Week4

픽셀 퍼펙트

B17 타일맵으로 플랫폼 만들기

1. 타일팔레트

Tile palette타일을 사용하기 위해 모아둔 프리펩

Windows->2d->tile palette -> new -> sprites들 넣기

2. 타일맵

Hierachy - 2d - tilemap : 타일을 일정하게 깔아두는 컴포넌트

3. 물리 설정

Tilemap Collider 2d: 타일맵에 맞춰 생성되는 콜라이더

카메라가 플레이어 따라가게 하기 - 플레이어에 카메라 넣기

물리모양은 sprite editor에서 편집 – custom physics shpae

타일 팔레트 먼저 삭제후 물리 모양편집이 훨씬 안전

B18. 몬스터 AI 구현하기

1. 준비하기

캐릭터, 애니메이션

2. 기본이동

Rigidbody 사용

3. 행동설정

행동지표를 바꿔주는 함수 만들어주기

Random : 랜덤 수를 생성하는 로직 관련 클래스

Range(): 최소~최대 범위의 램덤 수 생성

재귀함수 : 자신을 스스로 호출하는 함수 - 딜레이 없이 사용하면 매우 위험!

Invoke(): 주어진 시간이 지난 뒤, 지정된 함수를 실행

Cancellnvoke() - 중단

4. 지능 높이기

이동경로상태 예측을 위해 앞을 체크해야함

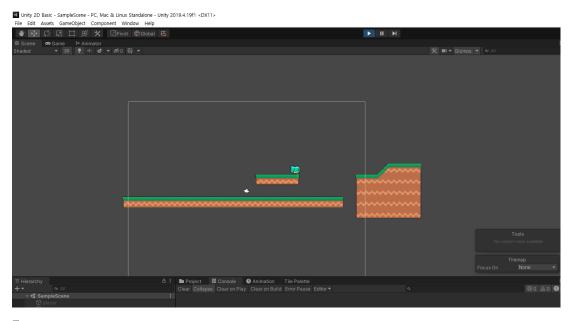
- → Ray 사용
- → 폭이 넓어서 nextMove*0.2f 해줌

5. 애니메이션

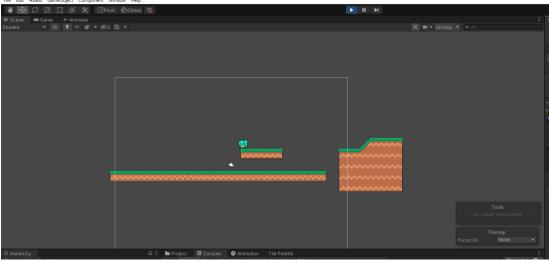
Int Walkspeed -> equal, nonequel, greater, less

앞뒤반전은 spite rendere에서!

재귀는 가장 아래쓰는 것이 일반적



Unity 2D Basic - SampleScene - PC, Mac & Linux Standalone - Unity 2019.4.19f1 <DX11>
File Edit Assets GameObject Component Window Help



B19 플레이어 피격 이벤트 구현하기

1. 함정맵 추가

타일맵 하나 추가

레이어와 태그 바꾸기

2. 레이어 설정

Edit – Project settings – physics 2d – layer enemy들끼리x(enemy 앞으로(z축변경))

3. 몬스터와의 충돌 이벤트

무적효과를 위해 적 레이어와 충돌 해제

무적상태 -> 투명, 레이어를 변경

4. 무적 시간 해제

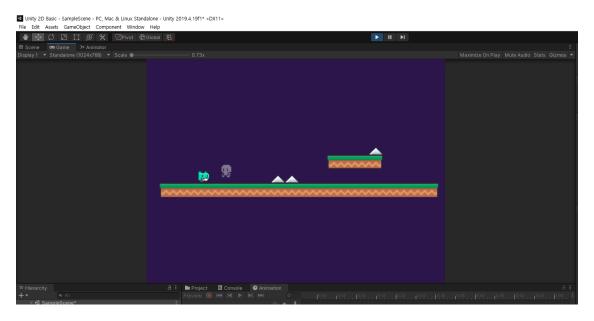
함수 만들어서 투명 없애고, 레이어 복귀 이때도 Invoke 사용

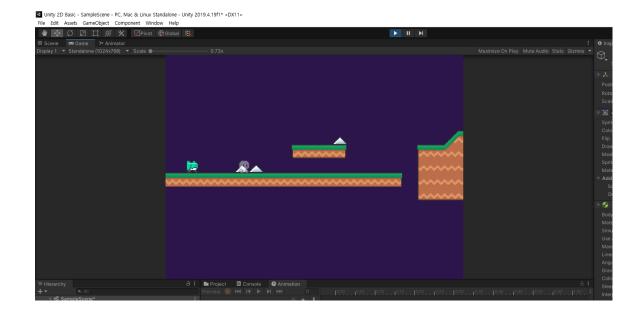
5. 애니메이션

Damaged를 만든다.

Trigger 하나 생성

Anystate -> exit : 상황 상관x 실행 후 복귀





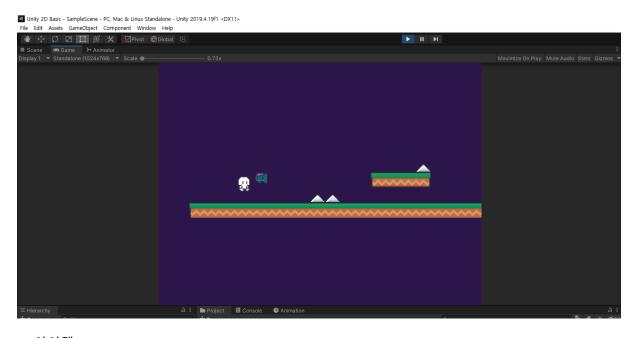
BE2: 2D 플랫포머 완성하기

1. 플레이어 이동 로직 수정

GetButtonDown은 겹치는 구간 발생 -> GetButton

2. 몬스터 밟아서 잡기

몬스터보다 위 + 아래로 낙하중 = 밟음



3. 아이템

콜라이더 + 스프라이트 + 애네메이션 작업으로 아이템 구현

Tag item으로

Item은 is trigger 체크를 해주자!

4. 결승점

Tag finish

다음 스테이지 – GameManager : create empty, new script

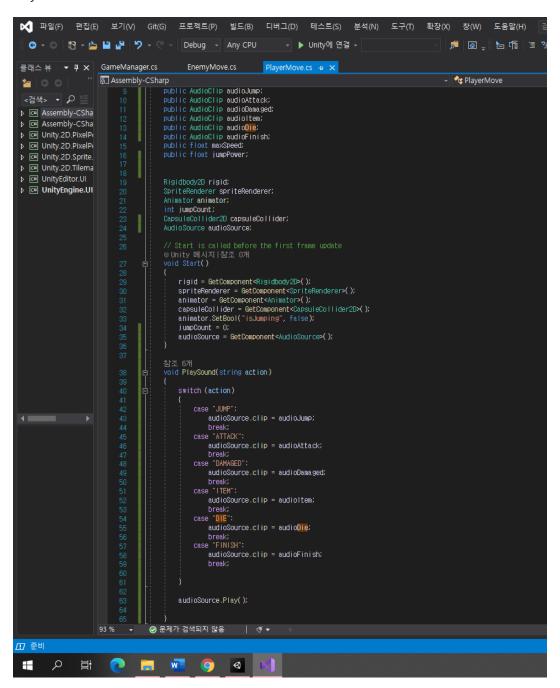
- 5. 매니저는 점수와 스테이지 관리
- 6. 스테이지
- 7. 유저 인터페이스

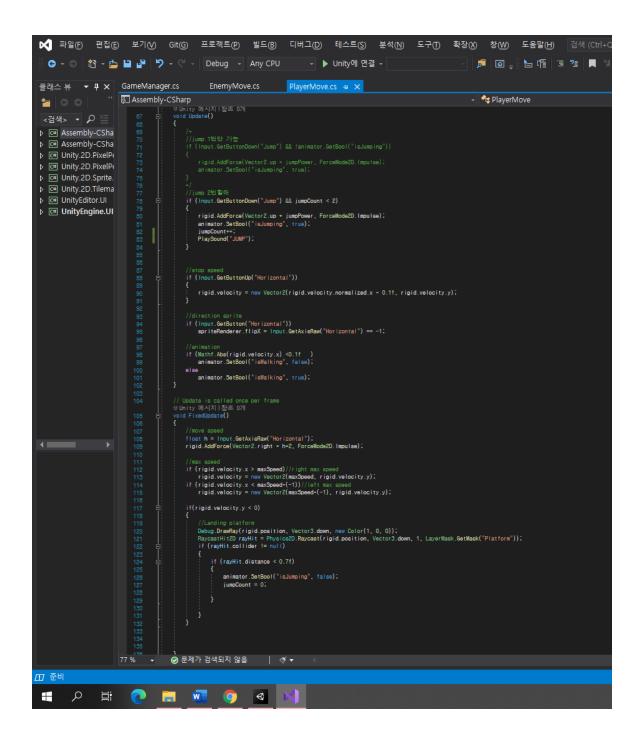
Canvas로. UnityEngine.UI import 해야함

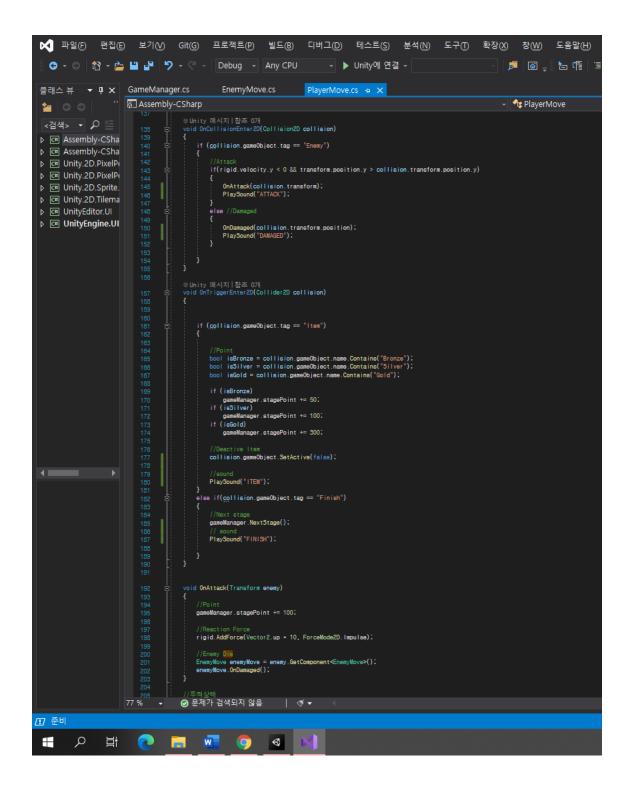
8. 사운드

플레이어에 audio source 하나, 효과음 audio clip 변수 추가

PlayerMove

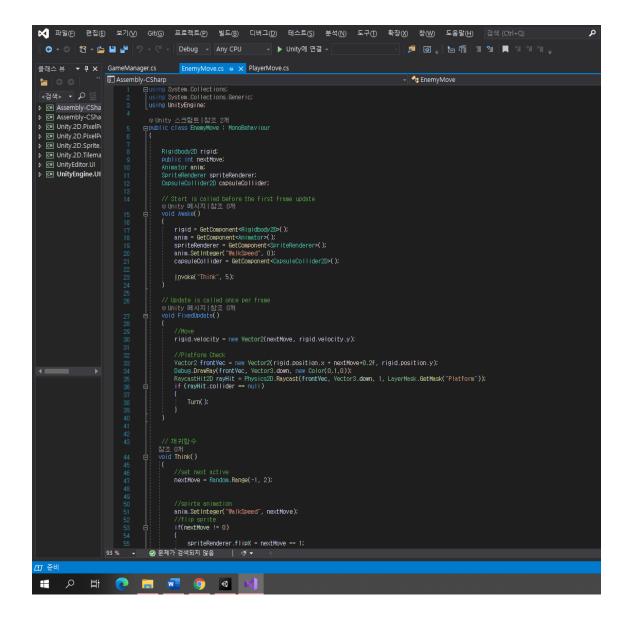






```
[M] 파일(F) 편집(E) 보기(V) Git(G) 프로젝트(P) 빌드(B) 디버그(D) 테스트(S) 분석(N) 도구(T) 확장(X) 창(W) 도움말(H)
   😋 - 🗇 📸 - 눮 💾 🥠 - ୯ - Debug - Any CPU - ▶ Unity에 연결 -
                                                                                                                                          🎜 🙆 🚅 ե 🖺 🖫
 클래스 뷰 ♡ ▼ 및 🗴 GameManager.cs EnemyMove.cs PlayerMove.cs 😕 🗶
 🍅 🕒 😁 '' 👸 Assembly-CSharp
                                                                                                                                         - PlayerMove
                                           void OnAttack(Transform enemy)
{
  <검색> ▼ 🔎 🔚
 ▶ C# Assembly-CSha
                                               //Point
gameManager.stagePoint += 100:
 ▶ C# Assembly-CSha
                                              //Reaction Force
rigid.AddForce(Vector2.up + 10, ForceMode20.Impulse);
 ▶ 🕮 Unity.2D.PixelPe
 ▶ C# Unity.2D.PixelPŧ
                                               //Enemy Die
Enemyllove enemyllove = enemy.GetComponent<Enemyllove>();
enemyllove.OnDamaged();
 ▶ C# Unity.2D.Sprite.
 ▶ C# Unity.2D.Tilema
 ▶ C# UnityEditor.UI
 ▶ 	☐ UnityEngine.UI
                                           //무적상태
참조 1개
woid OnDamaged(Vector2 targetPos)
{
                                               //Health down
gameManager.HealthDown();
                                               //change layer
gameObject.layer = 11:
                                               //view alpha
spriteRenderer.color = new Color(1, 1, 1, 0.4f);
                                               int dirc = transform.position.x - targetPos.x > 0 ? 1 : -1;
rigid.AddForce(new Vector2(dirc, 1) - 10, ForceMode2D.Impulse);
                                               animator.SetTrigger("doDamaged");
                                               Invoke("OffDamaged", 3):
                                           참조 0개
void OffDamaged()
                                               gameObject.layer = 10;
spriteRenderer.color = new Color(1, 1, 1, 1);
                                           public void OnDie()
                              238
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
                                               //spirte alpha
spriteRenderer.color = new Color(1, 1, 1, 0.4f);
                                               spriteRenderer.flipY = true:
                                               capsuleCollider.enabled = false:
                                               rigid.AddForce(Vector2.up = 10, ForceMode2D.Impulse):
PlaySound("DIE");
                                           참조 1개
public void VelocityZero()
                                              rigid.velocity = Vector2.zero;
                          77 % ▼ 🥝 문제가 검색되지 않음 │ 🚿 ▼
🗓 준비
           0
                    計
```

EnemyMove



```
ⓒ - ◎ 🏥 - 🍅 💾 🛂 🖖 - ♡ - ♡ - Debug - Any CPU - ▶ Unity에 연결 -
                                                                                                                                                                                                                                                                                                                                  클래스뷰 ▼ I X GameManager.cs EnemyMove.cs • X PlayerMove.cs • X Play
                                                                                                                                                                                                                                                                                                                                      - 🔩 EnemyMove
                                                                                              y-CSharp
// 서귀당수
참조 0개
B: void Think()
{
//set next active
nextMove = Random.Range(-1, 2);
//spirte animation
anim.SetInteger("WalkSpeed", nextMove);
                                                                                                                        if(nextMove != 0)
                                                                                                                          spriteRenderer.flipX = nextMove == 1;
}
                                                                                                                         //recursive
float nextThinkTime = Random.Range(2f, 5f);
!nvoke("Think", nextThinkTime);
                                                                                                                          nextMove *= -1;
spriteRenderer.flipX = nextMove == 1;
CancelInvoke();
!mvoke("Think", 5);
                                                                             76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
                                                                                                                          //Sprite Alpha
spriteRenderer.color = new Color(1, 1, 1, 0.4f);
                                                                                                                          spriteRenderer.flipY = true;
                                                                                                                         //Collider disable
capsuleCollider.enable ■ struct System.Boolean
                                                                                                                          //Distroy
<u>invoke("DeActive", 5)</u>;
                                                                                                                참조 0개
void DeActive()
                                                                                                                            gameObject.SetActive(false);
                                                               🗓 준비
   # 2 詳 🙋 🥫 🚾 🕥
```

GameManager

