```
>>> pygame.display.flip()
```

이렇게 이미지를 회전할 수도 있다.

파이게임 시작 - 자동차 움직이기

```
>>> for i in range(100):
```

```
...     screen.fill((0,0,0))    # 바탕 검은색으로 칠하기
```

```
...     screen.blit(car, (i,0))
```

화면에서 이미지를 움직일 때는, 바탕을 칠해서, 화면을 지우고, 새로운 위치에 다시 그리고, 칠하고를 반복하는 작업을 통해 이미지가 움직이는 것처럼 보이게 한다. 나중에 설명하겠지만, 이 작업은 **Sprite** 를 통해 작업하는 것이 더 좋다.

파이게임 시작 - 입력 제어

파이게임에서 유저의 입력에는, 마우스 클릭, 키보드, 조이스틱 등이 있다. 이들을 제어하는 함수들에 대해 알아보겠다.

```
>>> pygame.event.wait()
```

```
>>> pygame.event.poll()
```

```
>>> pygame.event.get()
```

wait 은 어떤 입력 이벤트가 발생할 때 까지 모든 작동을 멈춘다. 잘 쓰이지 않는다. **poll**은 어떤 이벤트가 실행되기를 기다리는지를 반환한다. 만약 아무 이벤트도 없다면 **NOEVENT** 가 나온다. **get**은 **poll**과 비슷하지만, 실행되고 있는 모든 이벤트를 알려준다. 이는 키보드 입력이나, 마우스 움직임 등의 이벤트를 감지한다.

파이게임 시작 - 프레임 타임 설정

프레임 타임은 게임에 있어서 특히, 유저에게 중요하다. 타임 컨트롤이 되지 못한다면, 게임이 실행되는 환경에 따라서, 속도가 너무 빨라질수도 느려질수도 있기 때문이다.

```
>>> clock = pygame.time.Clock()
```

```
>>> FRAMES_PER_SECOND = 30
```

```
>>> deltat = clock.tick(FRAMES_PER_SECOND)
```

`tick` 함수는 시계 객체가 1/30 초마다 멈추도록 설정한다. 즉, 화면 그리는 속도를 초당 30번으로 제한하는 것이다. 단, 느린 컴퓨터에선 초당 30프레임이 안될 수도 있다고 한다.

초당 프레임은 유저 입력을 얼마나 자주 허용하는지를 설정 하는 것이라고 생각하면 된다.

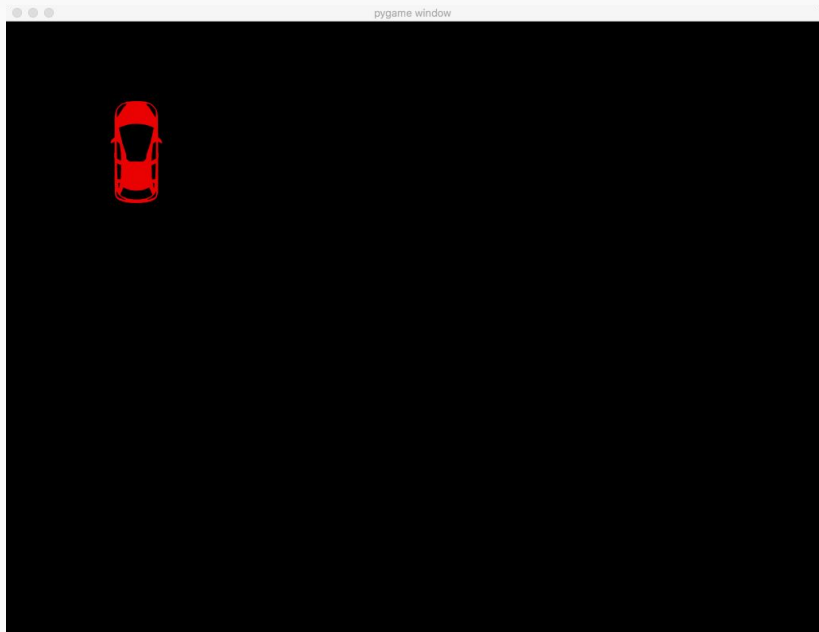
파이게임 시작 - 예제

```
import pygame
from pygame.locals import *
pygame.init()
screen = pygame.display.set_mode((1024,768))
car = pygame.image.load('car.png')
```

```
while 1:
    screen.fill(0)
    screen.blit(car,(100,100))
    pygame.display.flip()
```

```
for event in pygame.event.get():
    if event.type==pygame.QUIT:
        pygame.quit()
        exit(0)
```

파이게임 시작하기 - 예제



실행하면 다음과 같은 결과물이 나올 것이다.

과제

1. Rapid Game Development In Python.pdf - 3쪽 까지 읽고 마지막 코드 실행해보기
2. <http://toyfab.tistory.com/entry/10대들을-위한-파이선-게임-프로그램-시작하기> 코드를 쳐보면서 설명을 읽고 게임 만들어보기. (* 이미지들이 불러올 수 없습니다로 돼있는데, 이 이미지들은 글 위쪽에 원문보기 링크가 있으니, 거기서 보면 이미지를 볼 수 있어요)