```python
# 1 - 모듈 임포트
import pygame
import sys
import keyboard
import mouse
import math
import traceback

# 2 - 게임 변수, 딕셔너리 초기화
    # 2.1 - 게임 화면
pygame.init()
screen = pygame.display.set_mode((1024, 512))
    # 2.2 - 시간관련 변수
FPS = 30
fpsClock = pygame.time.Clock()
    # 2.3 - 화면 이미지 변수
lobbyimage = pygame.transform.scale(pygame.image.load("images\page_startingimage.png"), (1024,512))
infoimage = pygame.transform.scale(pygame.image.load('images\page_infoimage.png'), (1024,512))
endingimage = pygame.transform.scale(pygame.image.load('images\ending.jpg'), (1024,512))
None_None = pygame.transform.scale(pygame.image.load('images/None_None.png'),(1024,512))
Sungmin_None = pygame.transform.scale(pygame.image.load('images/Sungmin_None.png'), (1024,512))
Yungbum_None= pygame.transform.scale(pygame.image.load('images/Yungbum_None.png'), (1024,512))
Junhyuk_None = pygame.transform.scale(pygame.image.load('images/Junhyuk_None.png'), (1024,512))
Jinhyuk_None = pygame.transform.scale(pygame.image.load('images/Jinhyuk_None.png'), (1024,512))
None_B =  pygame.transform.scale(pygame.image.load('images/None_B.png'), (1024,512))
None_R =  pygame.transform.scale(pygame.image.load('images/None_R.png'), (1024,512))
Sungmin_B = pygame.transform.scale(pygame.image.load('images/Sungmin_B.png'),(1024,512))
Yungbum_B = pygame.transform.scale(pygame.image.load('images/Yungbum_B.png'),(1024,512))
Junhyuk_B = pygame.transform.scale(pygame.image.load('images/Junhyuk_B.png'),(1024,512))
Jinhyuk_B = pygame.transform.scale(pygame.image.load('images/Jinhyuk_B.png'),(1024,512))
Sungmin_R = pygame.transform.scale(pygame.image.load('images/Sungmin_R.png'),(1024,512))
Yungbum_R = pygame.transform.scale(pygame.image.load('images/Yungbum_R.png'),(1024,512))
Junhyuk_R = pygame.transform.scale(pygame.image.load('images/Junhyuk_R.png'),(1024,512))
Jinhyuk_R = pygame.transform.scale(pygame.image.load('images/Jinhyuk_R.png'),(1024,512))
    # 2.4 - 과녁 이미지 변수
```

```
dir_target = 'images\\target.png'
img_tar = pygame.transform.scale(pygame.image.load(dir_target), (50,30))
    # 2.5 - 캐릭터 이름당 주소
characters = {'sungmin':('images\character_sungmin.png',(-13,1560),(13,637)), 'junhyuk':('images\character_junhyuk.png',(-13,1560),(13,637)), 'yungbum': ('images\character_yungbum.png',(-13,1560),(13,637)), 'jinhyuk': ('images\character_jinhyuk.png',(-13,1560),(13,637))}
    # 2.6 - 장애물 screen
obstacles = {'greentree1':'images\greentree1.png','greentree2':'images\greentree2.png','greentree3':'images\greentree3.png','redtree':'images\\redtree.png','yellowtree':'images\yellowtree.png','sheep1':'images\sheep1.png','sheep2':'images\sheep2.png','sheep3':'images\sheep3.png','sheep4':'images\sheep4.png','sheep5':'images\sheep5.png'}
greentree1 = pygame.transform.scale(pygame.image.load(obstacles['greentree1']),(50,100))
greentree2 = pygame.transform.scale(pygame.image.load(obstacles['greentree2']),(50,100))
greentree3 = pygame.transform.scale(pygame.image.load(obstacles['greentree3']),(50,100))
yellowtree = pygame.transform.scale(pygame.image.load(obstacles['yellowtree']),(50,100))
redtree = pygame.transform.scale(pygame.image.load(obstacles['redtree']), (50,100))
sheep1 = pygame.transform.scale(pygame.image.load(obstacles['sheep1']), (50,50))
sheep2 = pygame.transform.scale(pygame.image.load(obstacles['sheep2']), (50,50))
sheep3 = pygame.transform.scale(pygame.image.load(obstacles['sheep3']), (50,50))
sheep4 = pygame.transform.scale(pygame.image.load(obstacles['sheep4']), (50,50))
sheep5 = pygame.transform.scale(pygame.image.load(obstacles['sheep5']), (50,50))
    # 2.7 - 운석 종류당 주소, 속도, 대미지
asteroids = {'fast':('images\\asteroid_fast.png', 4, 1), 'strong': ('images\\asteroid_strong.png', 6, 2)}
    # 2.8 - 기타 변수
        # 2.8.1 - 체력 및 특수능력 게이지
heart = pygame.transform.scale(pygame.image.load('images\heart.png'),(30,30))
cool = pygame.transform.scale(pygame.image.load('images\coolcount.png'), (40,40))
        # 2.8.2 - 음악
full = pygame.mixer.Sound('music\음원3.mp3')
oof = pygame.mixer.Sound('music\oof.mp3')
flick = pygame.mixer.Sound('music\\flick.mp3')
click = pygame.mixer.Sound('music\clicked.mp3')
explosion = pygame.mixer.Sound('music\Explosion2.mp3')
ending = pygame.mixer.Sound('music\ending.mp3')
        # 2.8.3 - 로고 및 윈도우 이름
pygame.display.set_caption('Asteroid Falls')
pygame_icon = pygame.image.load('images\logo.jpg')
pygame.display.set_icon(pygame_icon)

# 3 - 함수와 딕셔너리
    # 3.1 - 에러 시 탈출 함수
def prevent_crash():
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
```

```
        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_ESCAPE:
                sys.exit()
    # 3.2 - 스크린의 좌표, 크기를 게임 속 좌표, 크기로 변환하는 함수
def field2screen(pos, slope = 20*(math.pi/2)/90, slide = 87*(math.pi/2)/90):
          # 변환행렬 A = ((1 cosθcosϕ)(0 cosθsinϕ)) with slope = θ, slide = ϕ
    x = pos[0] - pos[1]*math.cos(slope)*math.cos(slide)
    y = pos[1]*math.cos(slope)*math.sin(slide)
    return x, y
def screen2field(pos, slope = 20*(math.pi/2)/90, slide = 87*(math.pi/2)/90):
    x = pos[0]*math.sin(slide)/(math.sin(slide)+math.cos(slide))
    y =
pos[0]*math.cos(slide)/(math.sin(slide)+math.cos(slide))+pos[1]/(math.cos(slo
pe)*(math.sin(slide)+math.cos(slide)))
    return x,y
    # 3.3 - 텍스트 출력 함수
def blit(message, fontSize, coord, color, condition = False, mode = "center",
Font = "freesansbold.ttf"):
    if Font == "freesansbold.ttf":
        font = pygame.font.Font(Font, fontSize)
    else:
        font = font = pygame.font.SysFont(Font, fontSize)
    text = font.render(str(message), True, color)
    textRect = text.get_rect()
    exec(f"textRect.{mode} = (coord[0], coord[1])")
    screen.blit(text, textRect)
    if condition:
        pygame.display.update()

# 4 - 게임 요소들 클래스
    # 4.1 - Character 클래스
class Character:
        # 4.1.1 - 변수 초기화 및 캐릭터 이미지 로딩
    def __init__(self, character = 'sungmin', fix = -10, start_loc = (240,
240)):
        try:
            self.dir = characters[character][0]
            self.character = pygame.image.load(self.dir)
            self.characterl = pygame.transform.scale(self.character, (70,
90))
            self.characterr =
pygame.transform.flip(pygame.transform.scale(self.character, (70, 90)), True,
False)
            self.start_loc = start_loc
            self.x = characters[character][1]
            self.y = characters[character][2]
            self.fix = fix
        except Exception as err:
            print('Problem loading character image: ', err)
            pygame.quit()
            exit(0)
    # 4.2 - Background 클래스
class Background:
        # 4.2.1 - 변수 초기화 및 배경 이미지 로딩
    def __init__(self, speed = 13, dir = 'images\map_dotted.png'):
        try:
            self.background = pygame.image.load(dir)
            self.background = pygame.transform.scale(self.background, (2400,
1600))
            self.backgroundpos = [-160, -280]
```

```
127             self.direction = 0
128             self.speed = speed
129         except Exception as err:
130             print('Problem loading background image: ', err)
131             pygame.quit()
132             exit(0)
133         # 4.2.2 - 움직임 나타내기
134     def move(self, ch, movex, movey):
135         x,y = 0,0
136         if keyboard.is_pressed("a"):
137             if keyboard.is_pressed("d"):                            # 반대방향 조작 상쇄
138                 pass
139             else:
140                 self.backgroundpos[0] += self.speed
141                 self.direction = 1
142                 x = -self.speed                                     # 내부좌표계 캐릭터 이동 변위
143                 if movex < ch.x[0]:                                 # 맵 밖으로 이동 못하도록 제어
144                     self.backgroundpos[0] -= self.speed
145                     x = 0
146         if keyboard.is_pressed("d"):
147             if keyboard.is_pressed("a"):
148                 pass
149             else:
150                 self.backgroundpos[0] -= self.speed
151                 self.direction = 0
152                 x = self.speed
153                 if movex > ch.x[1]:
154                     self.backgroundpos[0] += self.speed
155                     x = 0
156         if keyboard.is_pressed("w"):
157             if keyboard.is_pressed("s"):
158                 pass
159             else:
160                 self.backgroundpos[1] += self.speed
161                 y = -self.speed
162                 if movey < ch.y[0]:
163                     self.backgroundpos[1] -= self.speed
164                     y = 0
165         if keyboard.is_pressed("s"):
166             if keyboard.is_pressed("w"):
167                 pass
168             else:
169                 self.backgroundpos[1] -= self.speed
170                 y = self.speed
171                 if movey > ch.y[1]:
172                     self.backgroundpos[1] += self.speed
173                     y = 0
174         return x,y
175     # 4.3 - Asteroid 클래스
176 class Asteroid:
177         # 4.3.1 - 변수 초기화 및 소행성, 폭발 이미지 로딩
178     def __init__(self, dir_ast = 'images\\asteroid.png', freq = 6, damage = 1, dir_exp = 'images\explosion.png'):
179         self.img_ast = pygame.transform.scale(pygame.image.load(dir_ast), (50,50))
180         self.img_exp = pygame.transform.scale(pygame.image.load(dir_exp), (100, 80))
```

```
        self.freq = freq
        self.damage = damage

# 6 - 게임 작동 관련 함수
    # 6.1 - 시작 화면 함수
def startscreen(beginning = True):
    pygame.mouse.set_visible(True)
        # 6.1.1 - 음악 재생
    if beginning:
        full.play(-1)
        # 6.1.2 - 배경 blit
    screen.blit(lobbyimage,(0,0))
    pygame.display.flip()
    state = 'start'
        # 6.1.3 - 키보드 입력 판단
    prevent_crash()
    while state == 'start':
        for event in pygame.event.get():
            if event.type == pygame.MOUSEBUTTONDOWN:
                x,y = pygame.mouse.get_pos()
                if 358 <= x <= 666 and 203 <= y <= 307:
                    state = 'play'
                    click.play()
                    try:
                        state = 'setting'
                        character_choice,ability_choice = setting()
                        print(character_choice,ability_choice)
                        playerscore, asteroidscore = main(character_choice,
ability_choice)
                        print(playerscore, asteroidscore)
                        endscreen(playerscore,asteroidscore)
                    except:
                        print(traceback.format_exc(chain = True).split("\n")
[-2], 'Choose character and asteroid ability in settings')
                        state = 'start'
                elif 363 <= x <= 668 and 365 <= y <= 468:
                    state = 'explain'
                    click.play()
                    showinfo()
    # 6.2 - 게임설명 화면 함수
def showinfo():
    screen.blit(infoimage,[0,0])
    pygame.display.flip()
    state = 'explain'
    print(state)
    while state == 'explain':
        prevent_crash()
        for event in pygame.event.get():
            if event.type == pygame.KEYDOWN:
                if event.key == pygame.K_b:
                    print('returning')
                    state = 'start'
                    startscreen(False)
    # 6.3 - 게임 설정 화면 함수
def setting():
    li=[0,0,0]
    character_code = {11:'sungmin', 12:'yungbum', 13:'junhyuk', 14:'jinhyuk'}
    ability_code = {21:'fast', 22:'strong'}
    mode = None_None
    while True:
```

```
239         if li[2] == 1 and li[0] != 0 and li[1] != 0:
240                 break
241         else:
242             li[2] = 0
243         for event in pygame.event.get():
244             screen.blit(mode,[0,0])
245             if event.type == pygame.MOUSEBUTTONDOWN:
246                 click.play()
247                 x=pygame.mouse.get_pos()[0]
248                 y=pygame.mouse.get_pos()[1]
249                 if 126<x<223 and 47<y<173: #이성민
250                     li.pop(0)
251                     li.insert(0,11)
252                     if li[1]== 0 :
253                         mode=Sungmin_None
254                     elif li[1]== 21 :
255                         mode=Sungmin_B
256                     elif li[1]== 22 :
257                         mode=Sungmin_R
258                 elif 389<x<487 and 48<y<171: #진영범
259                     li.pop(0)
260                     li.insert(0,12)
261                     if li[1]== 0 :
262                         mode=Yungbum_None
263                     elif li[1]== 21 :
264                         mode=Yungbum_B
265                     elif li[1]== 22 :
266                         mode=Yungbum_R
267                 elif 126<x<227 and 295<y<417: #안준혁
268                     li.pop(0)
269                     li.insert(0,13)
270                     if li[1]== 0 :
271                         mode=Junhyuk_None
272                     elif li[1]== 21 :
273                         mode=Junhyuk_B
274                     elif li[1]== 22 :
275                         mode=Junhyuk_R
276                 elif 389<x<487 and 295<y<417: #최진혁
277                     li.pop(0)
278                     li.insert(0,14)
279                     if li[1]== 0 :
280                         mode=Jinhyuk_None
281                     elif li[1]== 21 :
282                         mode = Jinhyuk_B
283                     elif li[1]== 22 :
284                         mode = Jinhyuk_R
285                 elif 674<x<788 and 47<y<167: #B rock
286                     li.pop(1)
287                     li.insert(1,21)
288                     if li[0]== 0 :
289                         mode=None_B
290                     elif li[0]== 11 :
291                         mode=Sungmin_B
292                     elif li[0]== 12 :
293                         mode=Yungbum_B
294                     elif li[0]== 13 :
295                         mode=Junhyuk_B
296                     elif li[0]== 14 :
297                         mode=Jinhyuk_B
298                 elif 674<x<788 and 295<y<416: #R rock
```

```
                    li.pop(1)
                    li.insert(1,22)
                    if li[0]== 0 :
                        mode=None_R
                    elif li[0]== 11 :
                        mode=Sungmin_R
                    elif li[0]== 12 :
                        mode=Yungbum_R
                    elif li[0]== 13 :
                        mode=Junhyuk_R
                    elif li[0]== 14 :
                        mode=Jinhyuk_R
                elif 448<x<560 and 451<y<493: #confirm
                    li.pop(2)
                    li.insert(2,1)
        pygame.display.flip()
    return character_code[li[0]], ability_code[li[1]]
    # 6.4 - 종료 화면 함수
def endscreen(playerscore, asteroidscore):
    full.stop()
    ending.play()
    screen.fill((0,0,0))
    screen.blit(endingimage,(0,0))
    blit('End of Game',40,(512,100),(255,0,0), condition=True,mode='center')
    blit(f'Player Score : {playerscore}',20,(512,184),(50,50,255),
condition=True,mode='center')
    blit(f'Asteroid Score : {asteroidscore}',20,(512,201),(50,50,255),
condition=True,mode='center')
    blit('Press Spacebar to Restart', 30, (512, 300), (100,255,100),
condition=True,mode='center')
    state = 'end'
    # SPACEBAR 누르면 다시 처음으로 돌아가기
    while state == 'end':
        prevent_crash()
        events = pygame.event.get()
        for event in events:
            if event.type == pygame.KEYDOWN:
                if event.key == pygame.K_SPACE:
                    ending.stop()
                    startscreen()
    # 6.5 - 게임 메인 함수
def main(character, ablility):                          # 특수능력 이름을 파라미터로 함
        # 6.5.1 - 맵 위 오브젝트 게임내 좌표설정
    mapping = [(yellowtree, 555, 457)]
    mapping.append((yellowtree, 1012, 607))
    mapping.append((yellowtree, 1619, 481))
    mapping.append((greentree1, 675 ,360))
    mapping.append((greentree1, 415, 445))
    mapping.append((greentree1, 518, 613))
    mapping.append((greentree1, 800, 465))
    mapping.append((greentree1, 1490, 503))
    mapping.append((greentree1, 1047, 720))
    mapping.append((greentree2, 885, 338))
    mapping.append((greentree2, 1290, 406))
    mapping.append((greentree3, 811, 645))
    mapping.append((greentree3, 1640, 410))
    mapping.append((greentree3, 1750, 650))
    mapping.append((redtree, 505, 785))
    mapping.append((redtree, 915, 300))
    mapping.append((redtree, 1353, 300))
```

```
    mapping.append((sheep1, 640, 535))
    mapping.append((sheep2, 525, 365))
    mapping.append((sheep3, 798, 300))
    mapping.append((sheep1, 885, 730))
    mapping.append((sheep4, 1087, 485))
    mapping.append((sheep5, 1310, 568))
    mapping.append((sheep1, 1275, 355))
    mapping.append((sheep3, 1465, 423))
    mapping.append((sheep4, 1684, 716))
    mapping.append((sheep2, 350, 735))
        # 6.5.2 - 반복문 이전 변수 설정
    ch = Character(character)
    health = 5           # P1 체력
    coolcount = 0        # 운석 사용가능 특수능력 개수 변수
    cooltime = 0         # 운석 특수능력 쿨타임 측정 변수
    hit_range = 50       # 충돌 인정 범위
    player_score = 0     # P1 점수
    trial, hit = 0,0     # P2 점수 계산을 위한 변수
    asteroid_score = 0   # P2 점수
    framet = 0           # 운석 떨어지는 애니메이션을 위한 변수
    moveX, moveY = 0,0   # 시작 이후 캐릭터의 이동상태를 나타내는 변수
    fix = ch.fix         # tweaking 변수
    movex, movey = 0,0   # 운석 시작 이후 캐릭터의 이동상태를 나타내는 변수
    falling = False      # 운석 상태 변수
    fall_state = ''      # 운석 떨어지는 상태를 저장하기 위한 str 변수
    state = 'play'
        # 6.5.3 - 캐릭터 선택에 따른 능력 연관 변수 재설정, 배경 인스턴스 설정
    if character == 'sungmin':
        pass
        bg = Background()
    elif character == 'junhyuk':
        health = 8
        bg = Background()
    elif character == 'yungbum':
        hit_range = 70
        bg = Background()
    elif character == 'jinhyuk':
        bg = Background(15)
        # 6.5.4 - 반복문
    while state == 'play':
        prevent_crash()
        screen.fill((255,255,255))
        pygame.mouse.set_visible(False)
        movement = bg.move(ch, moveX, moveY)     # 캐릭터 이동 인식
        moveX += movement[0]                     # 캐릭터 변위 저장(x)
        moveY += movement[1]                     # 캐릭터 변위 저장(y)
        if bg.direction == 0:                    # 캐릭터 이동 방향에 따라 좌/우
형태 변경
            charaterscreen = ch.characterl
        else:
            charaterscreen = ch.characterr
        pos = pygame.mouse.get_pos()
        poss = field2screen((screen2field(pos)[0], screen2field(pos)[1]))
        mapping += [(charaterscreen, screen2field(ch.start_loc)[0]+moveX,
screen2field(ch.start_loc)[1]+moveY+fix)]     # mapping 변수에 캐릭터 스크린 추가
        if mouse.is_pressed('left'):    # 마우스 클릭 시 운석 낙하 시작
            if framet == 0:
                trial += 1
                end_pos = pygame.mouse.get_pos()[0]-20,
pygame.mouse.get_pos()[1]
```

```
                start_pos = (1024, 0)
                falling = True
                ast = Asteroid()          # Asteroid 인스턴스 생성
        if mouse.is_pressed('right'):     # 우클릭 시 특수능력 발동
            if coolcount > 0:             # 특수능력이 사용 가능할 때만 발동
                if framet == 0:
                    coolcount -= 1
                    trial += 1
                    end_pos = pygame.mouse.get_pos()[0]-20, pygame.mouse.get_pos()[1]-10
                    start_pos = (1024, 0)
                    falling = True
                    ast = Asteroid(asteroids[ablility][0], asteroids[ablility][1], asteroids[ablility][2])
            else:
                pass
        if falling:
            if framet<ast.freq:           # 지면에 충돌 전 상태
                if framet == ast.freq - 1:
                    explosion.play()
                framet += 1
                movex += movement[0]
                movey += movement[1]
                astpos = [0,0]
                astpos[0] = 1024+(end_pos[0]-start_pos[0])*framet//ast.freq - movex
                astpos[1] = (end_pos[1]-start_pos[1])*framet//ast.freq - movey
                mapping.append((ast.img_ast, astpos[0], astpos[1]))
                fall_state = 'falling'
            elif framet < ast.freq+2:     # 지면 충돌 후 폭발 이미지 표현
                framet += 1
                mapping.append((ast.img_exp, astpos[0], astpos[1]))
                fall_state = 'explosion'
            elif framet == ast.freq+2:    # 충돌 여부 판단
                if (astpos[0]-hit_range <= ch.start_loc[0] <= astpos[0]+hit_range) and (astpos[1]-100 <= ch.start_loc[1] <= astpos[1]):
                    print('Hit!')
                    hit += 1
                    health -= ast.damage
                framet += 1
                mapping.append((ast.img_exp, astpos[0], astpos[1]))
                fall_state = 'collisionCheck'
            else:                         # 운석 상태 변수 초기화
                framet,movex,movey = 0,0,0
                fall_state = ''
                falling = False
        screen.blit(bg.background, field2screen(bg.backgroundpos))      # 배경 blit
        screen.blit(img_tar, poss)        # 화면에 과녁 이미지 표시
        mapping.sort(key = lambda x:x[2])       # mapping 변수 내 screen들의 위치순서 판단, blit할 순서대로 리스트 sort
        if fall_state == 'falling':       # fall_state에 따라 mapping 내부 특별한 조작을 해줘야 하는 변수 종류가 달라짐
            for i in mapping:
                if i[0] == charaterscreen:
                    screen.blit(charaterscreen, ch.start_loc)
                elif i[0] == sheep1 or i[0] == sheep2 or i[0] == sheep3 or i[0] == sheep4 or i[0] == sheep5:
```

```
463                     screen.blit(i[0], field2screen((i[1]-moveX, i[2]-
moveY+50)))
464                 elif i[0] == ast.img_ast:
465                     screen.blit(ast.img_ast, field2screen(astpos))
466                 else:
467                     screen.blit(i[0], field2screen((i[1]-moveX,i[2]-moveY)))
468             # mapping 내 일회성 변수 지우기
469             mapping.remove((charaterscreen, screen2field(ch.start_loc)
[0]+moveX, screen2field(ch.start_loc)[1]+moveY+fix))
470             mapping.remove((ast.img_ast, astpos[0], astpos[1]))
471         elif fall_state:
472             for i in mapping:
473                 if i[0] == charaterscreen:
474                     screen.blit(charaterscreen, ch.start_loc)
475                 elif i[0] == sheep1 or i[0] == sheep2 or i[0] == sheep3 or
i[0] == sheep4 or i[0] == sheep5:
476                     screen.blit(i[0], field2screen((i[1]-moveX, i[2]-
moveY+50)))
477                 elif i[0] == ast.img_exp:
478                     astpos = i[1], i[2]
479                     screen.blit(i[0], (field2screen(astpos)[0]-22,
field2screen(astpos)[1]-30))
480                 else:
481                     screen.blit(i[0], field2screen((i[1]-moveX,i[2]-moveY)))
482             mapping.remove((charaterscreen, screen2field(ch.start_loc)
[0]+moveX, screen2field(ch.start_loc)[1]+moveY+fix))
483             mapping.remove((ast.img_exp, astpos[0], astpos[1]))
484         else:
485             for i in mapping:
486                 if i[0] == charaterscreen:
487                     screen.blit(charaterscreen, ch.start_loc)
488                 elif i[0] == sheep1 or i[0] == sheep2 or i[0] == sheep3 or
i[0] == sheep4 or i[0] == sheep5:
489                     screen.blit(i[0], field2screen((i[1]-moveX, i[2]-
moveY+50)))
490                 else:
491                     screen.blit(i[0], field2screen((i[1]-moveX,i[2]-moveY)))
492             mapping.remove((charaterscreen, screen2field(ch.start_loc)
[0]+moveX, screen2field(ch.start_loc)[1]+moveY+fix))
493         # 체력 및 쿨타임 업데이트
494         for i in range(health):
495             screen.blit(heart, (42*i,0))
496         if coolcount:
497             for i in range(coolcount):
498                 screen.blit(cool, (1024-42*(i+1),0))
499         fpsClock.tick(FPS)
500         player_score += 1
501         if coolcount != 3:
502             cooltime += 1
503         if cooltime % 100 == 0:      # 주기적으로 특수능력 발동 가능 횟수 증가
504             if coolcount < 3:        # 최대 3번까지 발동 가능 횟수 누적 가능
505                 coolcount += 1
506         pygame.display.update()
507         if health <= 0:
508             break
509     asteroid_score = int(player_score*4.5*hit//trial)
510     print(player_score,asteroid_score)
511     return player_score,asteroid_score
512 if __name__ == '__main__':      # 이 파일에서 실행 시 startscreen() 실행
513     startscreen()
```