Lecture #12. 게임 월드

2D 게임 프로그래밍

이대현 교수

한국공학대학교
TECH UNIVERSITY OF KOREA

학습 내용

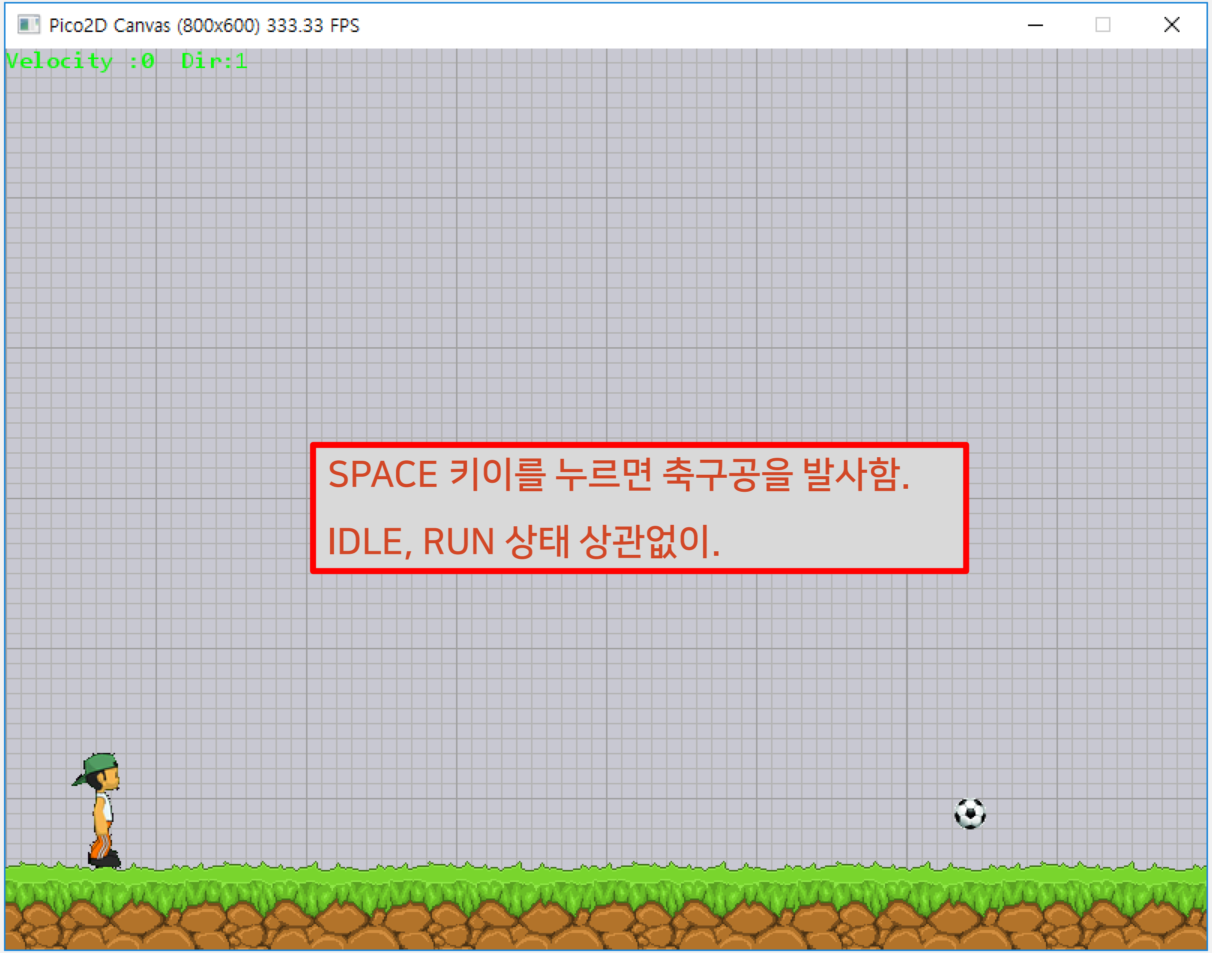
- 캐릭터 상태의 추가

- 특수 이벤트 처리

- 게임 월드 구성

실습

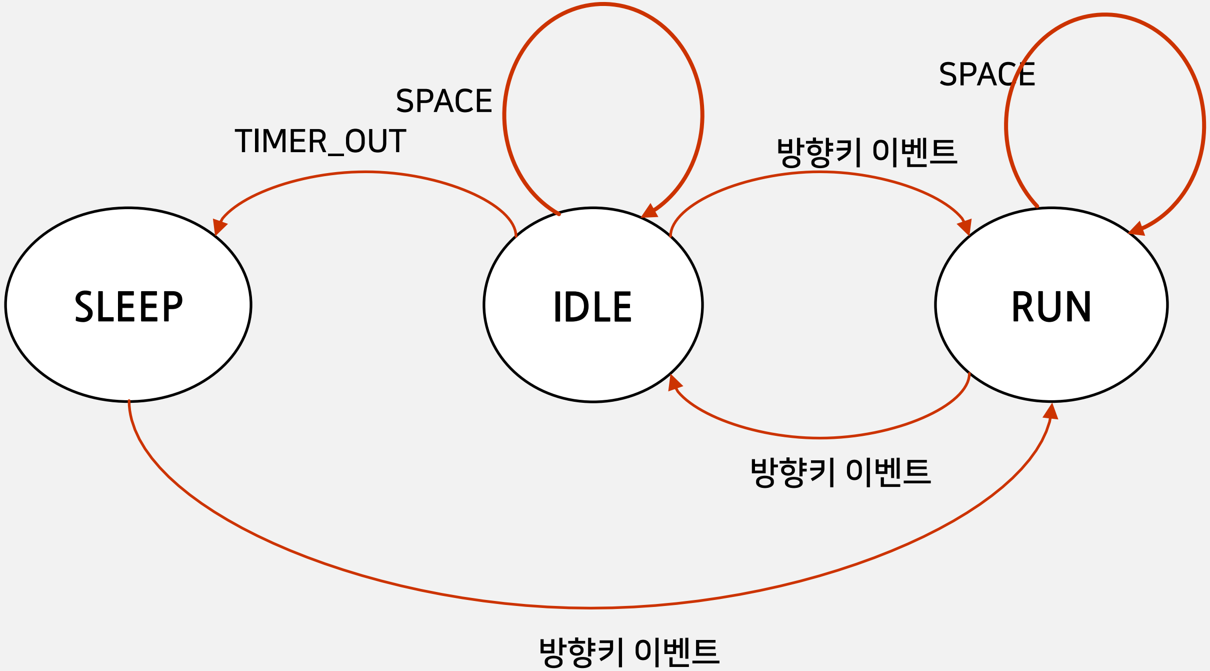
소년의 축구공 발사



SPACE 키이를 누르면 축구공을 발사함.

IDLE, RUN 상태 상관없이.

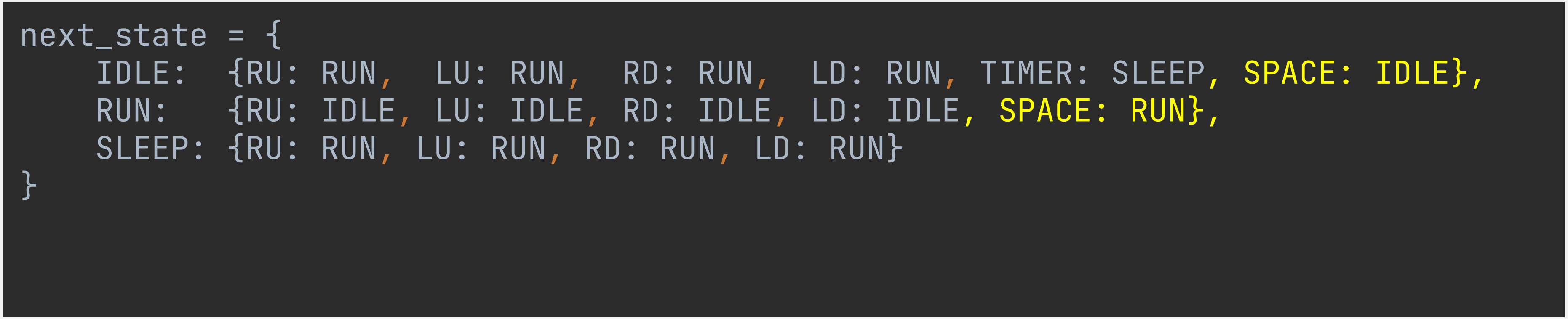
상태 다이어그램



boy.py – SPACE 이벤트 추가

```
RD, LD, RU, LU, TIMER, SPACE = range(6)

key_event_table = {
    (SDL_KEYDOWN, SDLK_SPACE): SPACE,
    (SDL_KEYDOWN, SDLK_RIGHT): RD,
    (SDL_KEYDOWN, SDLK_LEFT): LD,
    (SDL_KEYUP, SDLK_RIGHT): RU,
    (SDL_KEYUP, SDLK_LEFT): LU
}
```

boy.py – 상태 변화 추가



```
next_state = {
    IDLE:  {RU: RUN,  LU: RUN,  RD: RUN,  LD: RUN, TIMER: SLEEP, SPACE: IDLE},
    RUN:   {RU: IDLE, LU: IDLE, RD: IDLE, LD: IDLE, SPACE: RUN},
    SLEEP: {RU: RUN, LU: RUN, RD: RUN, LD: RUN}
}
```

boy.py – boy 의 fire_ball 함수 추가

```python
def fire_ball(self):
    print('FIRE BALL')
```

boy.py – RunState, IdleState의 exit() 함수 조정

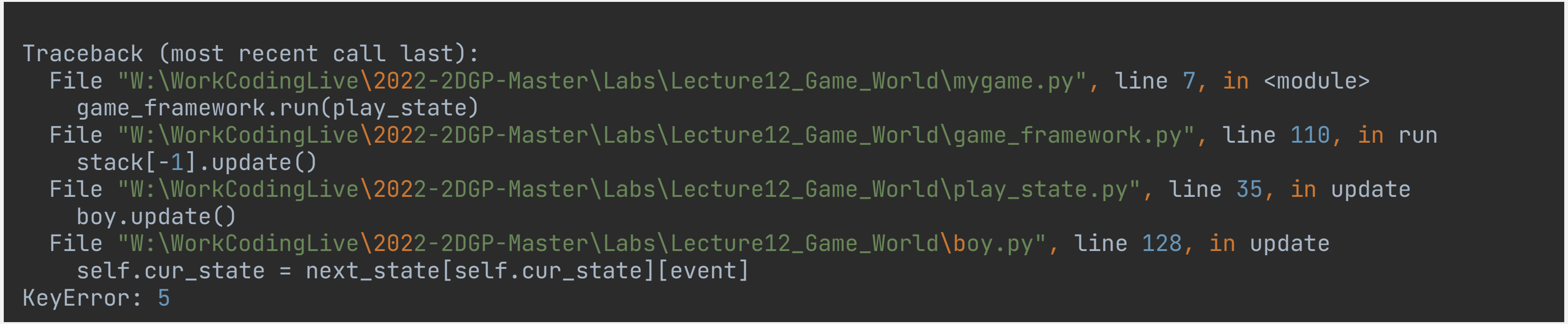
실습

```
class IDLE:
    @staticmethod
    def enter(self,event):
        print('ENTER IDLE')
        self.dir = 0
        self.timer = 1000

    @staticmethod
    def exit(self, event):
        print('EXIT IDLE')
        if event == SPACE:
            self.fire_ball()
```

```
class RUN:
    def enter(self, event):
        print('ENTER RUN')
        if event == RD:
            self.dir += 1
        elif event == LD:
            self.dir -= 1
        elif event == RU:
            self.dir -= 1
        elif event == LU:
            self.dir += 1

    def exit(self, event):
        print('EXIT RUN')
        self.face_dir = self.dir
        if event == SPACE:
            self.fire_ball()
```

실행하고 SPACE 를 눌러보자?

SLEEP 상태에서 SPACE 를 누르면?



```
Traceback (most recent call last):
  File "W:\WorkCodingLive\2022-2DGP-Master\Labs\Lecture12_Game_World\mygame.py", line 7, in <module>
    game_framework.run(play_state)
  File "W:\WorkCodingLive\2022-2DGP-Master\Labs\Lecture12_Game_World\game_framework.py", line 110, in run
    stack[-1].update()
  File "W:\WorkCodingLive\2022-2DGP-Master\Labs\Lecture12_Game_World\play_state.py", line 35, in update
    boy.update()
  File "W:\WorkCodingLive\2022-2DGP-Master\Labs\Lecture12_Game_World\boy.py", line 128, in update
    self.cur_state = next_state[self.cur_state][event]
KeyError: 5
```

상태 변환 디버그

```
RD, LD, RU, LU, TIMER, SPACE = range(6)
event_name = ['RD', 'LD', 'RU', 'LU', 'TIMER', 'SPACE']
```

```
def update(self):
    self.cur_state.do(self)

    if self.event_que:
        event = self.event_que.pop()
        self.cur_state.exit(self, event)
        try:
            self.cur_state = next_state[self.cur_state][event]
        except KeyError:
            print(f'ERROR: State {self.cur_state.__name__}    Event {event_name[event]}')
        self.cur_state.enter(self, event)
```

boy.py – 상태 변화 추가

```
next_state = {
    IDLE:  {RU: RUN,  LU: RUN,  RD: RUN,  LD: RUN, TIMER: SLEEP, SPACE: IDLE},
    RUN:   {RU: IDLE, LU: IDLE, RD: IDLE, LD: IDLE, SPACE: RUN},
    SLEEP: {RU: RUN, LU: RUN, RD: RUN, LD: RUN, SPACE: IDLE}
}
```

ball.py

```
from pico2d import *
import game_world

class Ball:
    image = None

    def __init__(self, x = 800, y = 300, velocity = 1):
        if Ball.image == None:
            Ball.image = load_image('ball21x21.png')
        self.x, self.y, self.velocity = x, y, velocity

    def draw(self):
        self.image.draw(self.x, self.y)

    def update(self):
        self.x += self.velocity
```

게임 월드 game_world.py 작성

```python
# layer 0: Background Objects
# layer 1: Foreground Objects
objects = [[], []]



def add_object(o, depth):
    objects[depth].append(o)

def add_objects(ol, depth):
    objects[depth] += ol

def remove_object(o):
    for layer in objects:
        if o in layer:
            layer.remove(o)
            del o
            return
    raise ValueError('Trying destroy non existing object')
```

게임 월드에 담겨있는 모든 객체들을 담고 있는 리스트. Drawing Layer 에 따라서 분류. 필요에 따라 Layer를 추가하면 됨. 현재는 두개의 Layer만.

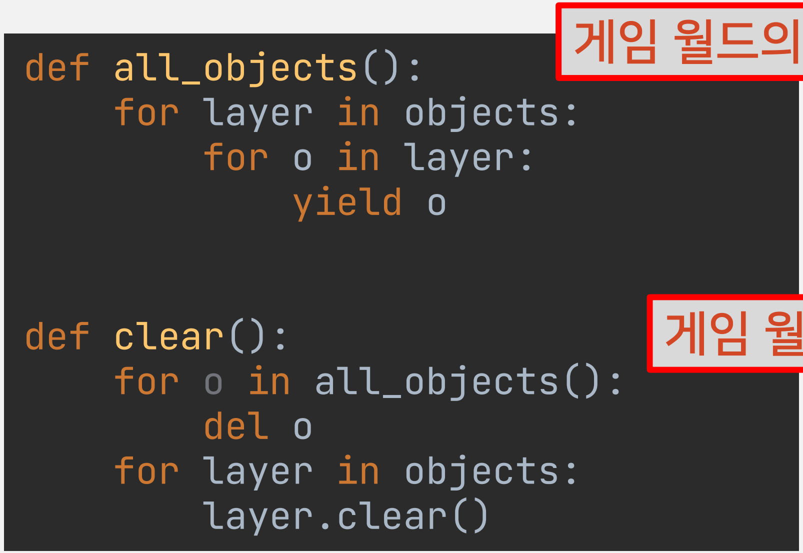
게임 월드에 객체 추가

게임 월드에 객체'들'을 추가

게임 월드에서 객체 제거

게임 월드 game_world.py

실습



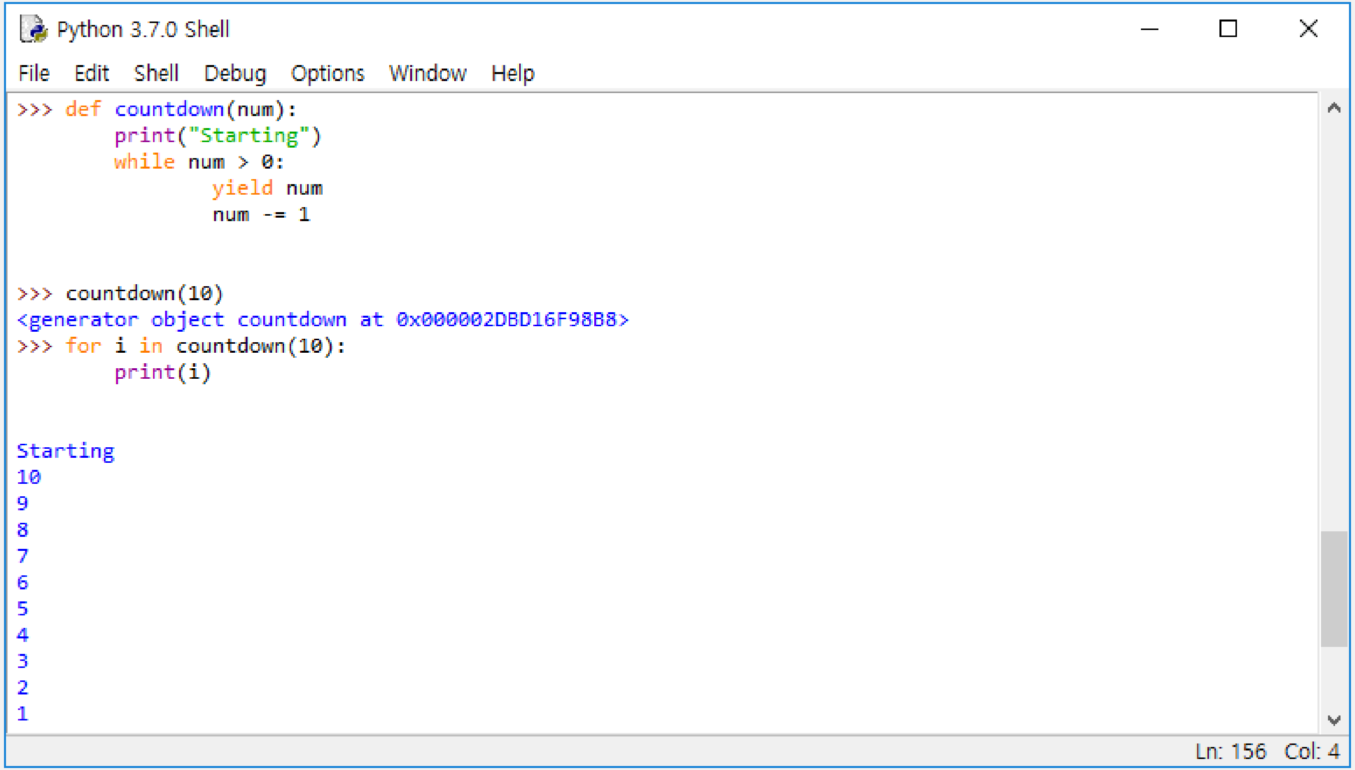
```python
def all_objects():
    for layer in objects:
        for o in layer:
            yield o


def clear():
    for o in all_objects():
        del o
    for layer in objects:
        layer.clear()
```

게임 월드의 모든 객체들을 하나씩 꺼내오기

게임 월드의 모든 객체 제거

Python Generator

- 객체들을 하나씩 만들어서(발전) 넘겨주는 기능
- for 문 등에서 효과적으로 사용.



```
>>> def countdown(num):
	print("Starting")
	while num > 0:
		yield num
		num -= 1


>>> countdown(10)
<generator object countdown at 0x000002DBD16F98B8>
>>> for i in countdown(10):
	print(i)


Starting
10
9
8
7
6
5
4
3
2
1
```

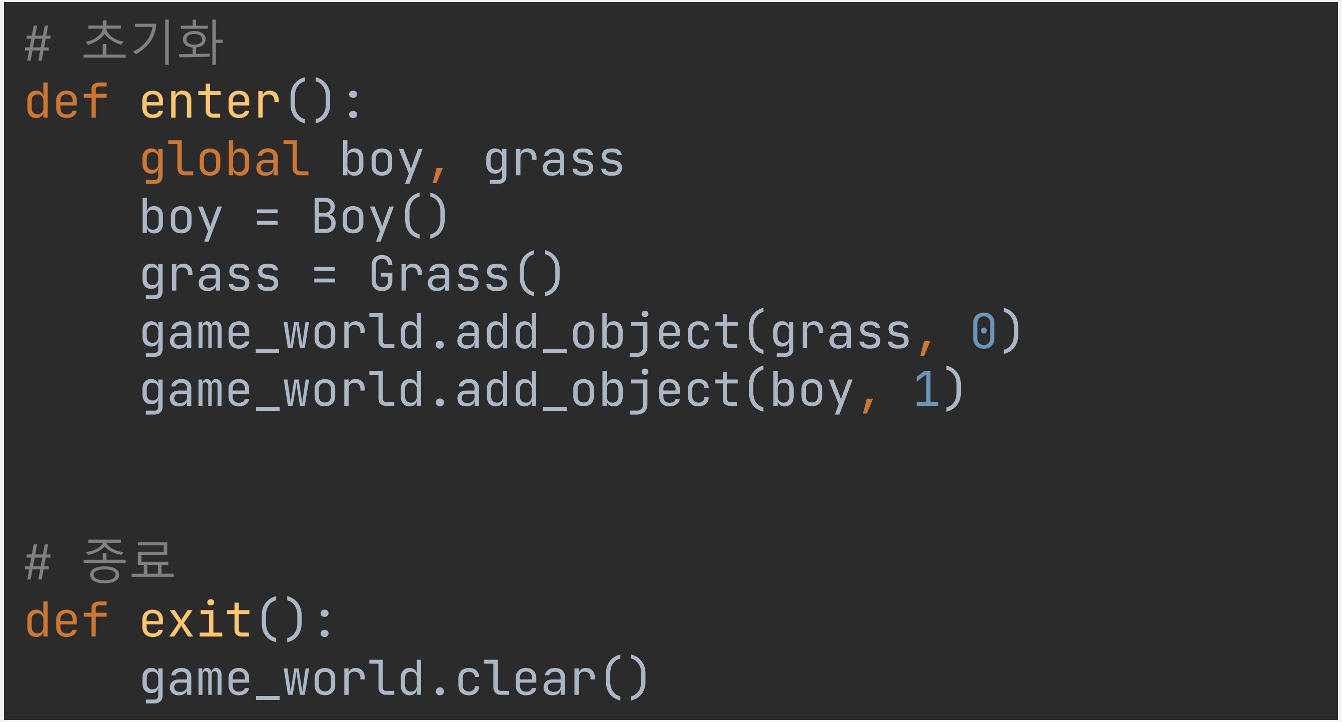
boy.py

```python
import game_world
```

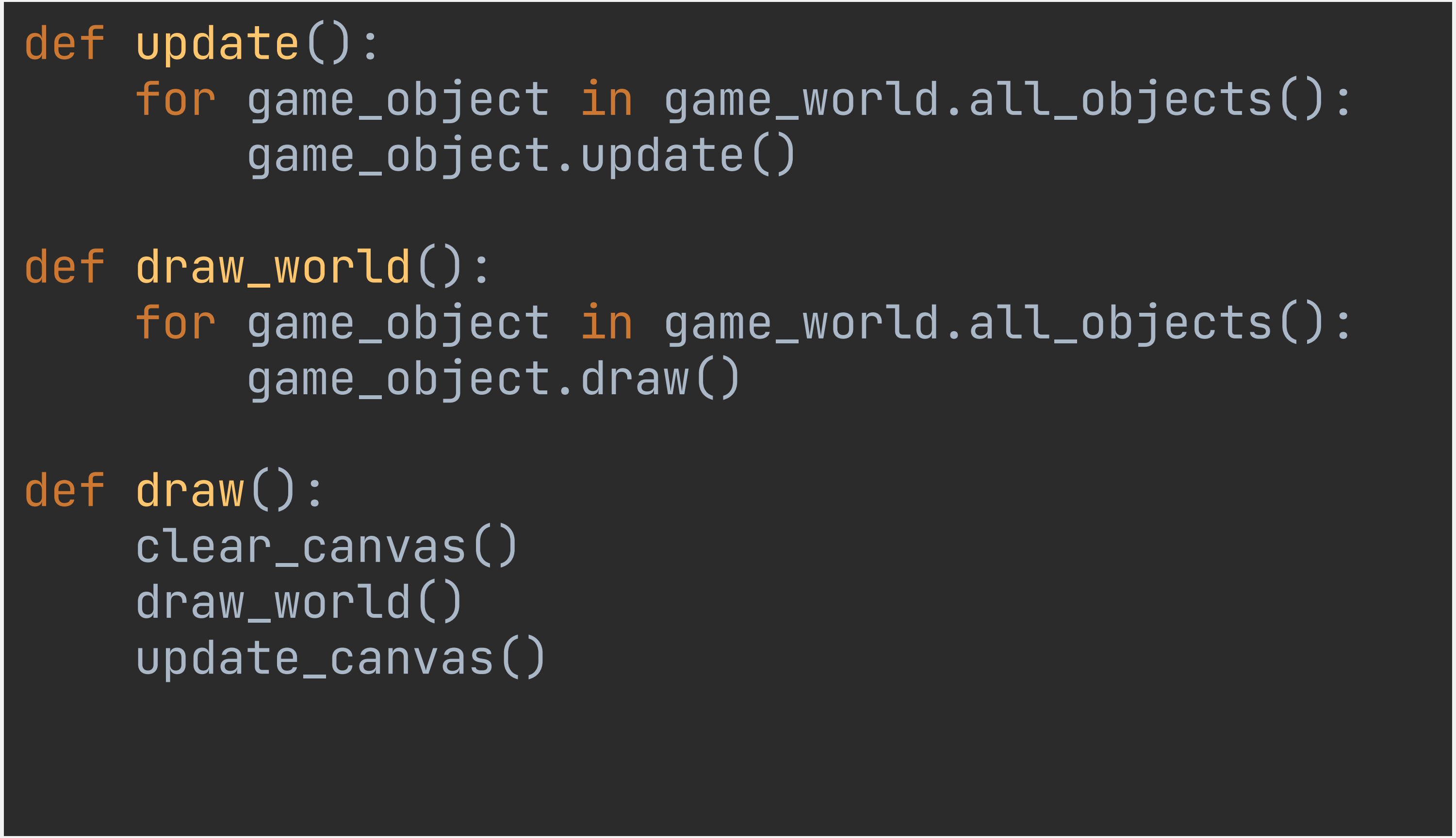
```python
def fire_ball(self):
    ball = Ball(self.x, self.y, self.face_dir*2)
    game_world.add_object(ball, 1)
```

play_state.py – 게임 월드를 이용하도록 조정

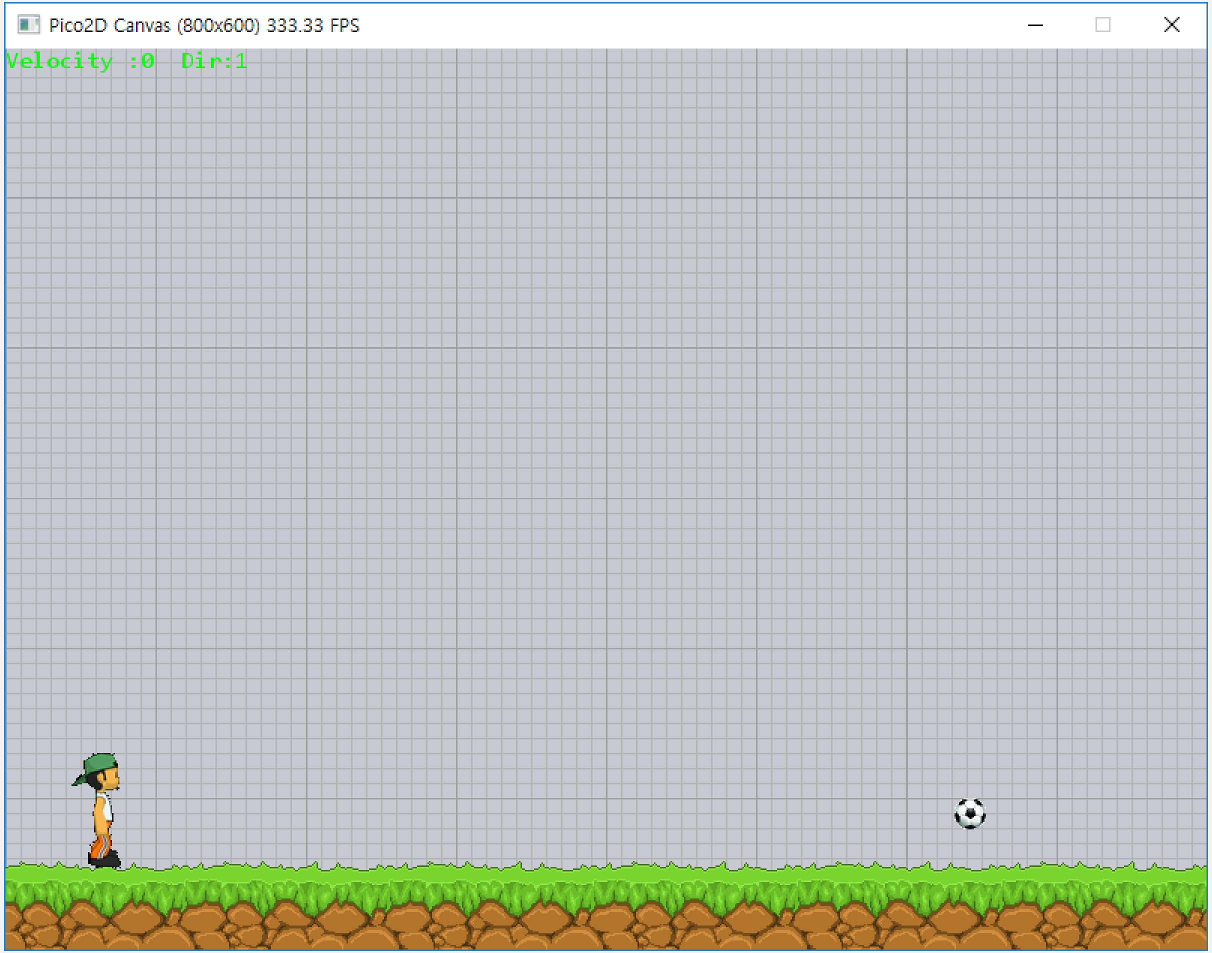
실습



```
# 초기화
def enter():
    global boy, grass
    boy = Boy()
    grass = Grass()
    game_world.add_object(grass, 0)
    game_world.add_object(boy, 1)


# 종료
def exit():
    game_world.clear()
```



```python
def update():
    for game_object in game_world.all_objects():
        game_object.update()

def draw_world():
    for game_object in game_world.all_objects():
        game_object.draw()

def draw():
    clear_canvas()
    draw_world()
    update_canvas()
```



```
Velocity :0  Dir:1
```

ball.py – ball의 제거

실습

```
from pico2d import *
import game_world

class Ball:
    image = None

    def __init__(self, x = 400, y = 300, velocity = 1):
        if Ball.image == None:
            Ball.image = load_image('ball21x21.png')
        self.x, self.y, self.velocity = x, y, velocity

    def draw(self):
        self.image.draw(self.x, self.y)

    def update(self):
        self.x += self.velocity

        if self.x < 25 or self.x > 800 - 25:
            game_world.remove_object(self)
```