

```

def main():
    global FPSLOCK, DISPLAYSURF, BASICFONT, BIGFONT
    pygame.init()
    FPSLOCK = pygame.time.Clock()
    DISPLAYSURF = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT))
    BASICFONT = pygame.font.Font('freesansbold.ttf', 18)
    BIGFONT = pygame.font.Font('freesansbold.ttf', 100)
    pygame.display.set_caption('2023055650_CHOIYEJOIN')

    showTextScreen('MY TETRIS')
    while True: # game loop
        startTime = time.time() # 게임 시작할 때 startTime 초기화
        if random.randint(0, 1) == 0:
            pygame.mixer.music.load('Platform_9.mp3')
        else:
            pygame.mixer.music.load('Hover.mp3')
        pygame.mixer.music.play(-1, 0.0)
        runGame()
        pygame.mixer.music.stop()
        showTextScreen('Game Over@!')

```

pygame.display.set\_caption('2023055650\_choiyeojin') 함수를 사용하여 상태창에 학번과 이름이 나오게 수정을 했으며, showtextscreen() 함수로 시작을 할 때 my tetris가 뜨도록 수정을 했다. 그 다음 게임을 시작할 때 시간을 초기화 하기 위해 starttime= time.time()을 사용하였다. 그 다음 노래 3개 중 하나를 나오게 하기 위해 if-else 문을 사용하였다.

```

while True: # game loop
    playTime = int(time.time() - startTime) # playTime을 초 단위로 계산
    if fallingPiece == None:
        fallingPiece = nextPiece
        nextPiece = getNewPiece()
        lastFallTime = time.time() # reset lastFallTime

    if not isValidPosition(board, fallingPiece):
        return # can't fit a new piece on the board, so game over

```

게임 시간을 초단위로 계산할 수 있게 playTime = int(time.time() - startTime)을 사용하였다.

```
def showTextScreen(text):
    # This function displays large text in the
    # center of the screen until a key is pressed.
    # Draw the text drop shadow
    DISPLAYSURF.fill(BG_COLOR) # 배경색을 검정색으로 설정합니다.
    titleSurf, titleRect = makeTextObjs(text, BIGFONT, YELLOW)
    titleRect.center = (int(WINDOWWIDTH / 2), int(WINDOWHEIGHT / 2))
    DISPLAYSURF.blit(titleSurf, titleRect)
```



DISPLAYSURF.fill(BG\_COLOR)로 게임 시작화면의 배경색을 검정색이 되게 했으며, titleSurf, titleRect = makeTextObjs(text, BIGFONT, YELLOW)로 게임 시작 화면의 문구를 노란색으로 변경하였고, TEXTCOLOR 와 TEXTSHADOWCOLOR을 YELLOW로 변경하였다.

```
def drawBox(boxx, boxy, color, pixelx=None, pixely=None):
    # draw a single box (each tetromino piece has four boxes)
    # at xy coordinates on the board. Or, if pixelx & pixely
    # are specified, draw to the pixel coordinates stored in
    # pixelx & pixely (this is used for the "Next" piece).
    if color == BLANK:
        return
    if pixelx == None and pixely == None:
        pixelx, pixely = convertToPixelCoords(boxx, boxy)
    pygame.draw.rect(DISPLAYSURF, COLORS[color], (pixelx + 1, pixely + 1, BOXSIZE - 1, BOXSIZE - 1))
    pygame.draw.rect(DISPLAYSURF, LIGHTCOLORS[color], (pixelx + 1, pixely + 1, BOXSIZE - 4, BOXSIZE - 4))
```

drawBox()가 각블록을 그리는 역할을 하는데 colors와 lightcolors로 각 블록의 기본색상을 저장하고 각 블록의 하이라이트된 색상을 저장하며 고유의 색을 저장하게 한다. \

<https://github.com/duwlsxx/osw-repository>