

과제중심수업 2023028077 김영연

코드수정

1. 현재 테트리스 게임의 배경음악을 주어진 3개의 음악 중 1개가 재생되도록 수정

main함수에서 아래 코드 부분에 음악 파일이름으로 변경

```
if random.randint(0, 1) == 0:  
    pygame.mixer.music.load('Hover.mp3')  
else:  
    pygame.mixer.music.load('Hover.mp3')
```

2. 상태창 이름을 학번_이름 으로 수정

main 함수에서 아래 코드 부분의 문자 수정

```
pygame.display.set_caption('2023028077_KIMYOUNGYEON')
```

3. 게임시작화면의 문구를 MY TETRIS으로 변경

main 함수에서 아래 코드 부분의 문자 수정

```
showTextScreen('MY TETRIS')
```

4. 게임시작화면의 문구 및 배경색을 노란색으로 변경

main함수 시작되기 전 부분에서 시작화면 문구와 배경색을 노란색으로 지정

```
TEXTCOLOR = YELLOW  
TEXTSHADOWCOLOR = YELLOW
```

5. 게임 경과 시간을 초 단위로 표시 (새 게임 시작시 0으로 초기화 되어야 함)

-전역 변수로 gameTime = None 선언

-main함수 윗부분에 게임 플레이 시간을 계산하는 함수 추가

```
def calculatePlayTime():  
    global gameTime  
    if gameTime is None:  
        return 0  
    return int(time.time() - gameTime)
```

-runGame 함수에 nextPiece = getNewPiece()와 while True: 사이에 다음 코드 추가
하여 게임 시작 시간을 초기화 함

```
global gameStartTime  
gameStartTime = time.time()
```

그리고 drawStatus(score, level)와 drawNextPiece(nextPiece) 사이에 다음의 코드를
추가하여 플레이 시간을 화면에 표시함

```
playTime = calculatePlayTime()  
drawPlayTime(playTime)
```

-runGame 함수 밑에 게임 플레이 시간을 화면에 표시하는 함수 추가

```
def drawPlayTime(playTime):  
timeSurf = BASICFONT.render('Play Time: %s' % playTime, True, TEXTCOLOR)  
timeRect = timeSurf.get_rect()  
timeRect.topleft = (20, 20) # 위치를 원하는 곳으로 설정  
DISPLAYSURF.blit(timeSurf, timeRect)
```

6. 7개의 블록이 각각 고유의 색을 갖도록 코드를 수정하거나 추가

-색상 추가 및 LIGHTCOLORS 삭제

```
ORANGE = (255, 165, 0)  
CYAN = (0, 255, 255)  
PURPLE = (128, 0, 128)
```

```
COLORS = (BLUE, GREEN, RED, YELLOW, ORANGE, CYAN, PURPLE)
```

-PIECES 밑에 블록별로 색깔 지정해주는 딕셔너리 추가

```
COLORSS = {'S': BLUE, 'Z': GREEN, 'J': YELLOW, 'L': RED, 'I': PURPLE, 'O':  
ORANGE, 'T': CYAN}
```

-getNewPiece 함수에서 'color': COLORSS[shape]로 변경

-drawBox 함수에서 아래와 같이 코드 수정

```
pygame.draw.rect(DISPLAYSURF, color, (pixelx + 1, pixely + 1, BOXSIZE - 1,  
BOXSIZE - 1))  
pygame.draw.rect(DISPLAYSURF, color, (pixelx + 1, pixely + 1, BOXSIZE - 4,  
BOXSIZE - 4))
```

함수의 역할

1. main 함수

게임의 주요 실행 흐름

- pygame.init()을 호출하여 Pygame 라이브러리를 초기화
- 게임 루프에서 사용할 전역 변수와 Pygame 윈도우를 생성하고, 폰트 객체를 초기화
- showTextScreen 함수를 호출하여 게임의 시작 화면을 표시
- 무한 루프를 통해 게임을 지속적으로 실행
- pygame.mixer.music.load() 및 pygame.mixer.music.play() 함수를 사용하여 배경 음악을 재생
- runGame 함수를 호출하여 실제 게임 플레이를 수행
- pygame.mixer.music.stop() 호출하여 배경 음악 정지, showTextScreen 함수를 사용하여 "Game Over" 메시지를 표시
- 위의 과정이 게임 종료 될때까지 반복됨

2. runGame 함수

실제 게임 플레이

- 게임 보드 초기화 및 마지막으로 이동한 시간 등의 변수 설정
- 현재 떨어지고 있는 조각이 없다면, 새로운 조각을 생성하고 다음 조각을 가져옴 또한, 조각이 생성될 수 없는 경우 게임 종료
- 사용자의 입력에 따라 조각을 이동, 회전, 게임 일시 정지
- 사용자의 입력에 따라 조각을 좌우로 이동하거나 아래로 떨어뜨림
- 일정 시간이 지나면 조각을 자동으로 아래로 이동 또한 조각이 바닥에 닿았는지 확인하고, 닿았다면 보드에 추가하고 다 채워진 줄이 있는지 확인
- 게임 화면을 갱신하고, 현재 게임 상태를 화면에 표시
- 게임이 지정된 FPS로 실행될 수 있도록 프레임을 제어
- 위의 과정에 게임 종료때까지 반복됨 이를 통해 조각을 이동시키고 채워진 줄을 없애는 등의 기능 가능

3. isValidPosition

조각이 유효한 위치에 있는지 확인

- 보드 내에 조각이 존재하는지를 확인, 보드를 벗어나면 False 반환
- 조각이 다른 조각이나 보드와 충돌 확인, 충돌 발생시 False 반환
- 보드 내에 있고 다른 조각과 충돌하지 않는 경우에만 True 반환

함수의 호출 순서 및 호출 조건에 대한 설명

프로그램의 시작점

1. main

- 프로그램이 시작될 때 호출
- showTextScreen('MY TETRIS') 호출, 게임 시작 화면 표시
- 게임 루프 시작, 게임 종료까지 runGame() 반복 호출

게임 루프

1. runGame()

- 게임 보드 초기화 getBlankBoard()
- 현재 조각과 다음 조각 생성 getNewPiece() 호출
- gameStartTime 현재 시간으로 설정
- 2. 게임 메인 루프 (while True):
 - fallingPiece 없다면 새로운 조각을 nextPiece에서 가져옴, 새로운 `nextPiece` 생성
 - 조각이 보드에 유효한 위치인지 확인 isValidPosition()
 - 키보드 입력 처리 pygame.event.get()
 - 조각이 착지한 경우 보드에 추가하고, 완성된 라인을 제거 addToBoard(), removeCompleteLines()
 - 게임 상태를 화면에 그림 drawBoard(), drawStatus(), drawPlayTime(), drawNextPiece(), drawPiece()

그외 함수

1. 보드 초기화 및 조각 생성

- getBlankBoard()
호출 조건: 게임 시작 시 또는 새로운 게임이 시작될 때.
- getNewPiece()
호출 조건: 게임 시작 시, 새로운 조각이 필요할 때.

2. 게임 상태 확인 및 조작

- isOnBoard(x, y)

호출 조건: 조각의 유효성을 확인할 때 (isValidPosition()에서 호출).

-isValidPosition(board, piece, adjX=0, adjY=0)

호출 조건: 조각을 이동, 회전, 또는 새로운 조각을 보드에 추가할 때.

-isCompleteLine(board, y)

호출 조건: removeCompleteLines()에서 호출.

-removeCompleteLines(board)

호출 조건: 조각이 착지한 후 (runGame()에서 호출).

3. 화면에 그리기

-drawBoard(board), drawStatus(score, level), drawPlayTime(playTime),
drawPiece(piece, pixelX=None, pixelY=None), drawNextPiece(piece)

호출 조건: 게임 루프에서 매 프레임마다 호출 (runGame()에서).

4. 게임 종료 및 종료 조건

-terminate()

호출 조건: checkForQuit()에서 QUIT 이벤트 또는 ESC 키 입력 시.

-checkForQuit()

호출 조건: 게임 루프에서 매 프레임마다 호출 (runGame()에서).

5. 텍스트 화면 표시

-showTextScreen(text)

호출 조건: 게임 시작 시, 게임 오버 시.

6. 기타

-calculatePlayTime()

호출 조건: 게임 루프에서 플레이 시간을 표시할 때 (runGame()에서).

-calculateLevelAndFallFreq(score)

호출 조건: 점수가 변경될 때 (runGame()에서).

-makeTextObjs(text, font, color)

호출 조건: showTextScreen()에서 텍스트를 그릴 때.

GitHub Repository 주소

<https://github.com/youngHYU/osw>