KAIST

PYTHON TUTORING #2

School of Computing, KAIST & 대덕고등학교

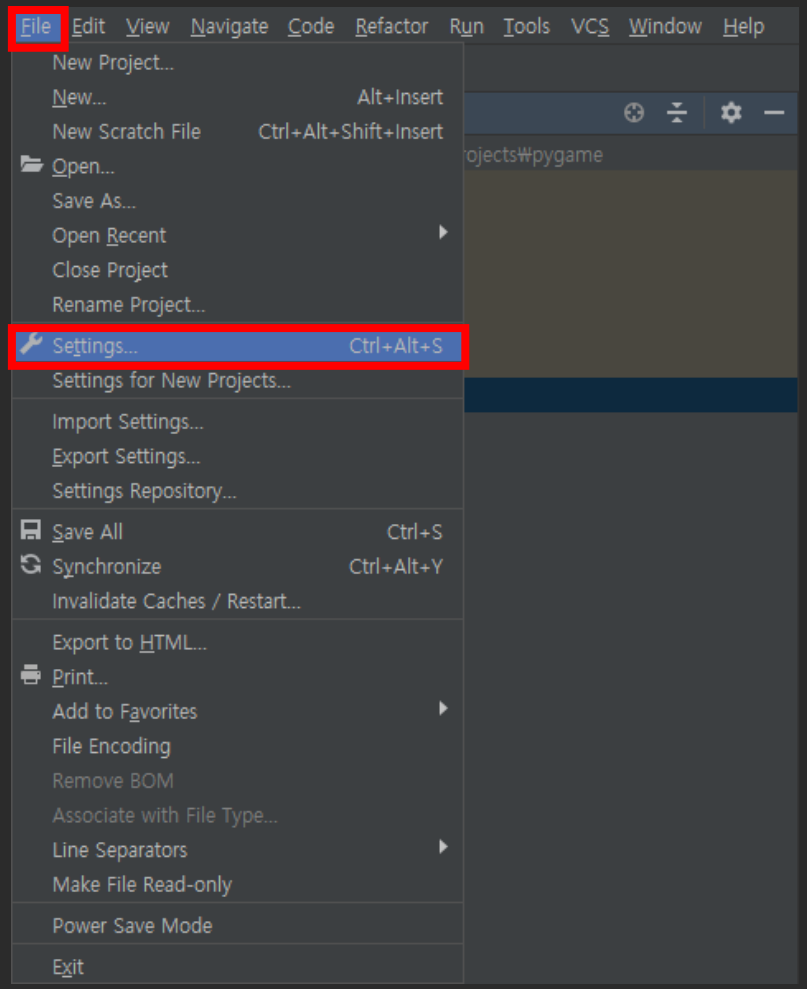
INTRO

① Pygame 라이브러리 설치

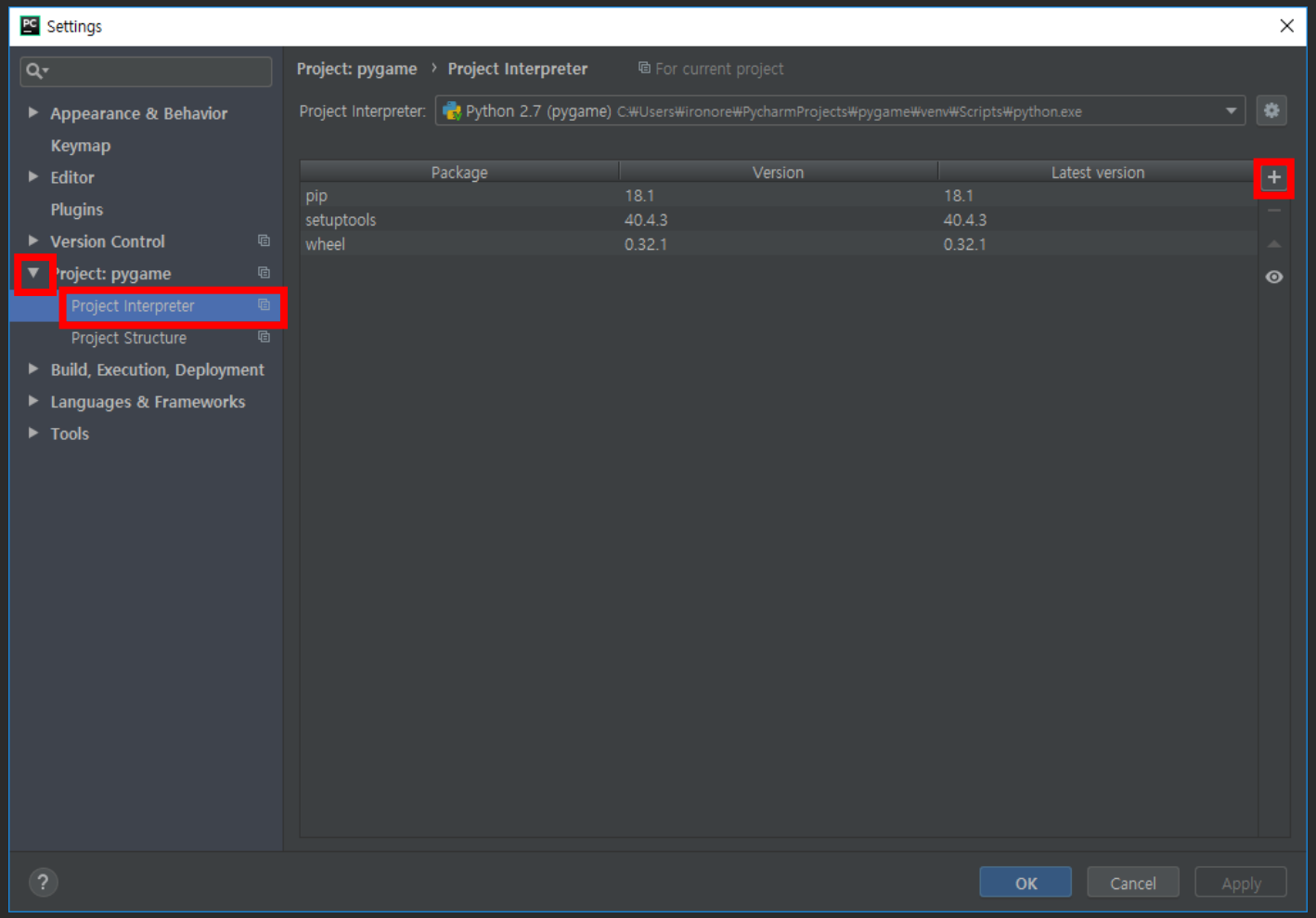
② 강의에 필요한 이미지 파일 다운로드

③ Pygame으로 Chess 구현하기 (1)

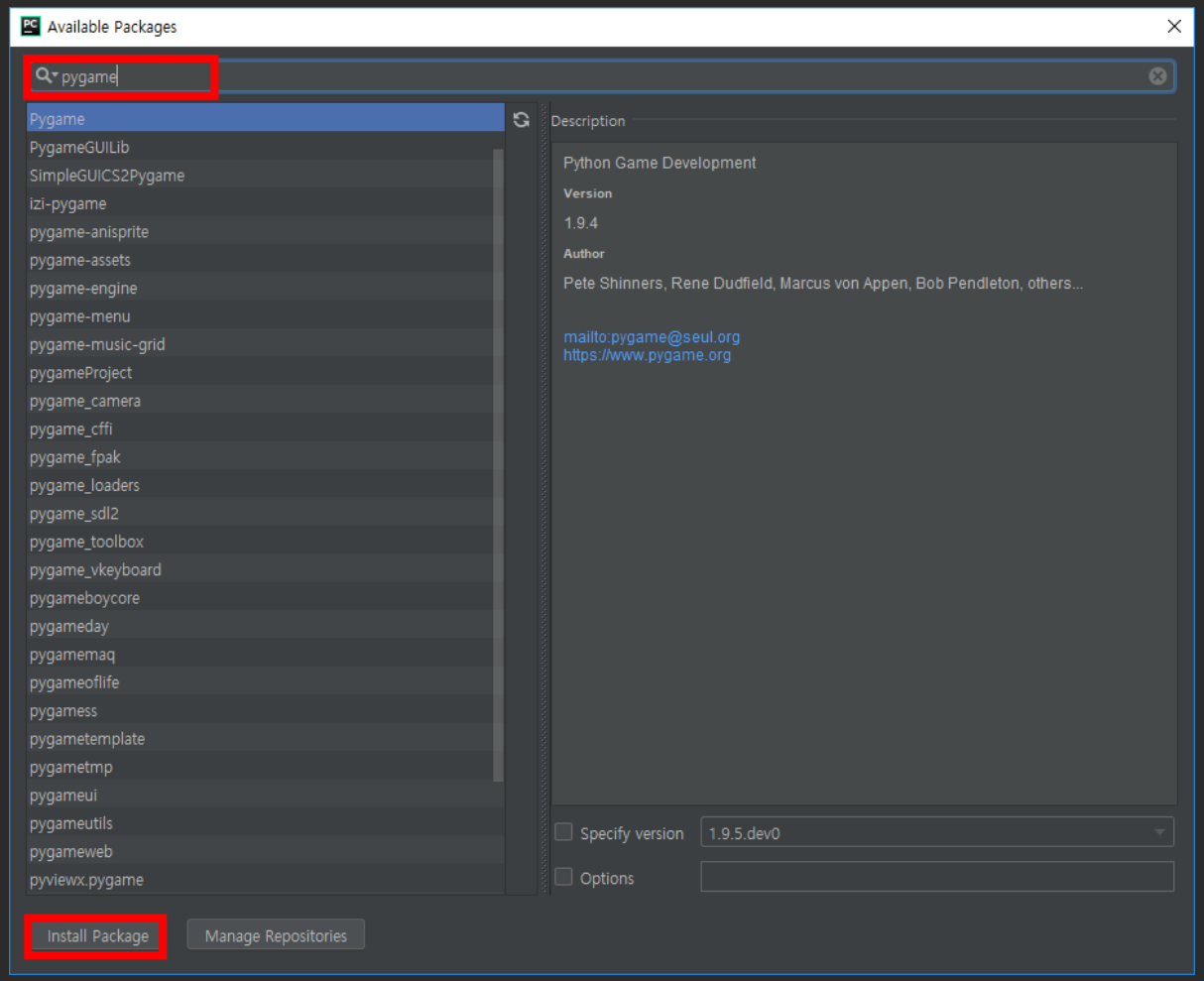
Pygame 라이브러리 설치



Pygame 라이브러리 설치

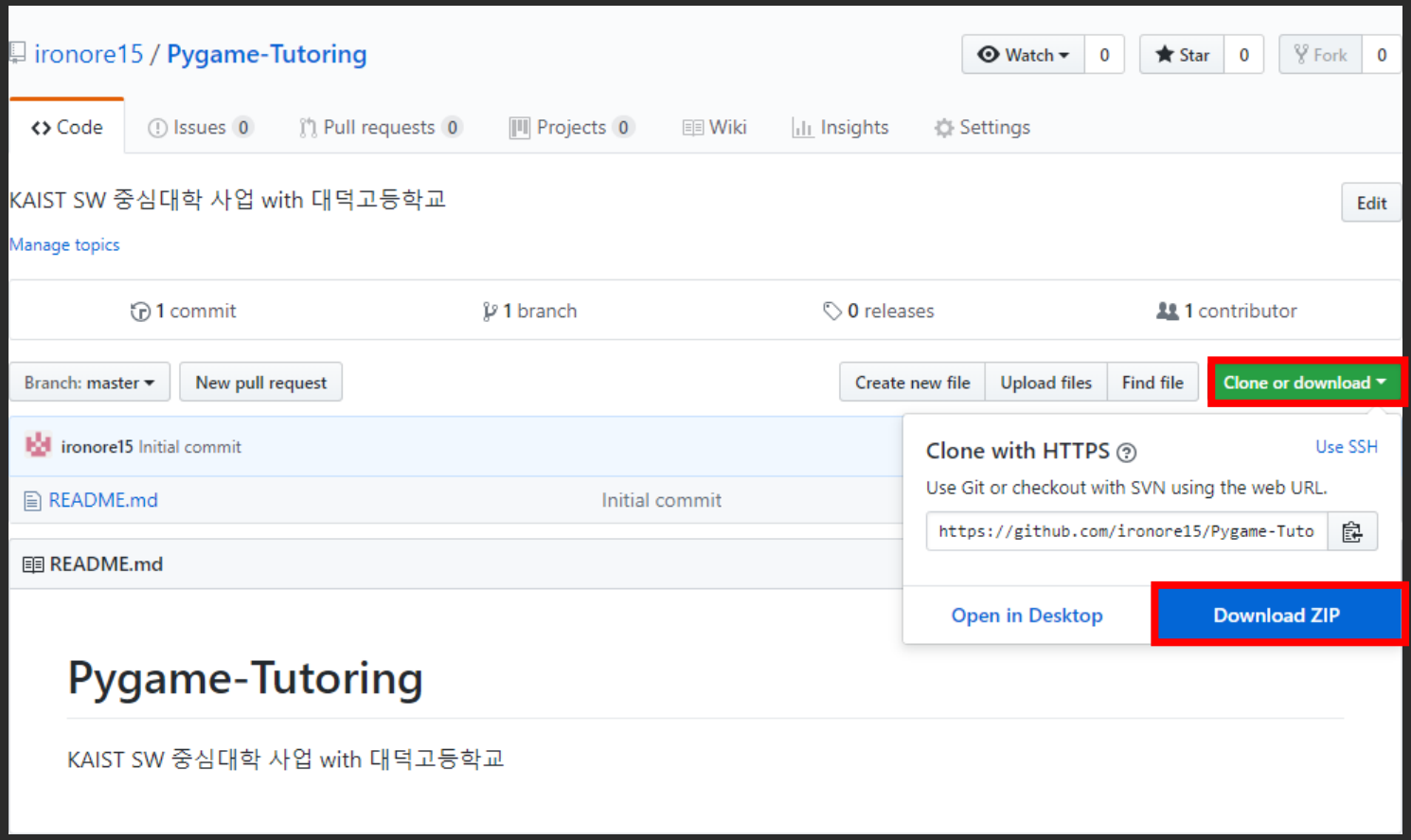


Pygame 라이브러리 설치

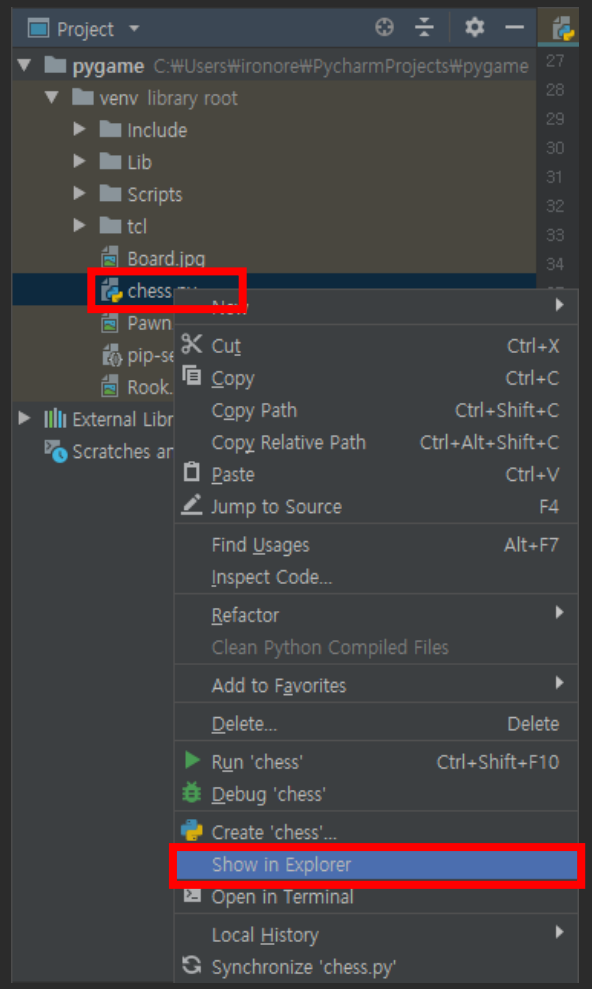


이미지 파일 다운로드

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



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'

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

```
(...)

screen = pygame.display.set_mode(size)

board = pygame.image.load("board.jpg")
screen.blit(board, (0, 0))
pygame.display.flip()

while True:
(...)


```

Resize images

```
(...)

screen = pygame.display.set_mode(size)

board = pygame.image.load("board.jpg")
board = pygame.transform.scale(board, size)
screen.blit(board, (0, 0))
pygame.display.flip()

while True:
(...)

```

Handle events

```
(...)

pawn = pygame.image.load("Pawn.png")
pawn = pygame.transform.scale(pawn, (width // 8, height // 8))

while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            exit()
        elif event.type == pygame.KEYDOWN:
            screen.blit(pawn, (0, 0))
            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

```
while True:
    for event in pygame.event.get():
        (...)
        elif event.type == pygame.MOUSEBUTTONDOWN:
            click = True
        elif event.type == pygame.MOUSEBUTTONUP:
            click = False

    if click:
        screen.blit(rook, pygame.mouse.get_pos())
        pygame.display.flip()

```

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?

```
while True:
    for event in pygame.event.get():

        (...)

        elif event.type == pygame.MOUSEBUTTONDOWN:

            (...)

            if condition:
                click = True

```

Is mouse inside rook?

```
while True:
    for event in pygame.event.get():
        (...)
        elif event.type == pygame.MOUSEBUTTONDOWN:
            (...)
            if condition:
                click = True

USE:
x, y = mouse.get_pos()
rect.left and rect.right
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

```
while True:
    for event in pygame.event.get():
    (...)
        elif event.type == pygame.KEYUP:
            if event.key == pygame.K_LEFT:
                rect = rect.move(-5, 0)
            elif event.key == pygame.K_RIGHT:
                rect = rect.move(5, 0)
            elif event.key == pygame.K_UP:
                rect = rect.move(0, -5)
            elif event.key == pygame.K_DOWN:
                rect = rect.move(0, 5)
```

Set rook always

```
while True:
    for event in pygame.event.get():
    (...)
        elif event.type == pygame.KEYUP:
            if event.key == pygame.K_LEFT:
                rect = rect.move(-5, 0)
            (...)
    if click:
        rect.center = pygame.mouse.get_pos()
    screen.blit(board, (0, 0))
    screen.blit(rook, rect)
    pygame.display.flip()
```