

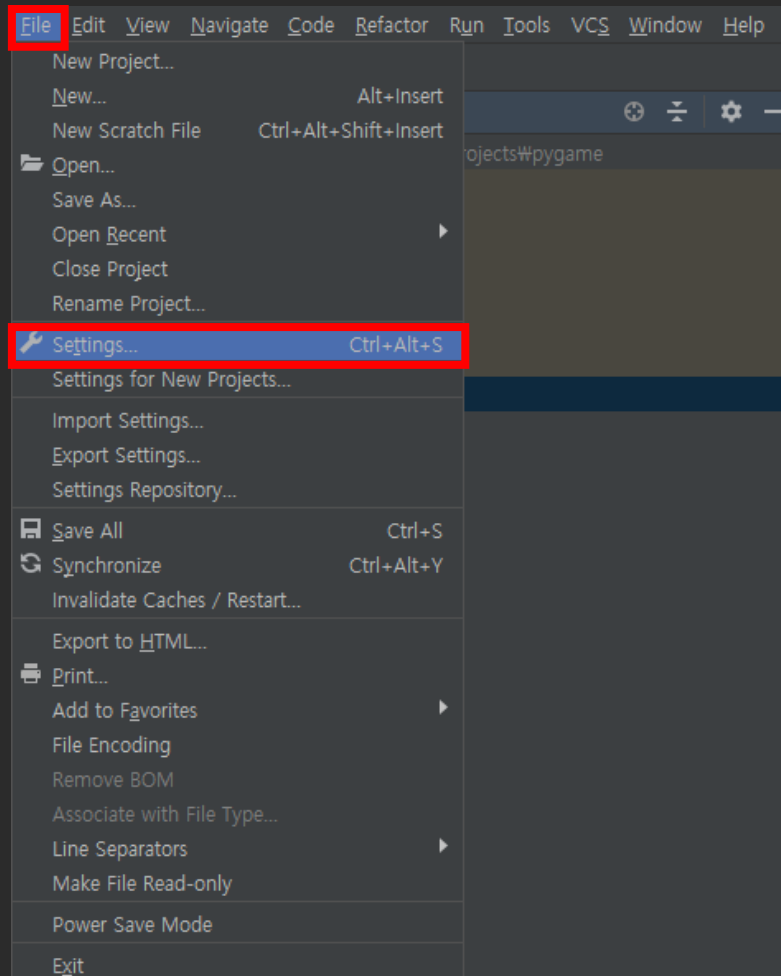
PYTHON TUTORING #2

School of Computing, KAIST & 대덕고등학교

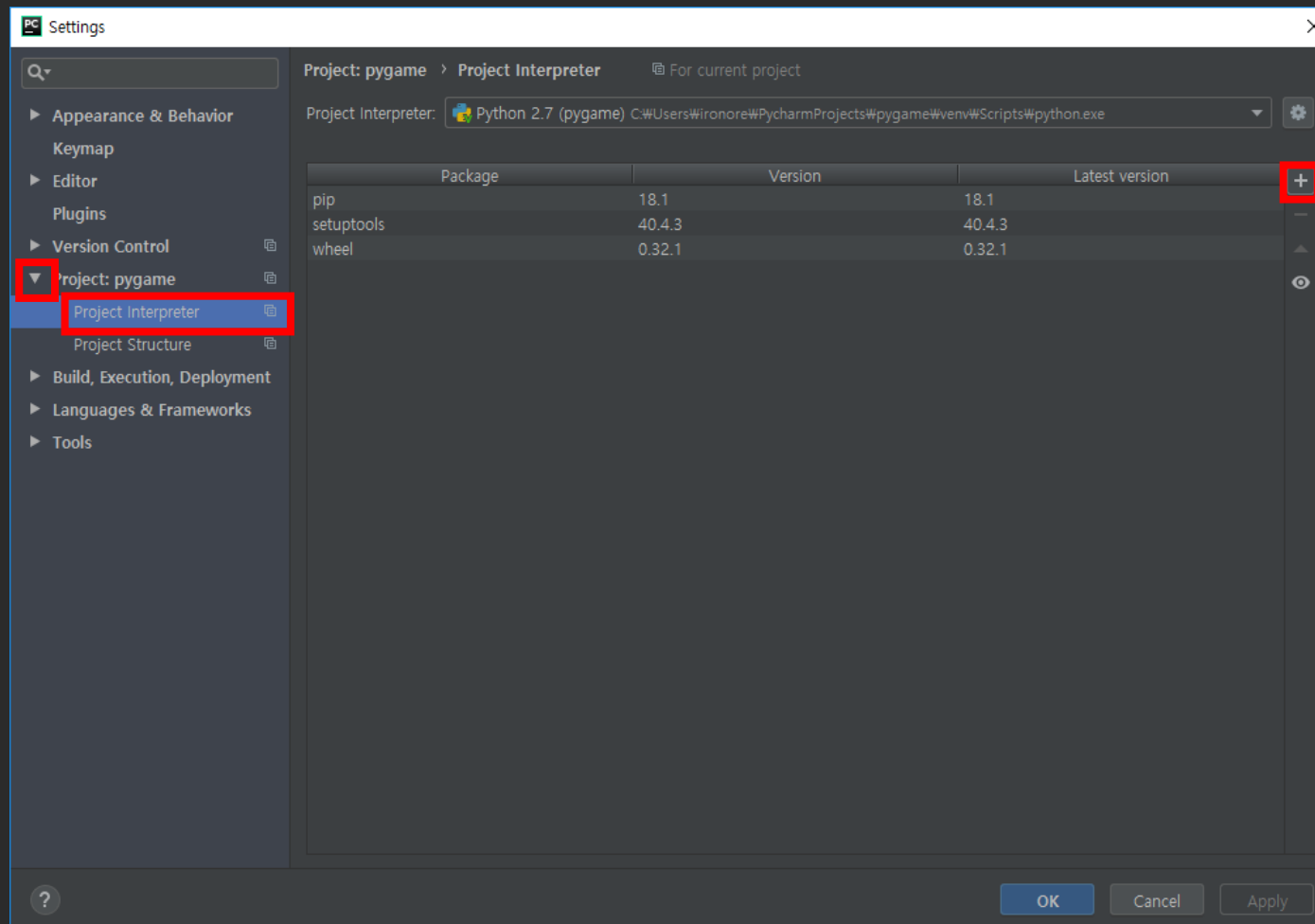
INTRO

- ① Pygame 라이브러리 설치
- ② 강의에 필요한 이미지 파일 다운로드
- ③ Pygame으로 Chess 구현하기 (1)

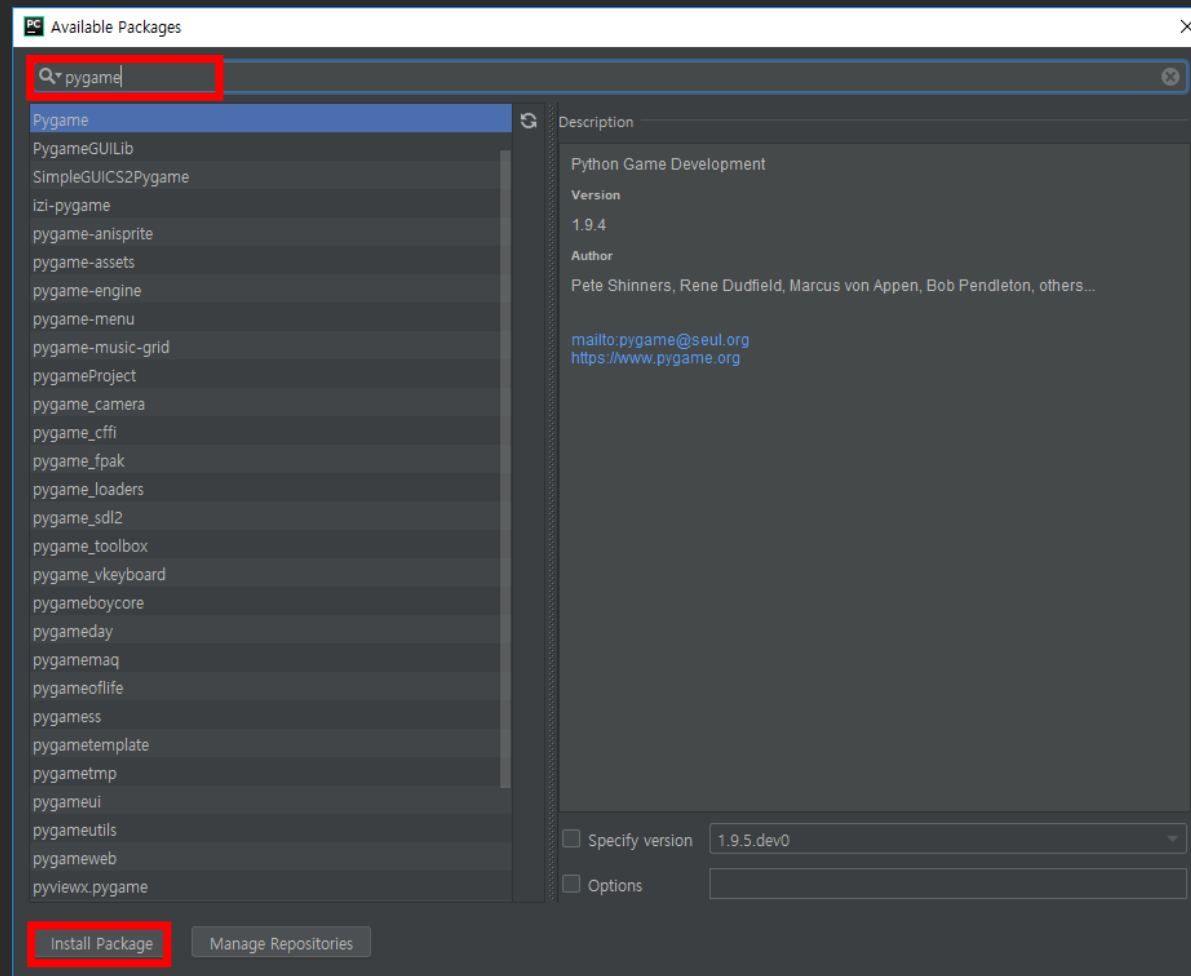
Pygame 라이브러리 설치



Pygame 라이브러리 설치

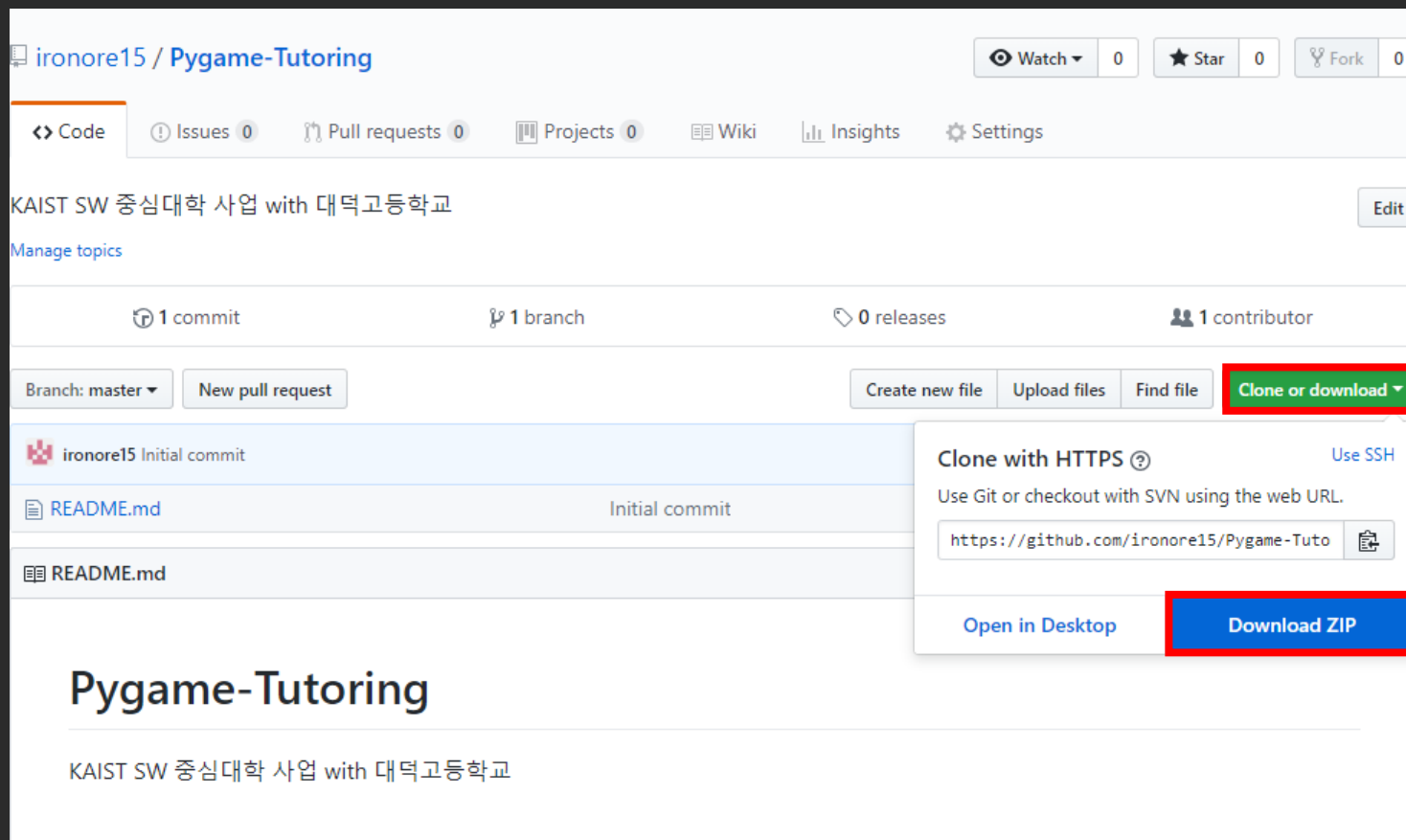


Pygame 라이브러리 설치



이미지 파일 다운로드

<https://github.com/ironore15/Pygame-Tutoring>



ironore15 / Pygame-Tutoring

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

KAIST SW 중심대학 사업 with 대덕고등학교

Manage topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request

Create new file Upload files Find file Clone or download

ironore15 Initial commit

README.md Initial commit

README.md

Pygame-Tutoring

KAIST SW 중심대학 사업 with 대덕고등학교

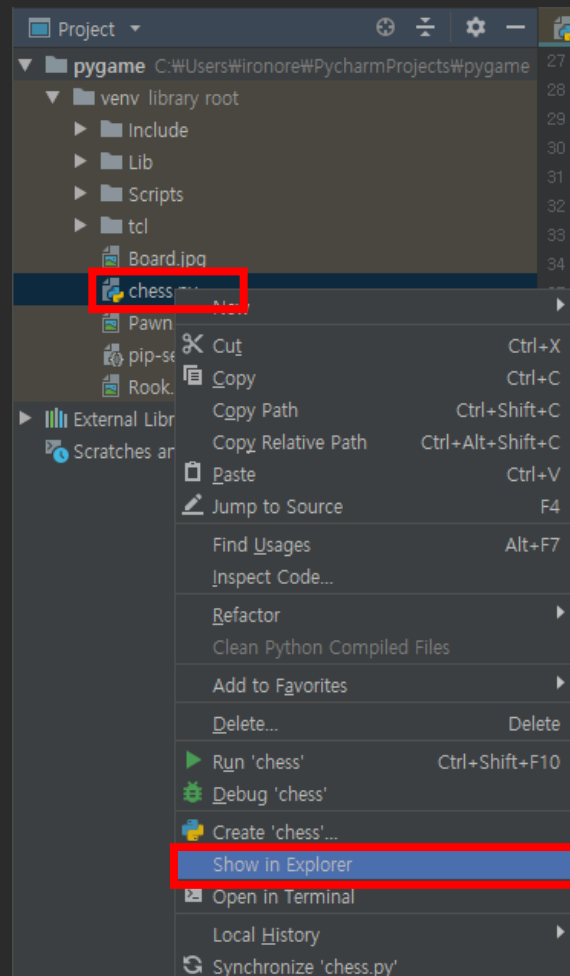
Clone with HTTPS Use SSH

Use Git or checkout with SVN using the web URL.

<https://github.com/ironore15/Pygame-Tuto>

Open in Desktop Download ZIP

이미지 파일 다운로드



Start Pygame

```
1 import pygame
2
3 pygame.init()
4
5
6
7
8
9
10 # Process finished with exit code 0
11
12
```


Make a screen

```
1 import pygame
2
3 pygame.init()
4
5 width = 800
6 height = 800
7 width, height = 800, 800
8
9 size = (width, height)
10
11 screen = pygame.display.set_mode(size)
12
```

Keep screen alive

```
1 import pygame
2
3 (...)
4
5 screen = pygame.display.set_mode(size)
6
7 while True:
8     for event in pygame.event.get():
9         if event.type == pygame.QUIT:
10             exit()
11
12
```

Load images

```
1 (...)  
2  
3 screen = pygame.display.set_mode(size)  
4  
5 board = pygame.image.load("Board.jpg")  
6 screen.blit(board, (0, 0))  
7 pygame.display.flip()  
8  
9 while True:  
10 (...)  
11  
12
```

Resize images

```
1 (...)
2
3 screen = pygame.display.set_mode(size)
4
5 board = pygame.image.load("Board.jpg")
6 board = pygame.transform.scale(board, size)
7 screen.blit(board, (0, 0))
8 pygame.display.flip()
9
10 while True:
11     (...)
12
```

Handle events

```
1 (...)  
2  
3 pawn = pygame.image.load("Pawn.png")  
4 pawn = pygame.transform.scale(pawn, (width // 8, height // 8))  
5  
6 while True:  
7     for event in pygame.event.get():  
8         if event.type == pygame.QUIT:  
9             exit()  
10        elif event.type == pygame.KEYDOWN:  
11            screen.blit(pawn, (0, 0))  
12            pygame.display.flip()
```

Make pawns!

```
1 dw = width // 8
2 dh = height // 8
3 count = 0
4
5 while True:
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             exit()
9         elif event.type == pygame.KEYDOWN:
10             screen.blit(pawn, (dw*(count % 8), dh*(count // 8)))
11             count += 1
12             pygame.display.flip()
```

Handle mouse events

```
1 click = False
2 rook = pygame.image.load("Rook.png")
3 rook = pygame.transform.scale(rook, (dw, dh))
4
5 while True:
6     for event in pygame.event.get():
7         (...)
8         elif event.type == pygame.MOUSEBUTTONDOWN:
9             click = True
10        elif event.type == pygame.MOUSEBUTTONUP:
11            click = False
12
```

Handle mouse events

```
1 while True:
2     for event in pygame.event.get():
3         (...)
4         elif event.type == pygame.MOUSEBUTTONDOWN:
5             click = True
6         elif event.type == pygame.MOUSEBUTTONUP:
7             click = False
8
9     if click:
10        screen.blit(rook, pygame.mouse.get_pos())
11        pygame.display.flip()
12
```


Center the rook

```
1 rect = rook.get_rect()
2
3 while True:
4
5     (...)
6
7     if click:
8         rect.center = pygame.mouse.get_pos()
9         screen.blit(rook, rect)
10        pygame.display.flip()
11
12
```

Erase and display

```
1 rect = rook.get_rect()
2
3 while True:
4
5     (...)
6
7     if click:
8         rect.center = pygame.mouse.get_pos()
9         screen.blit(board, (0, 0))
10        screen.blit(rook, rect)
11        pygame.display.flip()
12
```

Is mouse inside rook?

```
1 while True:
2     for event in pygame.event.get():
3
4         (...)
5
6         elif event.type == pygame.MOUSEBUTTONDOWN:
7
8             (...)
9
10            if condition:
11                click = True
12
```

Is mouse inside rook?

```
1 while True:
2     for event in pygame.event.get():
3         (...)
4         elif event.type == pygame.MOUSEBUTTONDOWN:
5             (...)
6             if condition:
7                 click = True
8
9 USE:
10 x, y = mouse.get_pos()
11 rect.left and rect.right
12 rect.width and rect.height
```

Is mouse inside rook?

```
1 rect = rook.get_rect()
2
3 while True:
4     for event in pygame.event.get():
5         elif event.type == pygame.MOUSEBUTTONDOWN:
6             if rect.collidepoint(pygame.mouse.get_pos()):
7                 click = True
8     if click:
9         rect.center = pygame.mouse.get_pos()
10        screen.blit(board, (0, 0))
11        screen.blit(rook, rect)
12        pygame.display.flip()
```

Is mouse inside rook?

```
1 # Show rook image at first display!!!
2 rook = pygame.image.load("Rook.png")
3 rook = pygame.transform.scale(rook, (dw, dh))
4 rect = rook.get_rect()
5
6 screen.blit(board, (0, 0))
7 screen.blit(rook, rect)
8 pygame.display.flip()
9
10 while True:
11     (...)
12
```

Move with keyboard

```
1 while True:
2     for event in pygame.event.get():
3         (...)
4         elif event.type == pygame.KEYUP:
5             if event.key == pygame.K_LEFT:
6                 rect = rect.move(-5, 0)
7             elif event.key == pygame.K_RIGHT:
8                 rect = rect.move(5, 0)
9             elif event.key == pygame.K_UP:
10                rect = rect.move(0, -5)
11            elif event.key == pygame.K_DOWN:
12                rect = rect.move(0, 5)
```

Set rook always

```
1 while True:
2     for event in pygame.event.get():
3         (...)
4         elif event.type == pygame.KEYUP:
5             if event.key == pygame.K_LEFT:
6                 rect = rect.move(-5, 0)
7             (...)
8         if click:
9             rect.center = pygame.mouse.get_pos()
10        screen.blit(board, (0, 0))
11        screen.blit(rook, rect)
12        pygame.display.flip()
```