

과제중심수업1_실습 보고서

2022068859 정보사회미디어학과 강다경

1. 1~6번의 과제를 코드의 어느 부분을 어떻게 수정했는지 설명

- 1) 현재 테트리스 게임의 배경음악을 주어진 3개의 음악 중 1개가 재생되도록 수정
 - `main` 함수에서 게임 시작 시 무작위로 선택된 음악을 재생하도록 수정
 - `random.choice` 함수를 사용해 3개의 음악 중 하나를 선택하여 재생하도록 수정

```
while True: # game loop
    music_choice = random.choice(['Hover.mp3', 'Our_Lives_Past.mp3', 'Platform_9.mp3']) #과제1
    pygame.mixer.music.load(music_choice)
    pygame.mixer.music.play(-1, 0.0)
    startTime = time.time()
    runGame()
    pygame.mixer.music.stop()
    showTextScreen('Game Over')
```

- 2) 상태창 이름을 학번_이름 으로 수정
 - `pygame.display.set_caption()` 함수를 통해 상태창의 이름을 "2022068859_DAKYOUNGKANG"으로 변경

```
def main():
    global FPSLOCK, DISPLAYSURF, BASICFONT, BIGFONT, startTime
    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('2022068859_DAKYOUNGKANG') #과제2
```

- 3) 게임시작화면의 문구를 MY TETRIS으로 변경
 - `showTextScreen()` 함수를 사용하여 게임 시작화면의 문구를 "MY TETRIS"로 변경

```
showTextScreen('MY TETRIS') #과제3
```

- 4) 게임시작화면의 문구 및 배경색을 노란색으로 변경
 - `TEXTCOLOR`, `TEXTSHADOWCOLOR`의 색깔을 "YELLOW"로 변경

```
BORDERCOLOR = BLUE
BGCOLOR = BLACK
TEXTCOLOR = YELLOW #과제4
TEXTSHADOWCOLOR = YELLOW
```

- 5) 게임 경과 시간을 초 단위로 표시 (새 게임 시작시 0으로 초기화 되어야 함)
- `drawStatus()` 함수에서 왼쪽 상단에 "Play Time: OO sec" 형식으로 게임 경과 시간 표시
 - `startTime` 변수는 게임이 시작될 때 초기화 됨

```
# draw the elapsed time text #과제5
elapsedTime = int(time.time() - startTime)
elapsedTimeSurf = BASICFONT.render(f'Play Time: {elapsedTime} sec', True, TEXTCOLOR)
elapsedTimeRect = elapsedTimeSurf.get_rect()
elapsedTimeRect.topleft = (10, 10) # Position in the left top corner
DISPLAYSURF.blit(elapsedTimeSurf, elapsedTimeRect)
```

- 6) 7개의 블록이 각각 고유의 색을 갖도록 코드를 수정하거나 추가
- 컬러가 (Blue, Green, Red, Yellow) 4가지밖에 없어서 (Cyan, Magenta, Orange) 3가지 컬러를 추가
 - 7개의 블록 각각에 고유의 색상 지정

```
# Assign specific colors to each shape
COLORS = { #과제6
    'S': BLUE,
    'Z': GREEN,
    'I': RED,
    'O': YELLOW,
    'J': CYAN,
    'L': MAGENTA,
    'T': ORANGE}
```

```
LIGHTCOLORS = {
    'S': LIGHTBLUE,
    'Z': LIGHTGREEN,
    'I': LIGHTRED,
    'O': LIGHTYELLOW,
    'J': LIGHTCYAN,
    'L': LIGHTMAGENTA,
    'T': LIGHTORANGE}
```

```
def getNewPiece():
    # return a random new piece in a random rotation and color
    shape = random.choice(list(PIECES.keys()))
    newPiece = {'shape': shape,
                'rotation': random.randint(0, len(PIECES[shape]) - 1),
                'x': int(BOARDWIDTH / 2) - int(TEMPLATEWIDTH / 2),
                'y': -2, # start it above the board (i.e. less than 0)
                'color': COLORS[shape]}
    return newPiece
```

2. 각 함수의 역할

- 1) `main()` :
 - 게임을 초기화하고 실행하는 함수
 - 초기화된 게임 창에 대한 설정을 진행하고, 게임 루프를 시작
 - 게임의 시작, 종료 및 재시작 처리
- 2) `runGame()` :
 - 실제 게임 루프를 실행하고, 게임 상태 관리
 - 키 입력에 따라 블록을 이동하거나 회전시키며, 라인이 완성되었는지 확인하여 점수 갱신
 - 테트리스 조각의 움직임, 회전, 점수 및 레벨 표시 등 관리
- 3) `showTextScreen()` :
 - 게임 시작화면이나 일시정지 화면과 같은 텍스트 화면을 표시
 - 화면 가운데에 큰 제목 "MY TETRIS"와 "Press a key to play! Pause key is p"와 같은 부가적 텍스트 표시
- 4) `checkForQuit()` :
 - QUIT 및 ESC 키 이벤트를 확인하여 게임을 종료하는 함수
- 5) 기타 함수들은 주요 기능을 수행하며, 게임의 동작에 필요한 블록 생성, 보드 조작 등을 담당

3. 함수의 호출 순서 및 호출 조건에 대한 설명

- 1) `main()` 함수 :
 - 먼저 ``pygame.init()``을 호출하여 게임 초기화
 - ``pygame.display.set_mode()``를 사용하여 게임 창 크기 설정하고, ``pygame.display.set_caption()``을 사용하여 창의 이름 설정한 후, ``showTextScreen()`` 함수 호출하여 게임 시작화면 표시
 - 이후 ``runGame()`` 함수를 호출하여 실제 게임 루프 진행
 - 프로그램이 실행되면 바로 호출됨
- 2) `runGame()` 함수 :
 - 게임 루프 내에서 주요 함수들을 호출하여 게임의 동작 관리
 - 키 입력에 따라 블록을 이동하거나 회전시키며, 시간에 따라 블록을 자동으로 이동
 - 이후, 각종 게임 상태를 확인하고 화면에 표시
 - ``main()`` 함수에서 호출됨. 게임이 시작되면 호출되며, 사용자가 플레이하는 동안 지속됨
- 3) `showTextScreen()` 함수 :
 - 텍스트 화면 표시를 위해 필요한 준비 작업 수행
 - 사용자가 키를 누를 때까지 기다림
 - 프로그램 시작 시 혹은 게임 종료 시 호출됨. 사용자에게 텍스트 화면 띄워주는 역할
- 4) `drawStatus()` 함수 :
 - ``runGame()`` 함수 내에서 호출됨
 - ``runGame()`` 함수가 루프를 돌며 매 프레임마다 호출됨
 - 게임 상태가 업데이트될 때마다 호출되어 새로운 스코어와 레벨 표시
 - ``runGame()`` 함수 실행 중에만 호출 가능
 - 게임 상태가 변할 때마다 호출되어야 하기 때문에, 루프의 일부분으로써 지속적으로 호출됨

4. GitHub에 결과물 업로드 한 Repository 주소 첨부

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