



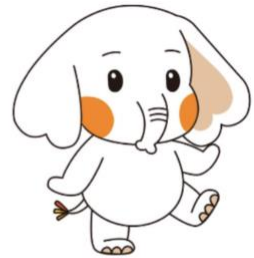
회의록 - OSSProj-12

- 12주차 중간보고서

회의일시	2022. 11. 05(토)	작성자	김규태
참석자	곽호은, 김규태, 민예지, 신승원		
회의장소	대면(18일), Webex(20일)		

1. 프로젝트 소개

프로젝트 소개



"Platformer 게임 개선"
아코를 제한시간 안에 탈출시켜라!

발판을 뛰어 넘으며 탈출구를 통해 맵을 통과하는 Platformer pygame을 기반으로 하여, 기존코드에서 요소를 변형하고 게임 진행을 다채롭게 할 수 있는 요소들을 추가하는 것을 목표로 함

2. 프로젝트 회의 안건

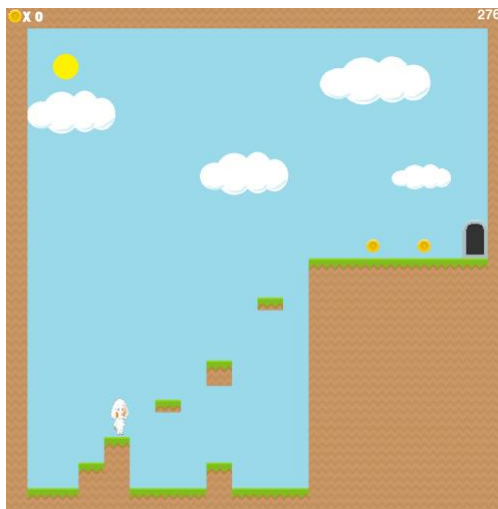
회의안건	<ul style="list-style-type: none">• DB 플로우 구체화 및 변경• 게임 제한시간을 설정하기 위한 타이머 기능 추가• 창화면 Fullsize• 클린 코드를 위한 기능별 함수화
------	---

3. 회의 내용

1. 게임 흐름 구체화 및 변경

- Sqlite 사용 (파일구조의 데이터베이스 -> 각자의 컴퓨터마다 db 파일 생성)
- 우팀의 오픈소스 코드를 클론받아 게임을 실행할 때마다 정보저장이 가능한 db파일이 생성
→ 게임을 실행시킬 경우 db 파일을 생성시키도록 코드를 구현하여 각자의 컴퓨터마다 개인화된 정보 저장 공간이 생김
- 최고 기록 점수 > 현재 기록 점수 (최고기록을 갱신하지 못했을 때)
restart & home 버튼과 함께 " 수고하셨습니다. 기록을 갱신하지 못하였습니다 " 문구 출력
- 최고 기록 점수 < 현재 기록 점수 (최고기록을 갱신 하였을 때)
restart & home 버튼과 함께 " 축하합니다. 최고 기록 갱신에 성공하였습니다" 문구 출력

2. 게임 제한시간을 설정하기 위한 타이머 기능 추가



```
world.draw()

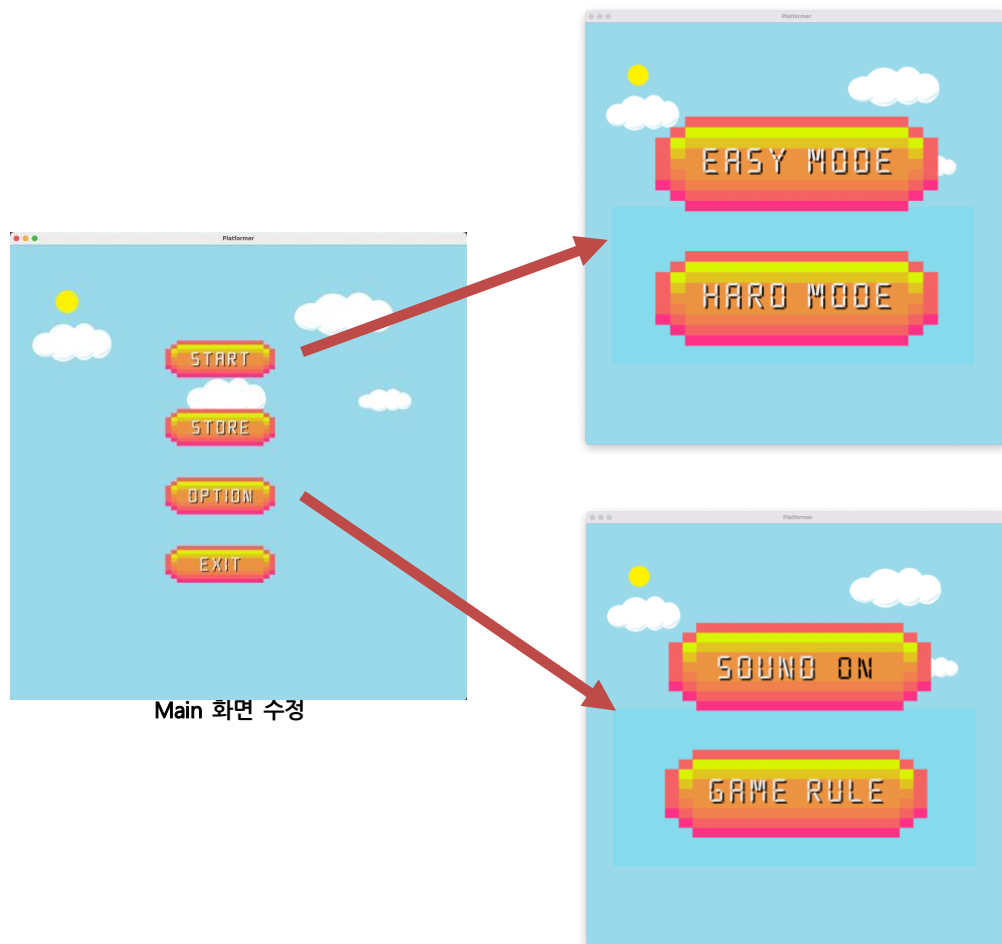
if game_over == 0:

    running=True

    # 경과 시간 계산
    elapsed_time = (pygame.time.get_ticks() - start_ticks) / 1000# 타이머 시간을 1000으로 나누어 초단위로 표시 (
    game_font = pygame.font.Font(None, 40)
    timer = game_font.render(str(int(total_time - elapsed_time)), True, (255,255,255)) # 타이머 위치 지정
    screen.blit(timer, (930,10)) # 타이머 위치 지정
    if total_time - elapsed_time <= 0:
        print("TIME OUT") # 만약 시간이 0 이하이면 게임 종료
        running = False
    pygame.display.update()
```

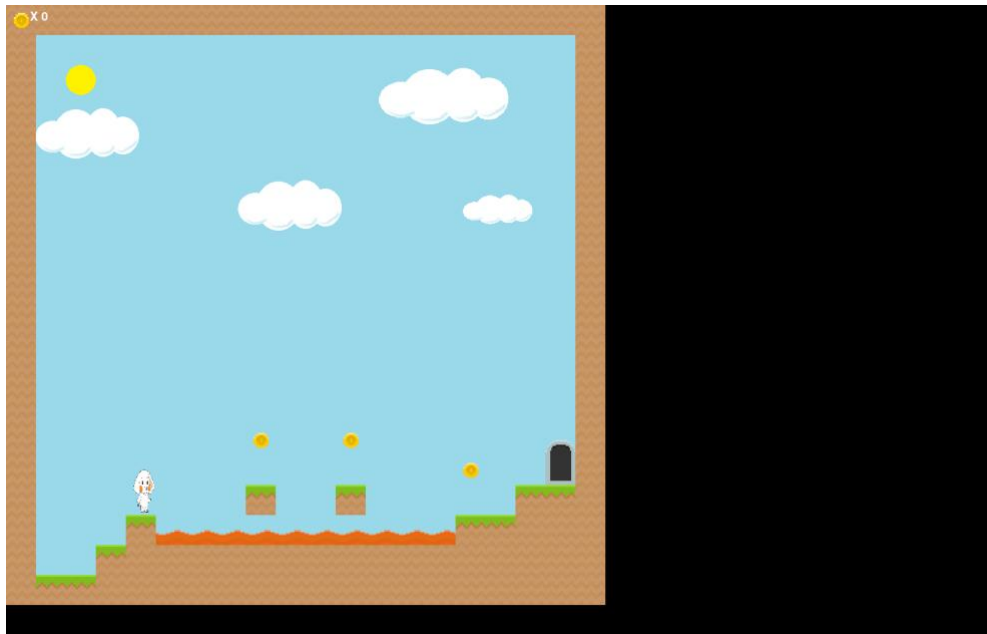
- 타이머 시간을 1000으로 나누어 초단위로 표시 함 (default : ms 단위)
- 게임 진행중 설정 된 타이머를 우측상단에 가시화 시킴
- 시간이 0에 이하로 떨어지는 경우 게임 종료

3. 화면 인터페이스 수정 및 추가 구현



- Main 화면에서 Score 버튼을 -> Store 버튼으로 수정
- Start 버튼을 눌렀을때 생성되는 화면 구성
- Option 버튼을 눌렀을때 생성되는 화면 구성

4. 창화면 Fullsize



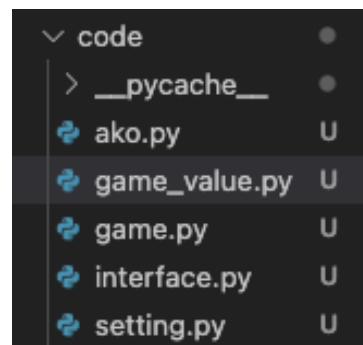
- 기존에는 width 및 height 변수로 크기가 지정되어 각각의 컴퓨터 환경설정의 해상도에 따라 창이 잘리거나 크기 조절이 어려웠음
- 아예 통일된 화면을 주고자, 조건없이 전체화면이 될 수 있도록 pygame.FULLSCREEN 플래그를 전달했음

```
screen_width = 1000
screen_height = 1000

screen = pygame.display.set_mode((screen_width, screen_height), pygame.FULLSCREEN)
pygame.display.set_caption('Platformer')
```

5. 클린 코드를 위한 기능별 함수화
- 변경 전 (좌) / 변경 후 (우)

main.py M



- 클린 코드와 기능별로 코드를 분리하기 위해 main에 모여있던 코드들을 각각의 파이썬 파일로 생성
- 팀원들과 협업을 용이하게 하기 위함

6. 13주차 향후 개발 계획

- 상점 인터페이스 구현 및 DB 연동
- Option 기능 구현
- 작업해뒀던 level_data 활용하여 Easy mode & Hard mode 구분하여 플레이

	내용	진행일정
결정사항	Webex 중간회의 진행 - 현황공유	11월 23일
	대면회의 진행 - 문제점 공유 및 해결	11월 26일