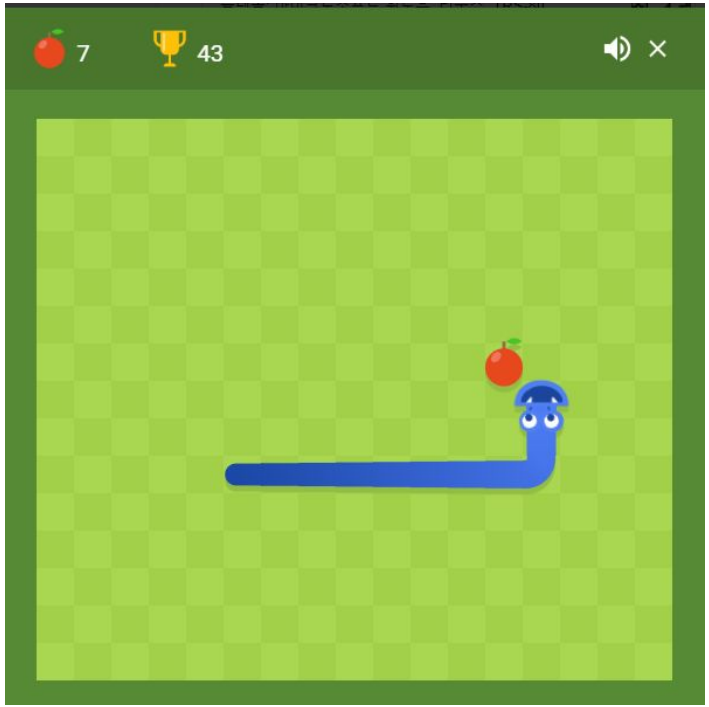
SNAKE GAME 업그레이드하기

소프트웨어 프로젝트2 AD

20191648 이 진
20191666 정주은

게임규칙설명

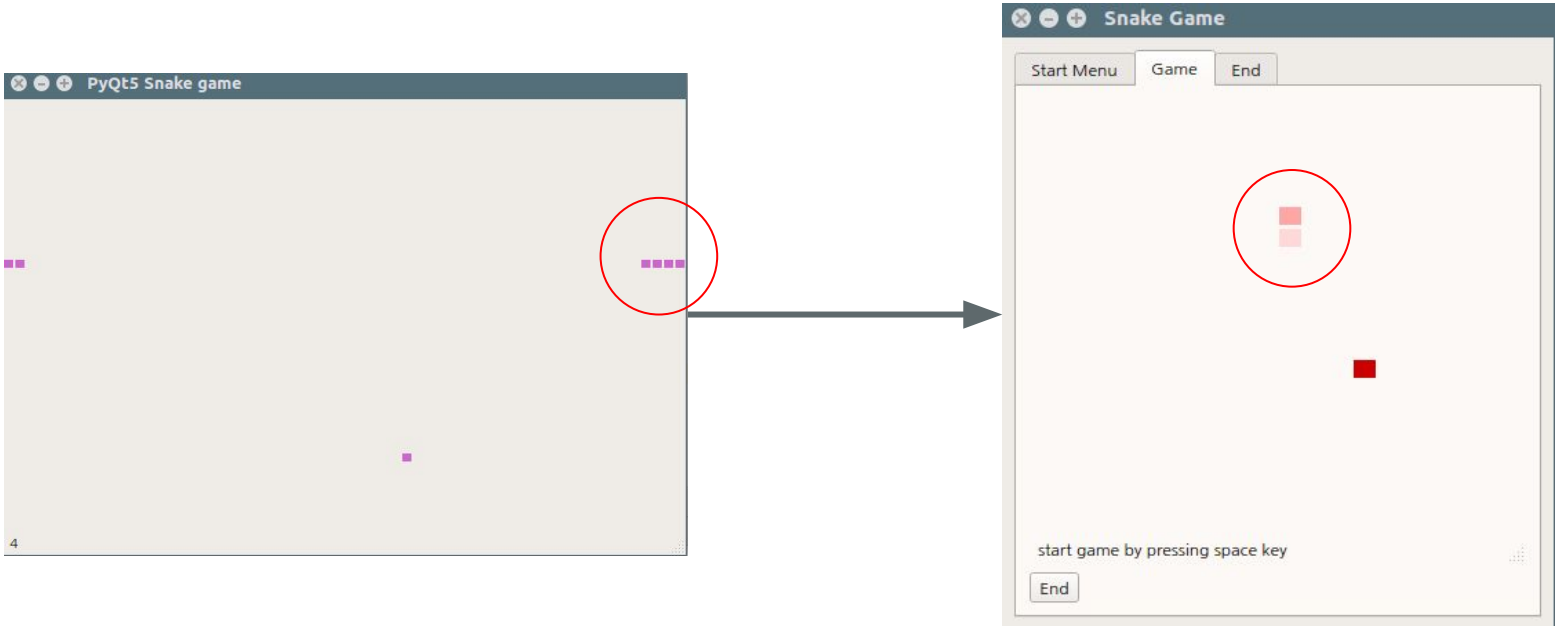


<구글의 스네이크 게임의 규칙과 동일하게 구현>

- 먹이(사과)를 랜덤하게 배치하고 방향키를 이동하여 스네이크가 먹이(사과)를 먹게 한다.
- 먹이(사과)를 먹으면, 스네이크 몸의 길이가 길어지고, 먹이 먹은 개수를 기록하는 수가 +1 증가
- 스네이크의 머리가 자신의 몸에 닿거나 벽에 부딪힐 경우, 게임 종료

SNAKE GAME

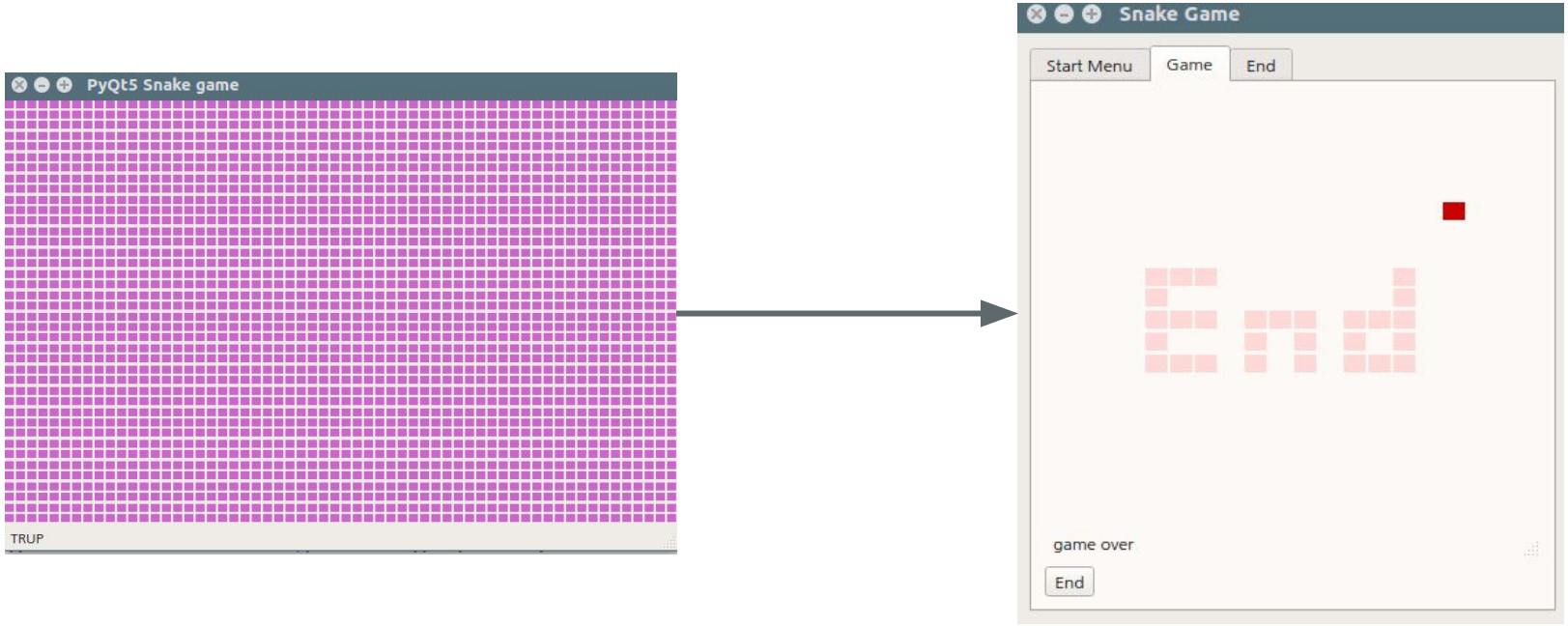
1) 기존의 버전보다 화면을 더 작고 뱀을 크게 바꾸기: 게임의 회전율을 높이기 위해!



```
self.resize(400, 400)

WIDTHINBLOCKS = 20

HEIGHTINBLOCKS = 20

def square_width(self): #window 너비의 블럭 갯수로 나눈 것의 값

    return self.contentsRect().width() / Board.WIDTHINBLOCKS

def square_height(self):

    return self.contentsRect().height() / Board.HEIGHTINBLOCKS
```

SNAKE GAME

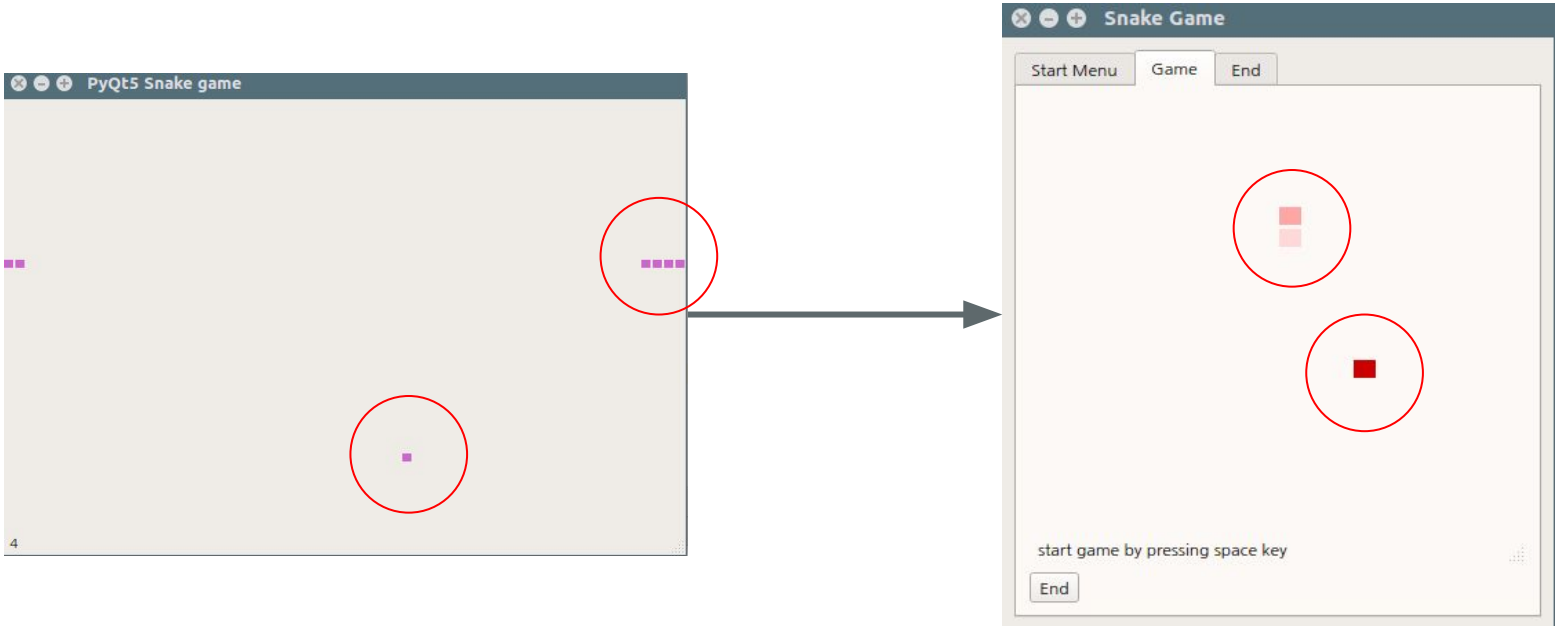
2) 게임오버화면 변경: End 띄우기



```
self.snake = [  # End 수놓기
[4, 8], [4, 9], [4, 10], [4, 11], [4, 12],
   [5, 8], [5, 10], [5, 12],
   [6, 8], [6, 10], [6, 12],
   [8, 10], [8, 11], [8, 12],
   [9, 10],
   [10, 10], [10, 11], [10, 12],
   [12, 10], [12, 11], [12, 12],
   [13, 10], [13, 12],
   [14, 8], [14, 9], [14, 10], [14, 11], [14, 12], [14, 13]
]
```

SNAKE GAME

3) 기존 버전의 뱀과 먹이 색깔 바꾸기



```
def paintEvent(self, event):
    painter = QPainter(self)
    rect = self.contentsRect()
    boardtop = rect.bottom() - Board.HEIGHTINBLOCKS * self.square_height()

    #뱀머리 색만 다르게
    self.draw_square(painter, rect.left() + self.snake[0][0] * self.square_width(),
                     boardtop + self.snake[0][1] * self.square_height(), QColor(0xFFA7A7))

    for pos in range(1, len(self.snake)):
        self.draw_square(painter, rect.left() + self.snake[pos][0] * self.square_width(),
                         boardtop + self.snake[pos][1] * self.square_height(), QColor(0xFFD8D8))
    for pos in self.food:
        self.draw_square(painter, rect.left() + pos[0] * self.square_width(),
                         boardtop + pos[1] * self.square_height(), QColor(0xCC0000))

def draw_square(self, painter, x, y, color):
    painter.fillRect(x + 1, y + 1, self.square_width() - 2,
                     self.square_height() - 2, color)
```

SNAKE GAME

4) 음악 추가하기 : BGM, Game Over할 때

-배경음악(게임 시작부터 종료 전까지 - 캐롤)

```
pygame.mixer.init()
pygame.mixer.music.load("Jingle_Bells_Instrumental_Jazz(wav).wav")
if self.timer:
    pygame.mixer.music.play(0)
```

-게임 종료 후, 효과음(아기 울음소리)

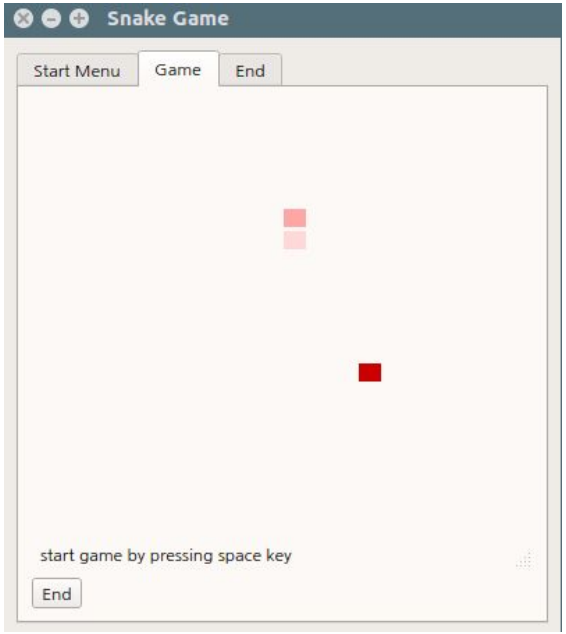
```
pygame.mixer.music.stop()
pygame.mixer.music.load("Babies_Cry(wav).wav")
pygame.mixer.music.play(27)
```

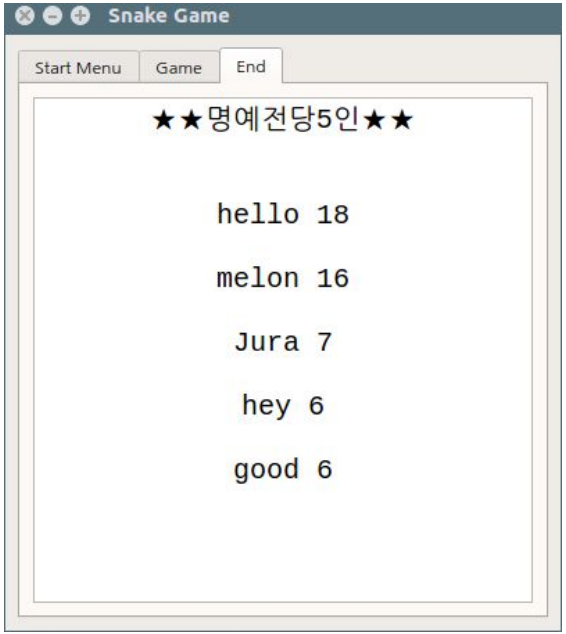
SNAKE GAME

5) 레이아웃 구성: 시작, 게임 실행, 엔딩 화면 표시하기

SNAKE GAME

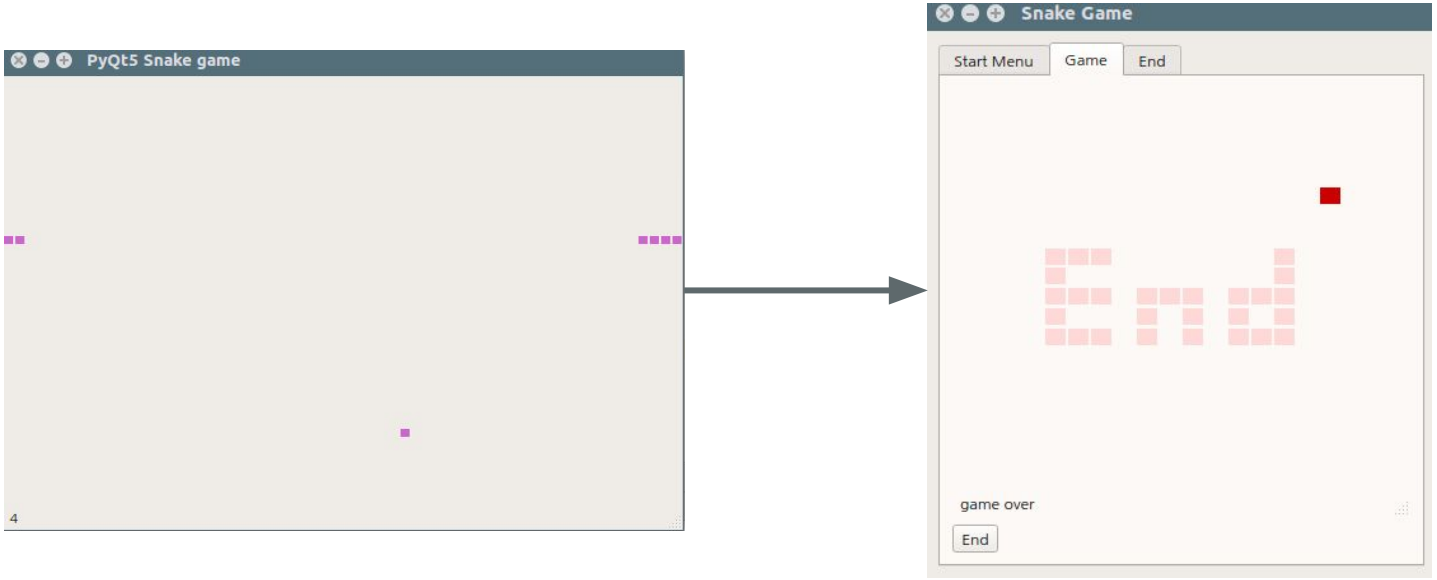
5) 레이아웃 구성: 시작, 게임 실행, 엔딩 화면 표시하기





SNAKE GAME

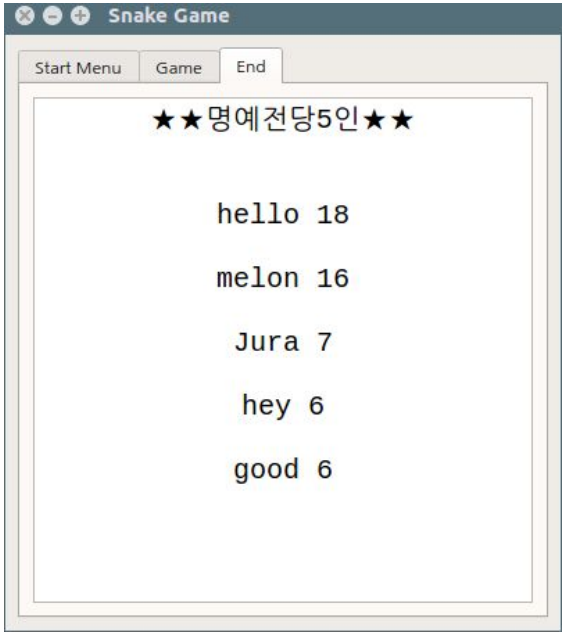
6) 벽 구현: 벽에 부딪히면 Game Over



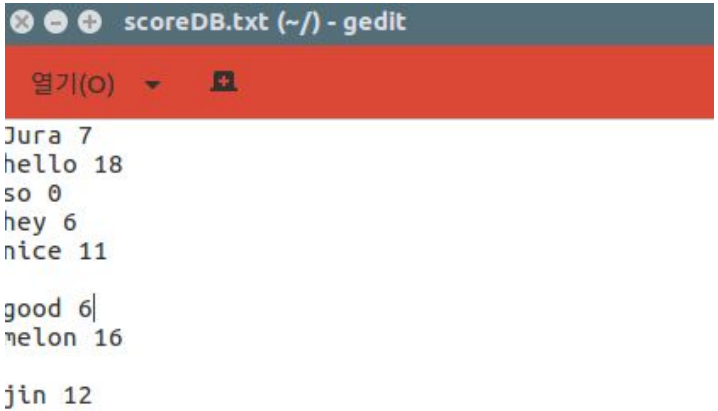
```
def move_snake(self):

    if self.direction == 1: #왼쪽
        self.current_x_head, self.current_y_head = self.current_x_head - 1, self.current_y_head
        if self.current_x_head < 0: #벽
            self.game_over()

    if self.direction == 2: #오른쪽
        self.current_x_head, self.current_y_head = self.current_x_head + 1, self.current_y_head
        if self.current_x_head == Board.WIDTHINBLOCKS:
            self.game_over()

    if self.direction == 3: #아래
        self.current_x_head, self.current_y_head = self.current_x_head, self.current_y_head + 1
        if self.current_y_head == Board.HEIGHTINBLOCKS:
            self.game_over()

    if self.direction == 4: #위쪽
        self.current_x_head, self.current_y_head = self.current_x_head, self.current_y_head - 1
        if self.current_y_head < 0:
            self.game_over()
```

SNAKE GAME

7) 다른 플레이어와의 기록 비교, 명예 전당 보여주기



```
def on_click_select_tab3(self):
    out = open("/home/jje/scoreDB.txt", "a")
    out.write("\n"+self.name+" "+str(Board.SCORE))
    out.close()
    # read scoreDB
    self.scoreBoard.setText("★★명예전당5인★★\n")
    self.scoreBoard.setAlignment(Qt.AlignCenter)

    file = open('/home/jje/scoreDB.txt', "r")
    medi = []
    for line in file:
        medi += line.split()
    print(medi)
    winner = {}

    for i in range(0, len(medi) - 1, 2):
        winner[medi[i]] = medi[i + 1]

    fiveWinner = collections.Counter(winner).most_common(5)
    fiveWinner.sort(key=lambda x: eval(x[1]), reverse=True)

    for win in fiveWinner:
        self.scoreBoard.append("\n"+win[0] + " " + win[1])
        self.scoreBoard.setAlignment(Qt.AlignCenter)

    return self.tabs.setCurrentIndex(2)
```



SNAKE GAME

8) 난이도 조정: 획득하는 먹이 개수 증가할수록 속도 빨라지도록

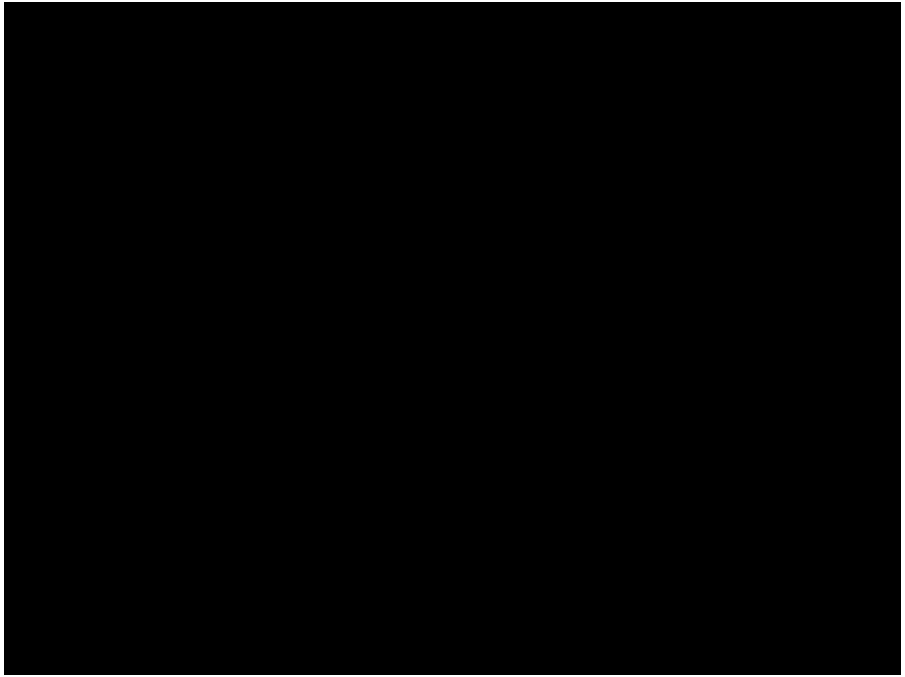
```
class Board(QFrame):
   SPEED = 150

self.timer = QBasicTimer()

def setSpeed(self, speed):
   Board.SPEED = speed

def raiseLevel(self):
   if self.score <= 2:
      self.level = 1
      self.setSpeed(150)
      self.timer.start(Board.SPEED, self)
   elif lenself.score <= 4:
      self.level = 2
      self.setSpeed(130)
      self.timer.start(Board.SPEED, self)
   elif self.score <= 6:
      self.level = 3
      self.setSpeed(100)
      self.timer.start(Board.SPEED, self)
   else:
      self.level = 4
      self.setSpeed(70)
      self.timer.start(Board.SPEED, self)
```

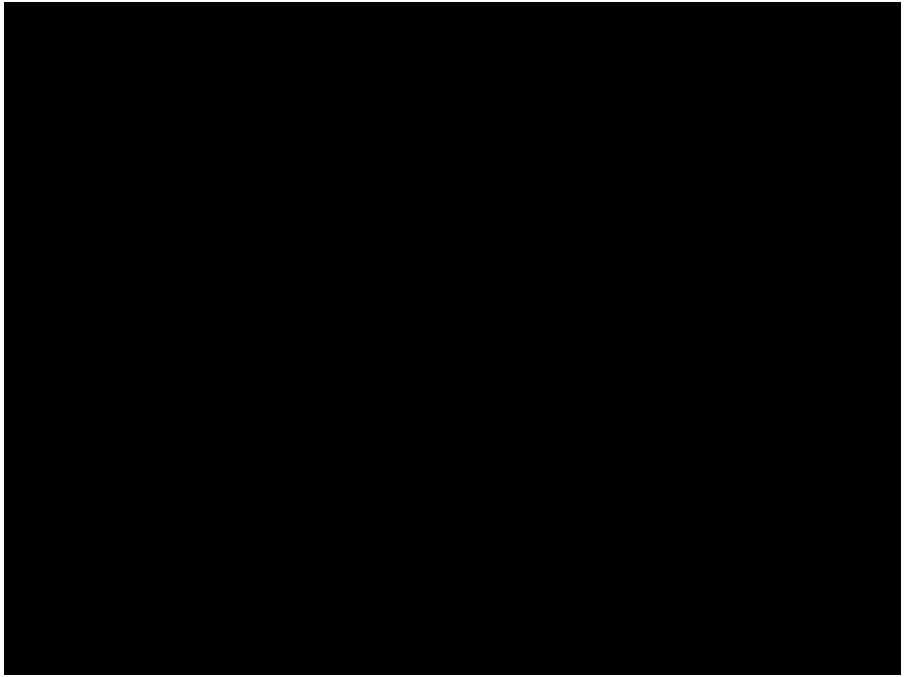
SNAKE GAME

9) 플레이어가 space를 눌렀을 때, snake가 움직이며 게임 시작

```
self.bgame = False #게임의 진행상태


def start(self):
    if not self.bgame:
        self.msg2statusbar.emit("start game by pressing space key")

def keydown(self, key):
    if (key == Qt.Key_Space):
        self.bgame = True
        self.msg2statusbar.emit("Score: " + str(len(self.snake) - 2) + " / level: " + str(self.level))  # 메세지 방출
        self.timer.start(Board.SPEED, self)
```

SNAKE GAME





감사합니다.