

설계-테트리스

2019년 10월 11일 금요일 오후 7:15



- <테트리스의 특징>
1. 블록은 위에서 아래로 떨어진다
 2. 각 블록은 다른 색깔을 가진다
 3. 라인이 완성되면 라인이 터진다
 4. 라인이 터지면 위에 있던 블록들이 아래로 떨어진다
 5. 충돌 범위는 블록과 가로 라인을 기준으로 한다
 6. 최대 4줄이 터진다

발견한 오류

- 제일 우측 하단에 블록을 놓으면 아래로 들어감

- 조작방법->
- Up arrow -> 회전
 - Left/right arrow -> 이동
 - Down arrow -> 가속

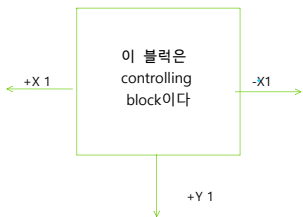
<Script>

- Block Movement :
블록의 이동과 관련된 것들. 구현했습니다. 좌우 하강 움직임, 회전
- Block Collision
prefab의 각 블록에 부착시켜 상호작용이 일어날 수 있게 했습니다.
tag랑 좌표를 이용해서 결과에 따라 이벤트를 호출합니다.
- Block Signal Reciver
현재 컨트롤이 되고 있는 블록의 BlockCollision에서 이벤트 호출 신호를 받아서 한번 필터링 했습니다. 여러 블록에서 같은 이유로 이벤트를 호출하는걸 방지할 목적으로 만들었습니다.
- Controller
컨트롤을 담당하게 만들었습니다. 블록을 Spawner에서 호출해 만들어서 사용합니다. 기본 1초당 1번 움직이게 만들었고 레벨이 올라가면 행동의 interval을 줄 일려 했습니다.

<Script>

- EventManger
델리게이트를 이용하여 이벤트 처리를 하는 패턴을 적용시켰습니다. 디서너리를 이용해서 이벤트 list에 이벤트들을 추가하여 이벤트가 호출 될 때 list에 있는 모든 이벤트가 실행되게 했습니다.
- Line Spector
Static int floor를 이용해서, 자신이 담당하는 층을 배정해줍니다. 배정된 층에 대해서 Block Constructor의 결과에 따라 행동합니다.같은 층의 블록을 List<GameObejct>Line으로 받아옵니다. 라인을 터트리는 이벤트가 실행되면 터트려 주고 윗층의 line spector에서 블록들을 받아옵니다.
- Block Constructor
10개의 lineSpector list를 가지고 있습니다. 블록을 스테이지에 놓을 때, 가지고 있는 list의 라인을 검사합니다. 검사하면서 라인을 터트릴지 판단합니다. 라인을 터트렸다면 그 위로 있는 블록들을 아래칸으로 당겨줍니다.

<블록에 대한 충돌 방향>
From = Controlling Block
To = Collider other



<충돌처리>
Vector3.Distance(form, to) :=From - To
X축 방향으로 닿았다면 그 방향으로 움직임을 제한
Y축 방향이라면 release Block

<이벤트 처리 방식>

- Delegate로 참여 조건을 만들어줍니다. (이벤트 이름)
- Dic으로 event list와 해당 이벤트에 참가하게 되는 스크립트(component)의 함수를 추가시켜줍니다.
Dic<event list, delegate type list>
- Add event를 통해서 dic의 이벤트에 참가하게 될 스크립트를 받아줍니다
함수는 list형태로 만들어서 추가해줍니다.
- Notify event로 해당 이벤트를 활성화 시킵니다.
- 이 이벤트에 참가하기로한 컴포넌트 리스트를 순회하면서 실행시켜줍니다

<블록이 사이드로 닿았을 때>

```
BLOCK_NOT_CONTACT_LEFT,  
BLOCK_NOT_CONTACT_RIGHT,  
BLOCK_CONTACT_LEFT,  
BLOCK_CONTACT_RIGHT,
```

<블록이 만들어지고 bottom으로 닿고 터지는 과정>

```
//블록이 스폰됨  
BLOCK_SPAWNED,  
//블록이 닿았다고 판정했을 때 처리하는 과정  
BLOCK_CONSTRUCTING,  
//판정이 끝나고 난 후 처리  
BLOCK_CONSTRUCT_FINISH,  
//그 줄에서 작업한 것이 완료됐다  
LINE_WORK_COMPLETE,  
//줄이 터졌다. 위에 있는 것들은 내려와야 된다  
LINE_POPED,  
//게임오버  
GAME_OVER
```

<블록이 사이드로 닿았을 때>

블록 측면이 무언가에 닿았다. 떨어졌다.

| 이벤트 종류 | 추가 | 호출 시점 | 설명 |
|-------------------------|--|---|---|
| BLOCK_NOT_CONTACT_LEFT | Controller.cs -> EventManger.Instance.AddEvent(EventType.BLOCK_NOT_CONTACT_LEFT, allowToLeft); | BlockCollision->BlockSignal->OnTriggerExit(Collider other) | 블록의 좌측면에 아무것도 닿지 않는 상태 왼쪽 키입력을 감근 것을 풀어준다 |
| BLOCK_NOT_CONTACT_RIGHT | Controller.cs -> EventManger.Instance.AddEvent(EventType.BLOCK_NOT_CONTACT_RIGHT, allowToRight); | BlockCollision->BlockSignal->OnTriggerExit(Collider other) | 블록의 우측면에 아무것도 닿지 않는 상태 오른쪽 키입력을 풀어준다 |
| BLOCK_CONTACT_LEFT | Controller.cs -> EventManger.Instance.AddEvent(EventType.BLOCK_CONTACT_LEFT, notAllowToLeft); | BlockCollision->BlockSignal->OnTriggerEnter(Collider other) | 블록의 우측면에 무언가 닿은 상태 왼쪽 키 입력을 감귀 못 가게 한다 |
| BLOCK_CONTACT_RIGHT | Controller.cs -> EventManger.Instance.AddEvent(EventType.BLOCK_CONTACT_RIGHT, notAllpwToRight); | BlockCollision->BlockSignal->OnTriggerEnter(Collider other) | 블록의 우측면에 무언가 닿은 상태 우측 키를 감귀서 못 가게 한다 |

<블록이 만들어지고 bottom으로 닿고 터지는 과정>

| Event type | add | notify | 설명 |
|-------------------------|--|--|--|
| BLOCK_SPAWNED | | | 블럭이 생성될 때 할 일 |
| BLOCK_CONSTRUCTING, | LineSpector->receiveBlocks() | BlockCollision-> BlockSignal -> OnTriggerEnter | 블럭의 하단이 gameObject와 닿았을 때 호출
LineSpector로 controlling object의 정보를 가져와서 저장을 한다 |
| BLOCK_CONSTRUCT_FINISH, | BlockConstructor->CheckLine() | Controller->ReleaseControl | Controlling object를 release하기 전에 할 작업들
완성된 라인이 있는지 검사를 한다
BlockConstructor의 CheckLine에서 검사 후
조건이 충족되면 line을 터트린다
Line_Poped 이벤트도 호출한다
마무리로 LineSpectorer들을 정리해준다 |
| LINE_Work_COMPLETE | Controller->ReleaseControl | blockCollision-> BlockSignal->
OnTriggerEnter | LineSpector에서 해야될 작업이 끝났다.
Controlling Object를 null로 바꿔주기 전에
Construcut Finish 이벤트를 호출한다 |
| LINE_POPEd | GameManager->addScore
GameManger->addExp
GameManger->addPopCount | BlockConstructor-> BlockSignal->
CheckLine() | Line이 pop된 직후 호출
Game manager의 정보를 업데이트한다 |

| evnetType | add | notify | |
|-----------|--|--|--------------------|
| Game Over | Controller->TurnOffController
Spawner->turnOffSpawn
GameManager->Show Result | blockCollision-> OnTriggerEnter
tag==DeadLine | 게임이 종료되고 결과창을 보여준다 |

개선해야 했던 점

- Line 검사과정에서 최적화
- Game manager 완성
- Block이 wall과 충돌할 경우 signal reciver로 신호를 보내면 안 됐다
- List끼리 값을 복사해주는 것에서 나는 오류
- Private로 처음 만들었다가 public으로 함수를 바꿨을 때, 이름을 고치지 않은 것
- 충돌처리 오류