KAIST

# PYTHON TUTORING #4

School of Computing, KAIST & 대덕고등학교

# INTRO

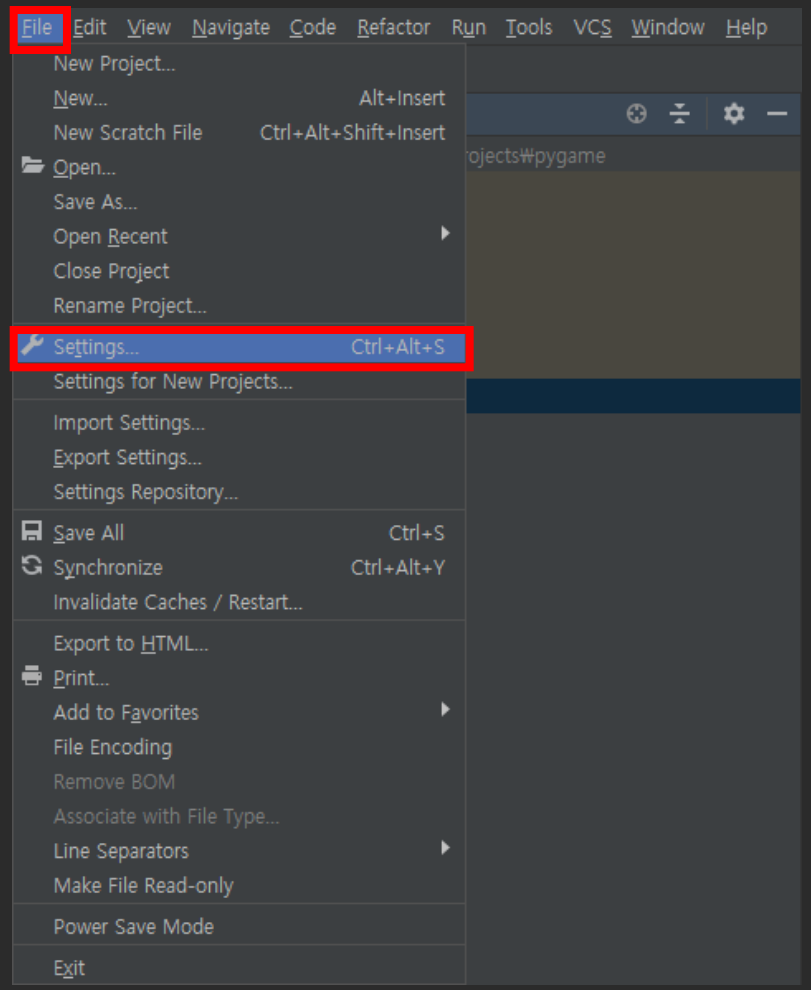
① Pygame 라이브러리 설치

② 강의에 필요한 이미지 파일 다운로드

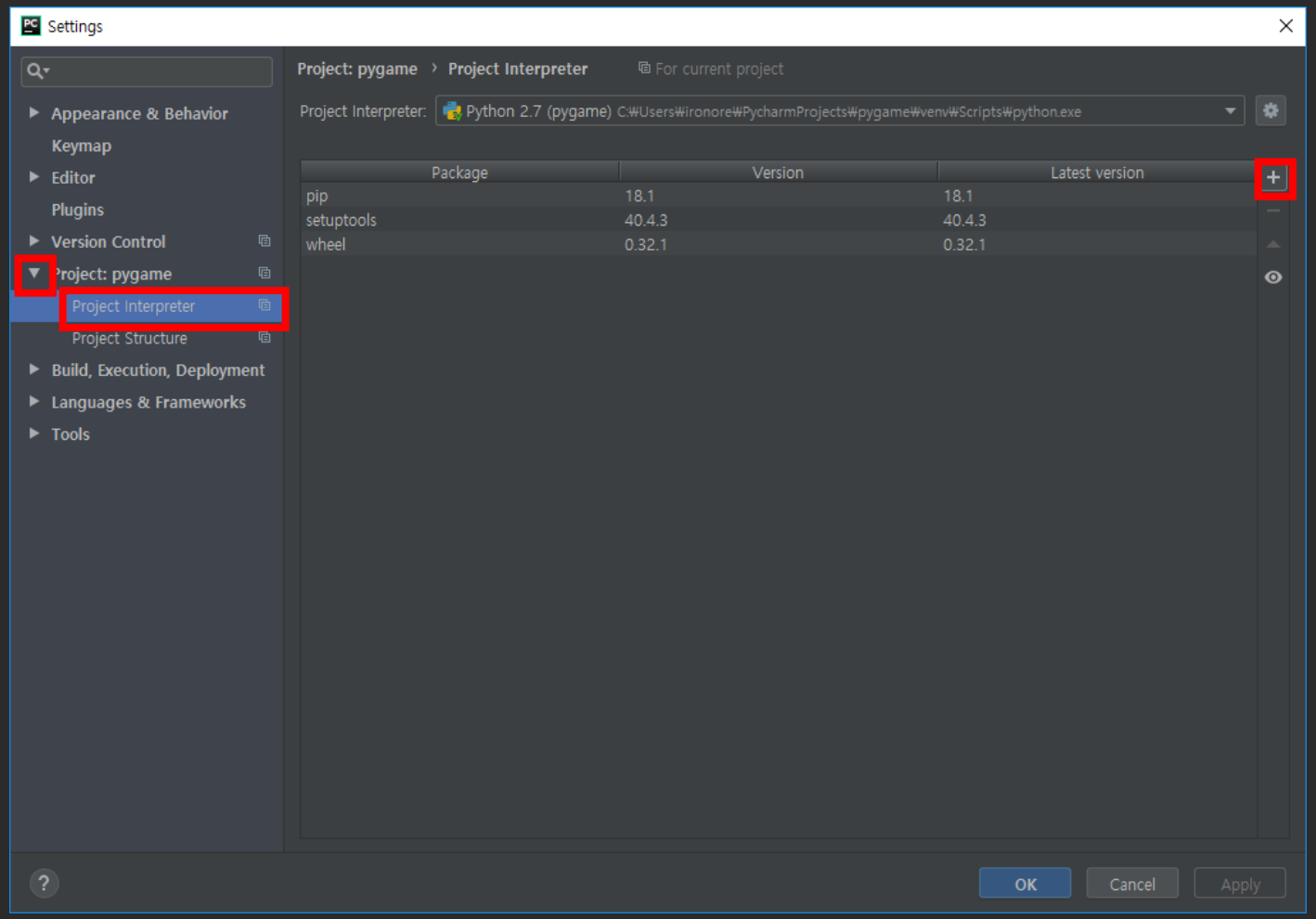
③ GALAGA 구현하기 (2)

④ Object Oriented Programming

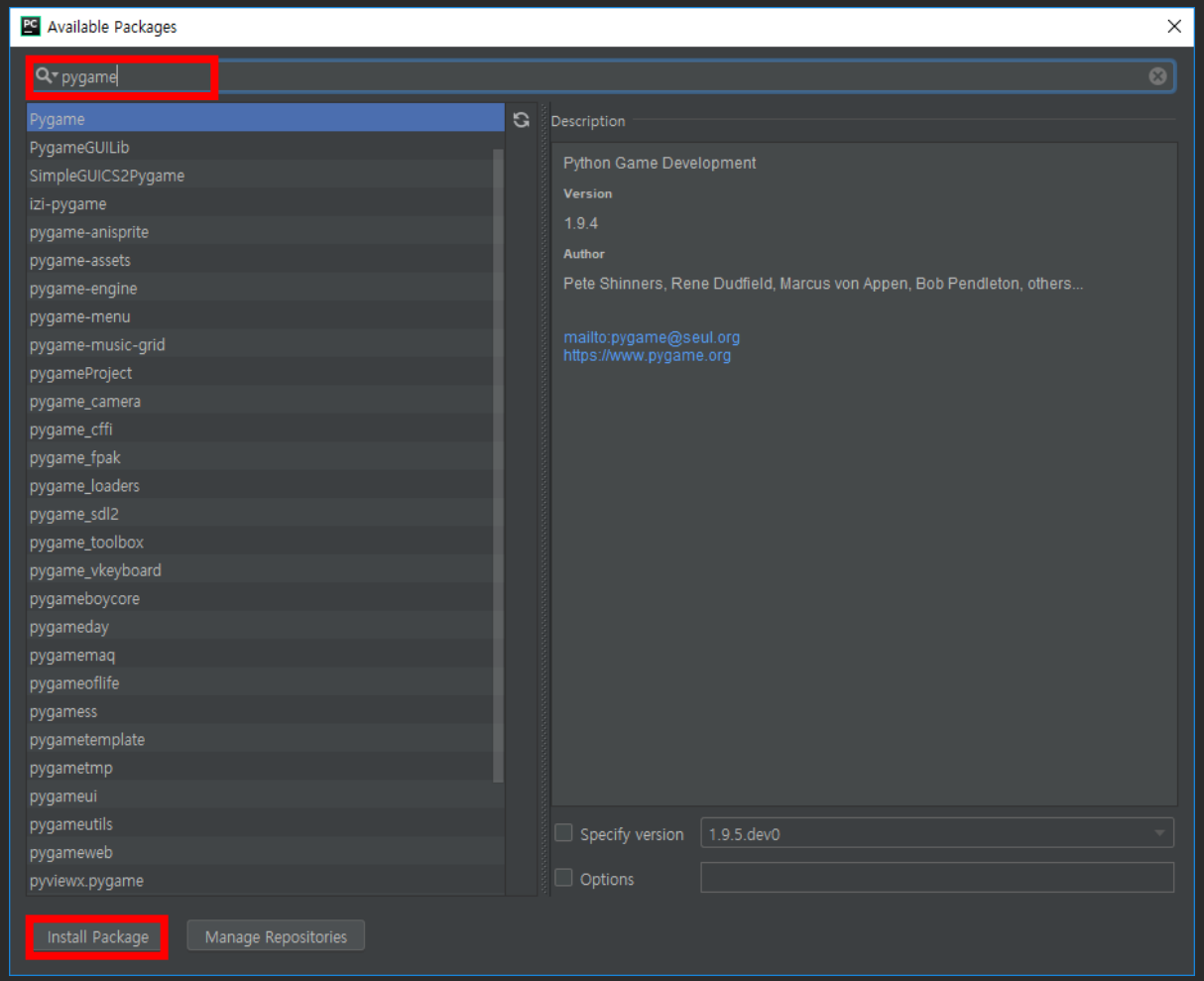
# Pygame 라이브러리 설치



# Pygame 라이브러리 설치

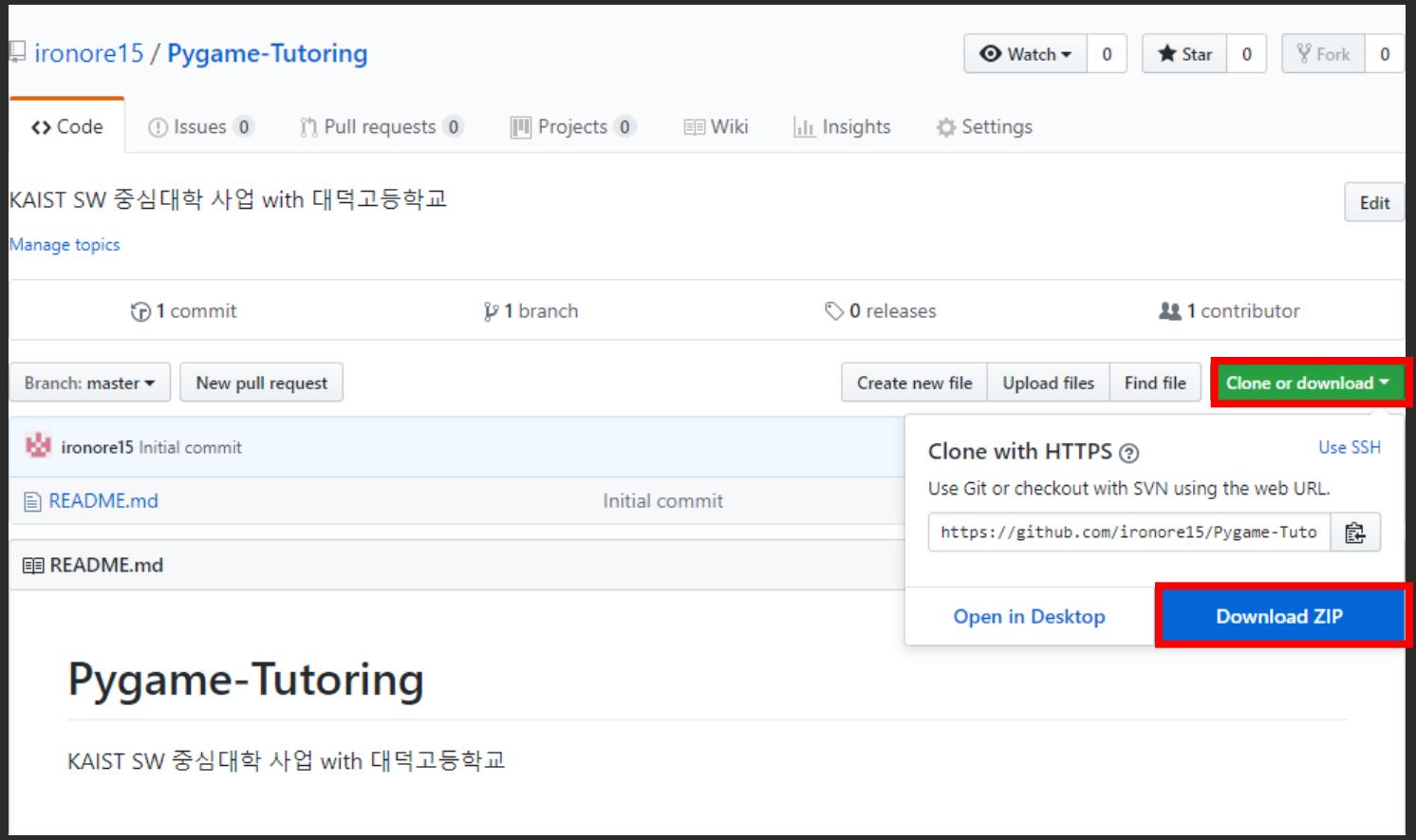


# Pygame 라이브러리 설치

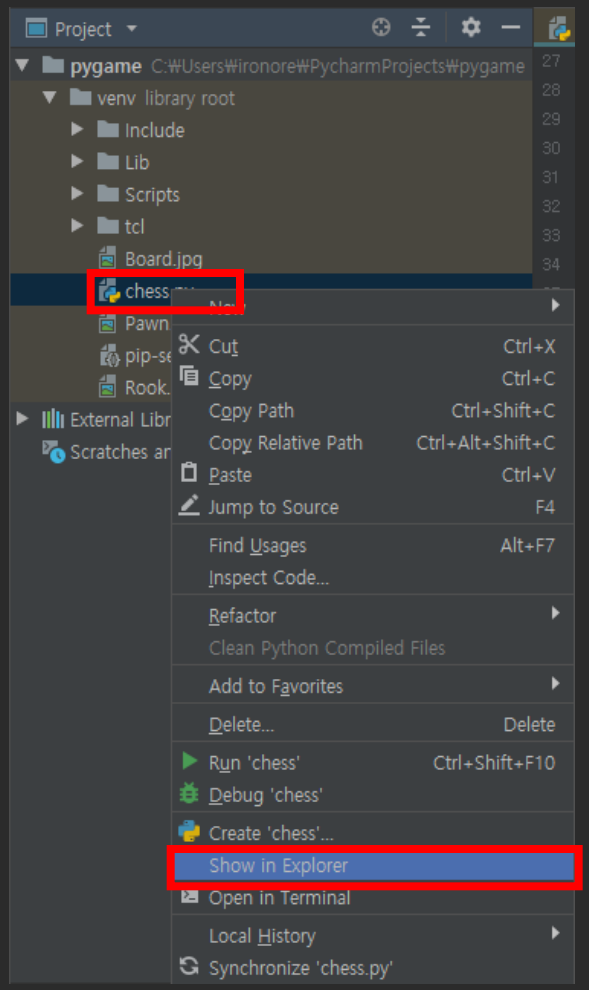


# 이미지 파일 다운로드

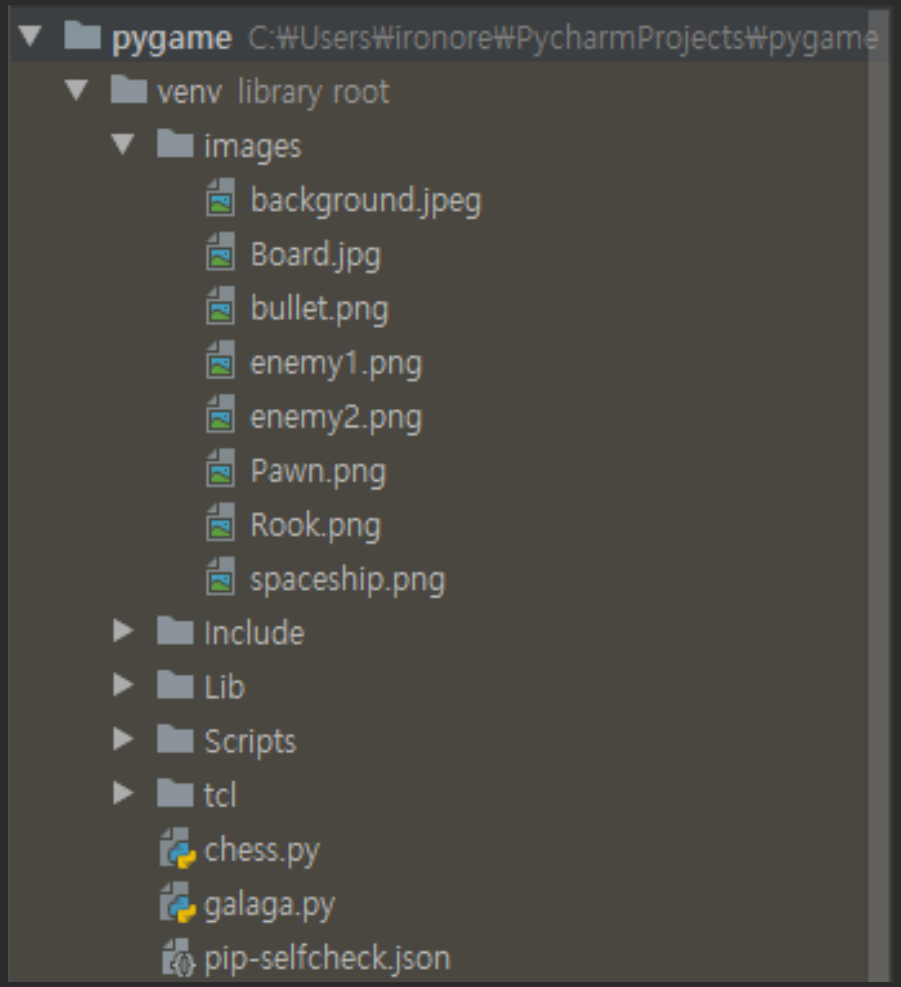
https://github.com/313usually/Pygame-Tutoring



# 이미지 파일 다운로드


Project
pygame C:₩Users₩ironore₩PycharmProjects₩pygame
venv library root
Include
Lib
Scripts
tcl
Board.jpg
Pawn
External Libr
Scratches ar
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Relative Path Ctrl+Alt+Shift+C
Paste Ctrl+V
Jump to Source F4
Find Usages Alt+F7
Inspect Code...
Refactor
Clean Python Compiled Files
Add to Favorites
Delete... Delete
Run 'chess' Ctrl+Shift+F10
Debug 'chess'
Create 'chess'...
Show in Explorer
Open in Terminal
Local History
Synchronize 'chess.py'

# 이미지 파일 다운로드


pygame C:₩Users₩ironore₩PycharmProjects₩pygame
venv library root
images
background.jpeg
Board.jpg
bullet.png
enemy1.png
enemy2.png
Pawn.png
Rook.png
spaceship.png
Include
Lib
Scripts
tcl
chess.py
galaga.py
pip-selfcheck.json

# Create enemies (1)

```
1  import random
2
3  def createEnemy():
4      if random.randint(0, 120) == 0:
5          enemy_rect = enemy.get_rect()
6          enemy_rect.center = (random.randint(0, width), 0)
7          enemy_list.append(enemy_rect)
8
9  enemy = pygame.image.load('images\\enemy1.png')
10 enemy_list = []
11
12 score = 0
```

# Create enemies (2)

```
1
2  def drawScreen():
3      screen.blit(background, (0, 0))
4      screen.blit(spaceship, ship_rect)
5      for bullet_rect in bullet_list:
6          screen.blit(bullet, bullet_rect)
7      for enemy_rect in enemy_list:
8          screen.blit(enemy, enemy_rect)
9      pygame.display.flip()
10     clock.tick(60)
11
12
```

# Create enemies (3)

```
1
2  while True:
3      (...)
4
5      createEnemy()
6      ship_rect.move_ip(ship_dx, ship_dy)
7      moveBullets()
8      drawScreen()
9
10
11
12
```

# Move enemies

```
1  def moveEnemies():
2      for enemy_rect in enemy_list:
3          enemy_rect.move_ip(0, 3)
4
5  while True:
6      (...)
7      createEnemy()
8      ship_rect.move_ip(ship_dx, ship_dy)
9      moveEnemies()
10     moveBullets()
11     drawScreen()
12
```

# Remove enemy (1)

```
1  def removeEnemy(enemy_rect):
2      global score
3      for bullet_rect in bullet_list:
4          if enemy_rect.colliderect(bullet_rect):
5              score += 800
6              bullet_list.remove(bullet_rect)
7              return True
8
9      if enemy_rect.top > height:
10         score -= 200
11         return True
12
       return False
```

# Remove enemy (2)

```
while True:
    (...)

    createEnemy()
    ship_rect.move_ip(ship_dx, ship_dy)
    moveEnemy()
    moveBullets()
    for enemy_rect in enemy_list[:]:
        if removeEnemy(enemy_rect):
            enemy_list.remove(enemy_rect)
    drawScreen()
```

# Remove bullets

```
1  def removeBullet():
2      for bullet_rect in bullet_list[:]:
3          if bullet_rect.bottom < 0:
4              bullet_list.remove(bullet_rect)
5
6  while True:
7      (...)
8      for enemy_rect in enemy_list[:]:
9          if removeEnemy(enemy_rect):
10             enemy_list.remove(enemy_rect)
11     removeBullet()
12     drawScreen()
```

# Check collision

```
1  def checkCrash():
2      for enemy_rect in enemy_list:
3          if ship_rect.colliderect(enemy_rect):
4              gameOver()
5
6  while True:
7      (...)
8      moveBullets()
9      checkCrash()
10     for enemy_rect in enemy_list[:]:
11     (...)
12
```

# Game over

```
def gameOver():
    text = game_font.render("GAME OVER!", True, (255, 0, 0))
    text_rect = text.get_rect()
    text_rect.center = (width / 2, height / 2)
    screen.blit(background, (0, 0))
    screen.blit(text, text_rect)
    pygame.display.flip()
    pygame.time.wait(2000)
    exit()
screen = pygame.display.set_mode(size)
game_font = pygame.font.SysFont(None, 80)
score_font = pygame.font.SysFont(None, 20)
```

# Show score

```
1  def showScore():
2      text = 'Score: ' + str(score)
3      text = score_font.render(text, True, (255, 255, 0))
4      text_rect = text.get_rect()
5      text_rect.center = (width / 2, 30)
6      screen.blit(text, text_rect)
7
8  def drawScreen():
9      (...)
10     showScore()
11     pygame.display.flip()
12     clock.tick(60)
```

# Setting difficulty (1)

```
1
2  def moveEnemies():
3      speed = 1 + score // 6000
4      if speed > 8:
5          speed = 8
6      elif speed <= 0:
7          speed = 1
8      for enemy_rect in enemy_list:
9          enemy_rect.move_ip(0, speed)
10
11
12
```

# Setting difficulty (2)

```
1
2  def createEnemy():
3      if score < 0:
4          prob = 120
5      else:
6          prob = 30 + 90 // (1 + score // 5000)
7      if random.randint(0, prob) == 0:
8          enemy_rect = enemy.get_rect()
9          enemy_rect.center = (random.randint(0, width), 0)
10         enemy_list.append(enemy_rect)
11
12
```