



[10주차] 3D BombGame-2

인천정보과학고등학교 3학년 게임프로그래밍

© all right reserved 인천정보과학고 IT소프트웨어과 정수영

목차

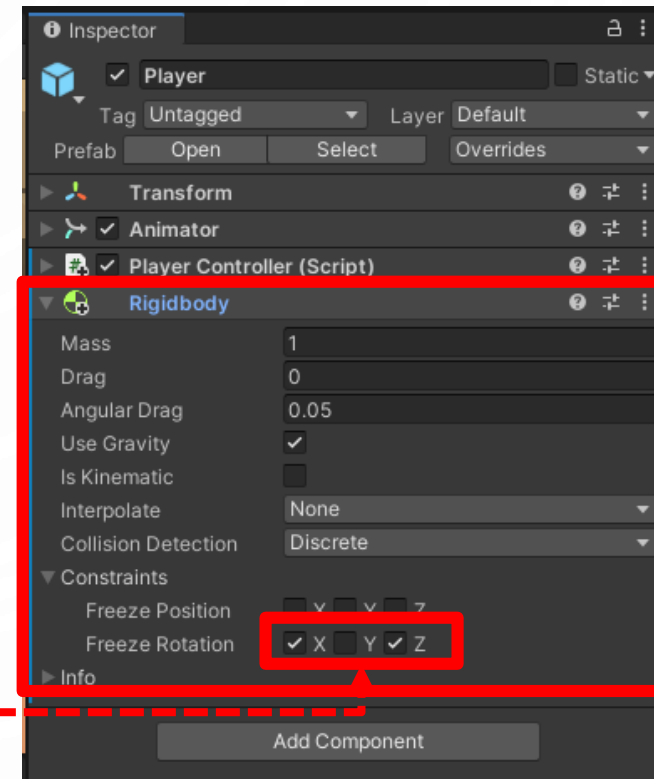
- Player에 Rigidbody와 Collider 넣기
- 배경 프리팹에 Collider 넣기
- 이동 및 충돌 애니메이션
- UI 만들기
- 깃허브 올리기

PLAYER에 RIGIDBODY와 COLLIDER 넣기

Rigidbody

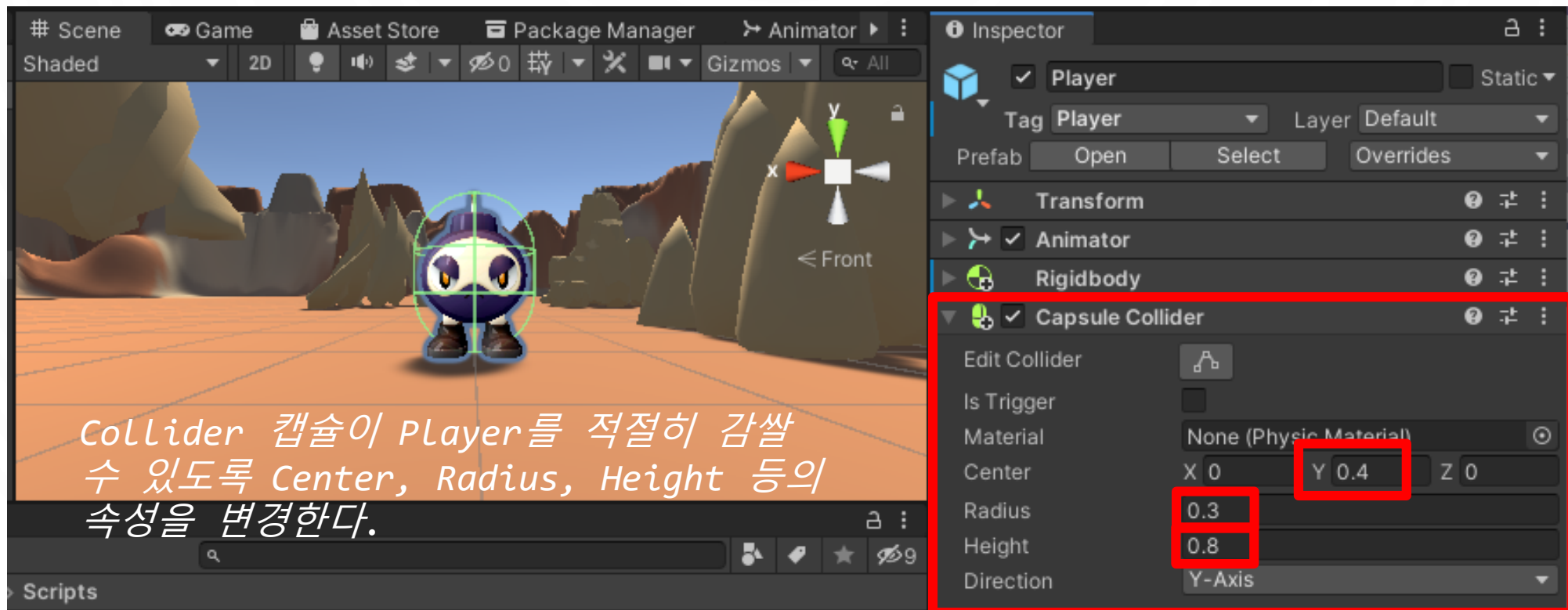
- Rigidbody는 오브젝트의 물리값(중력, 미는힘 등)을 담당
- Player 선택 후 인스펙터 창에서 Add Component > Rigidbody

Constraints 항목에서
X, Z축 회전을 고정시킨다.

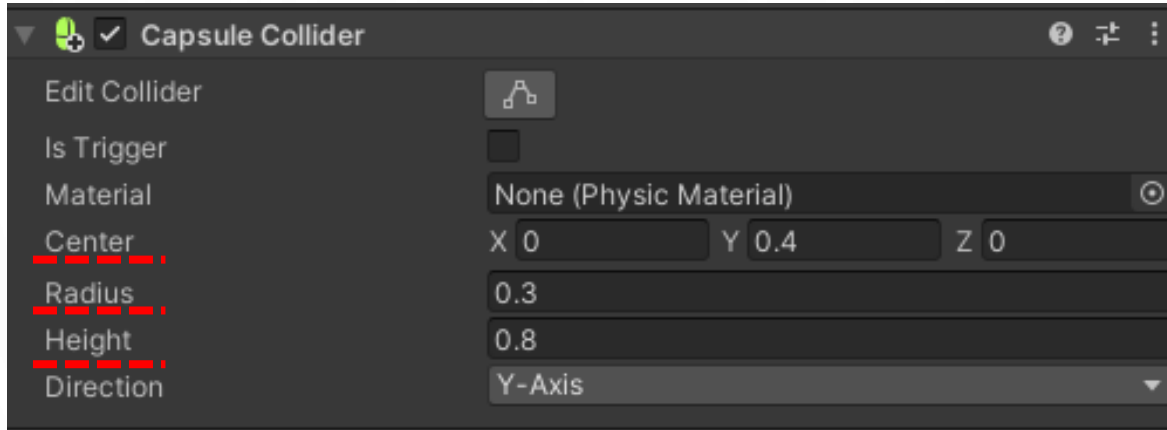


Collider

- Collider는 Collider가 있는 오브젝트끼리의 충돌을 감지한다.
- Player 선택 후 인스펙터 창에서 Add Component > Capsule Collider



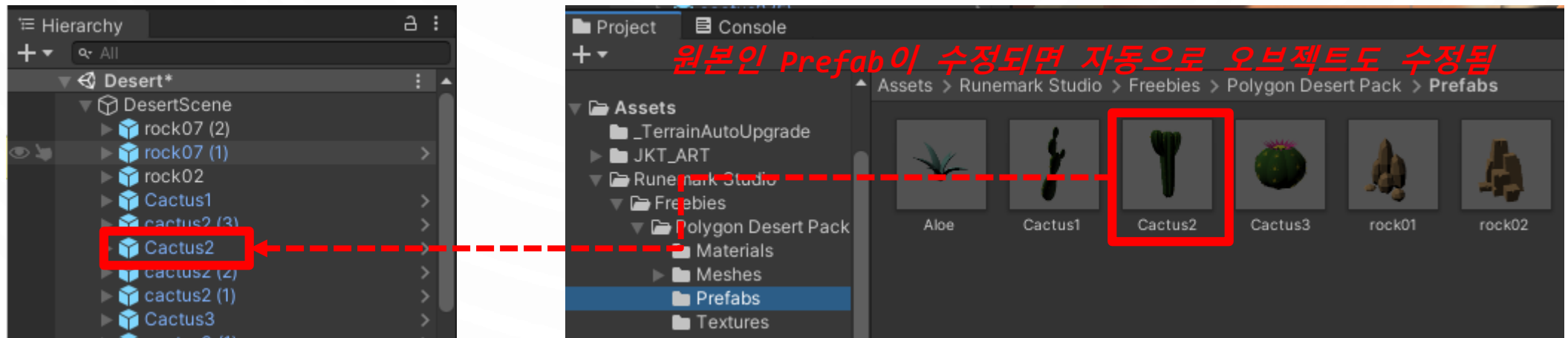
Capsule Collider 설정



배경 프리팹에 COLLIDER 넣기

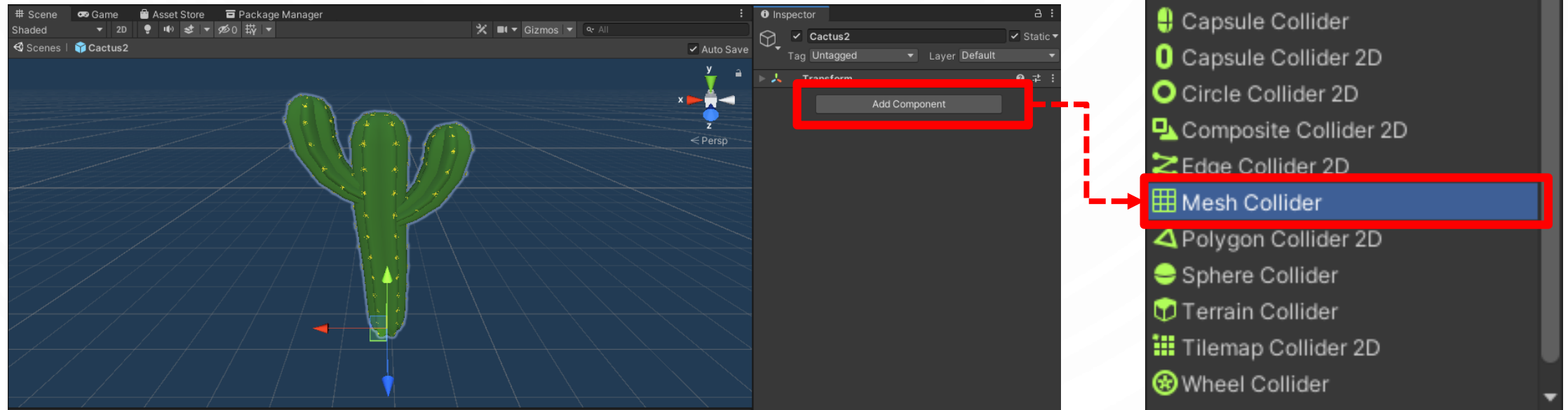
배경 요소들에 Collider 넣기

- 수많은 선인장과 돌들에 하나하나 콜라이더를 넣는 것은 엄청난 시간 낭비입니다.
- 하이어라키뷰에서 파란색의 오브젝트는 프리팹(Prefab)에 연결되어 있습니다.
- 이 프리팹들에 Collider를 씌워주면 해당 오브젝트들도 자동으로 Collider가 씌워집니다.

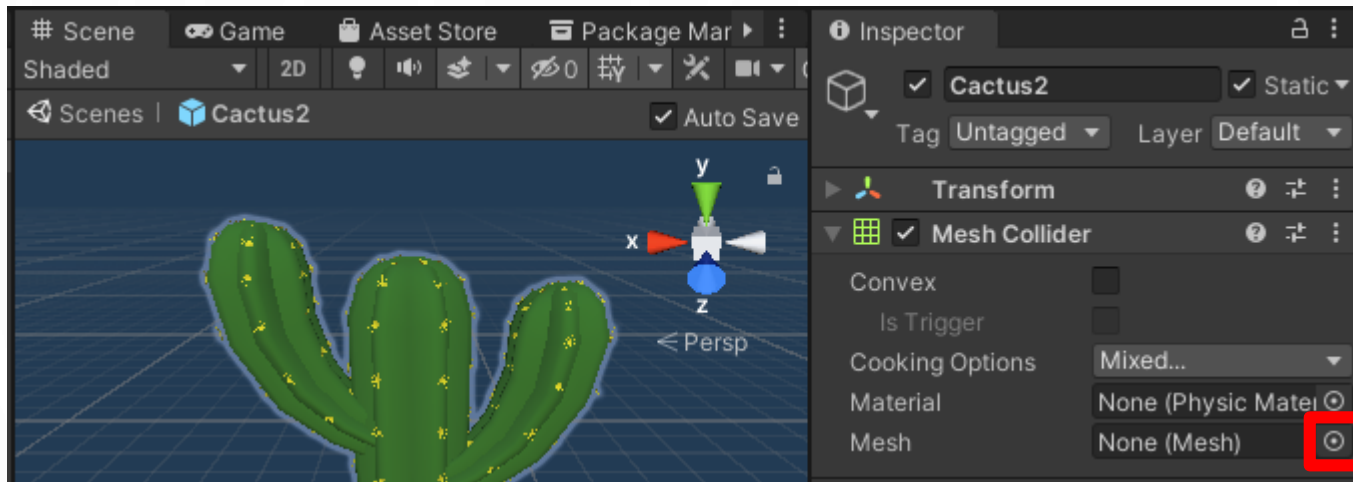


Prefab에 Collider 추가

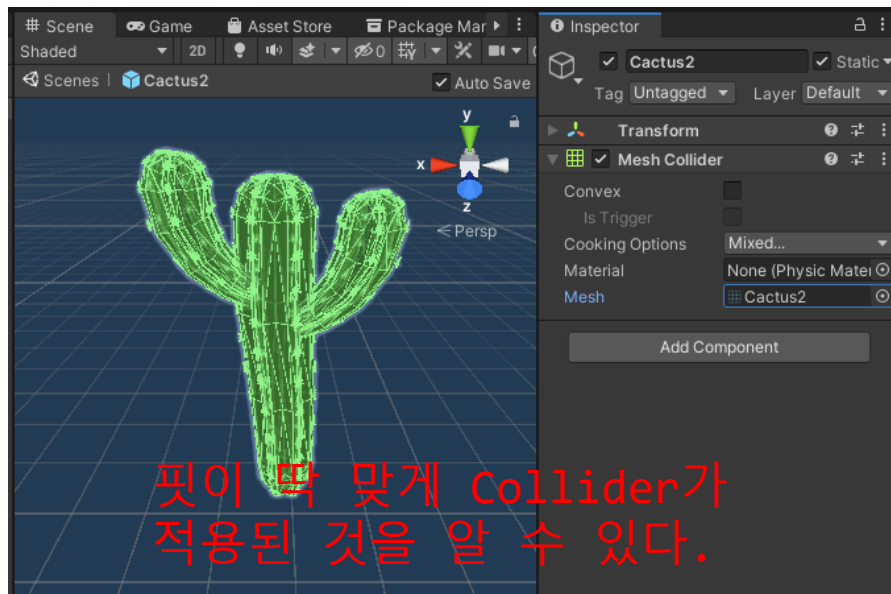
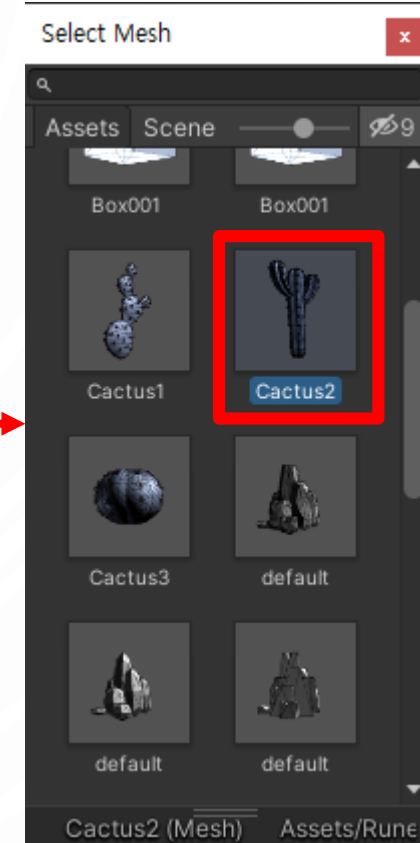
- Assets/Runemark Studio/Freebies/Polygon Desert Pack/Prefabs 폴더에서
- Cactus2를 더블클릭하여 프리팹을 열어준다.
- Collider 중 MeshCollider를 선택한다.



Mesh Collider



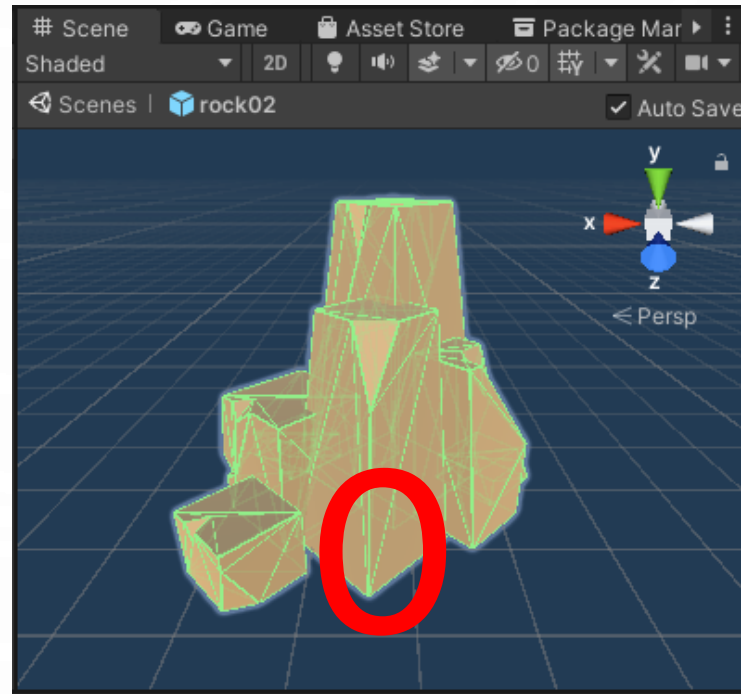
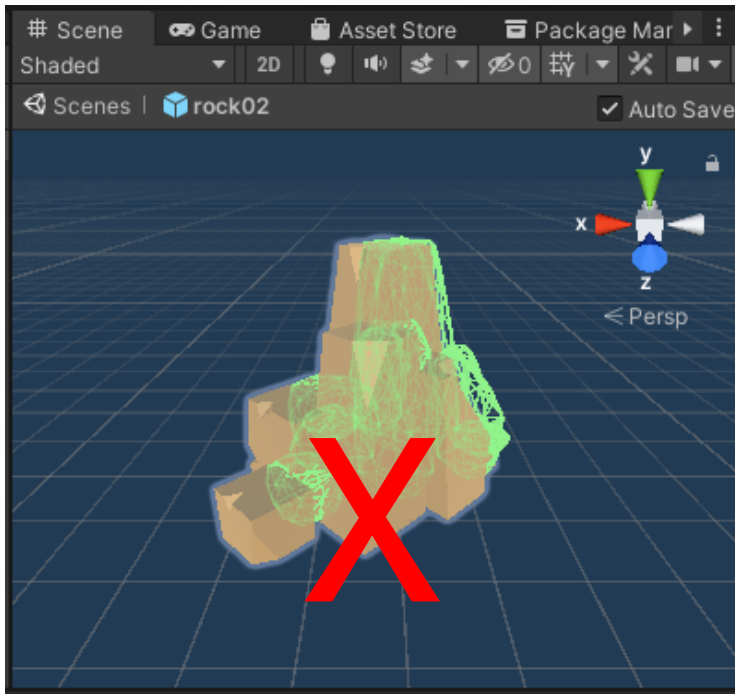
Mesh로 Cactus2를
선택하면



핏이 딱 맞게 Collider가
적용된 것을 알 수 있다.

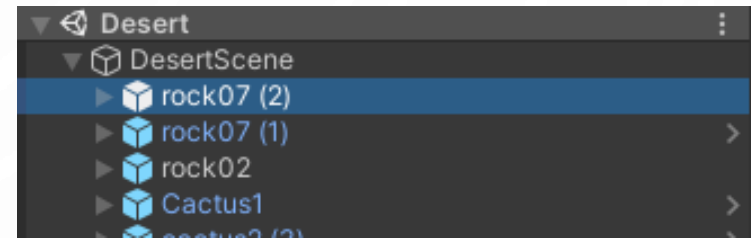
모든 Prefab에 Mesh Collider

- 이제 Prefabs 안의 모든 프리팹들에 대해 똑같이 Mesh Collider를 적용한다.
- Collider의 핏이 잘 맞는지 확인하면서 진행한다.

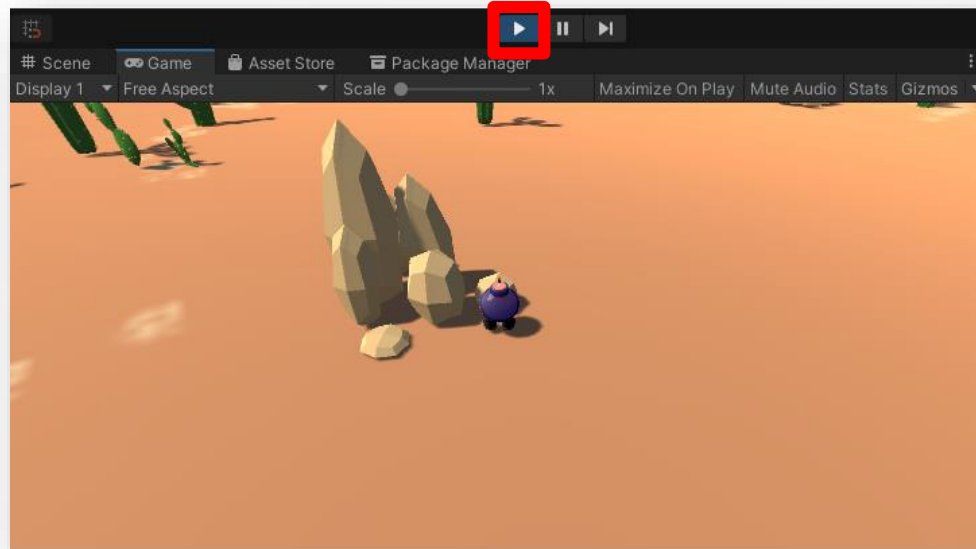


충돌 테스트

- 다음과 같이 흰색글씨로 된 오브젝트들은 해당 프리팹이 없는 경우이기 때문에 직접 오브젝트를 선택하여 Mesh Collider를 주거나 그냥 삭제해버린다. (삭제 권장)



- Play하여 충돌이 잘 일어나는지 확인한다.

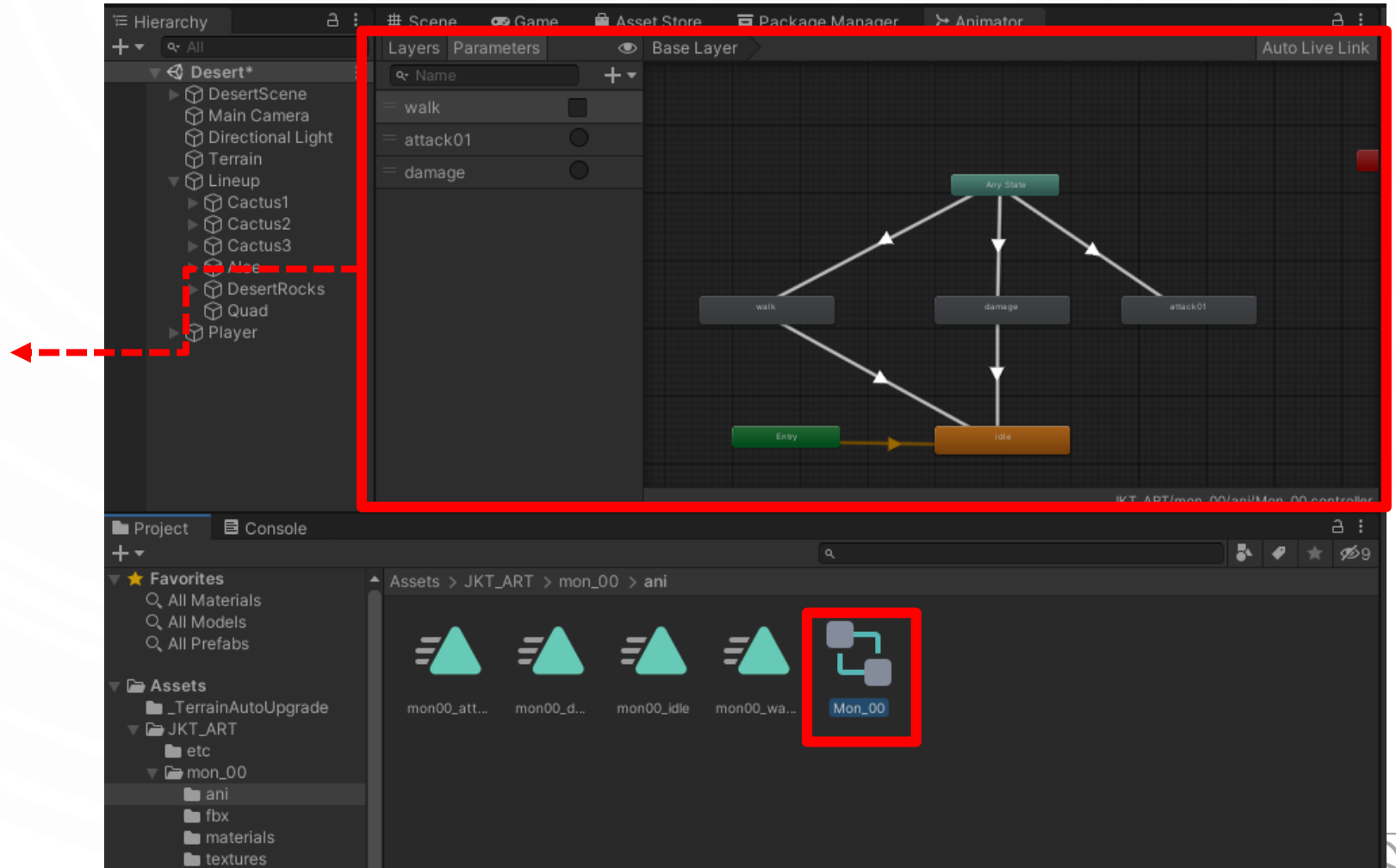


이동 및 충돌 애니메이션

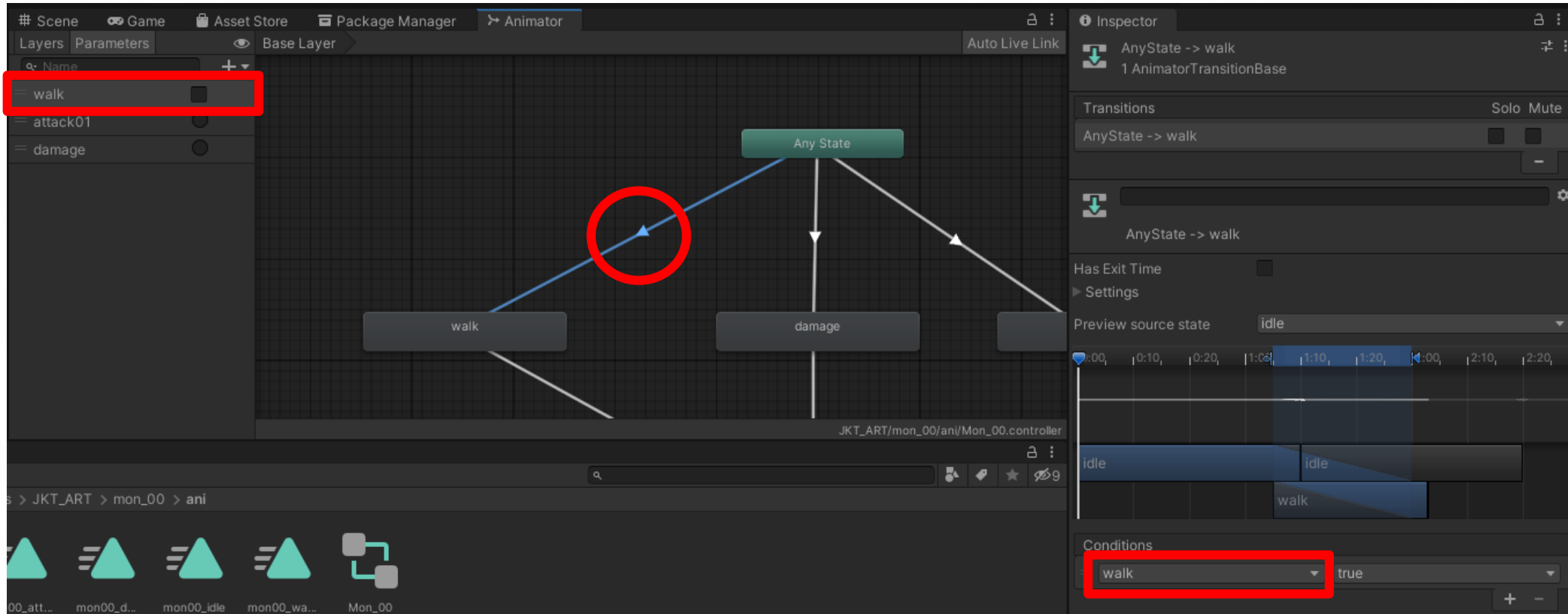
Player의 애니메이션

- Player의 애니메이션은 Assets/JKT_ART/mon_00/ani 폴더에서
- Mon_00 더블 클릭

Player(원본 Mon_00)에
설정된 애니메이션
컨트롤러를 볼 수 있다.

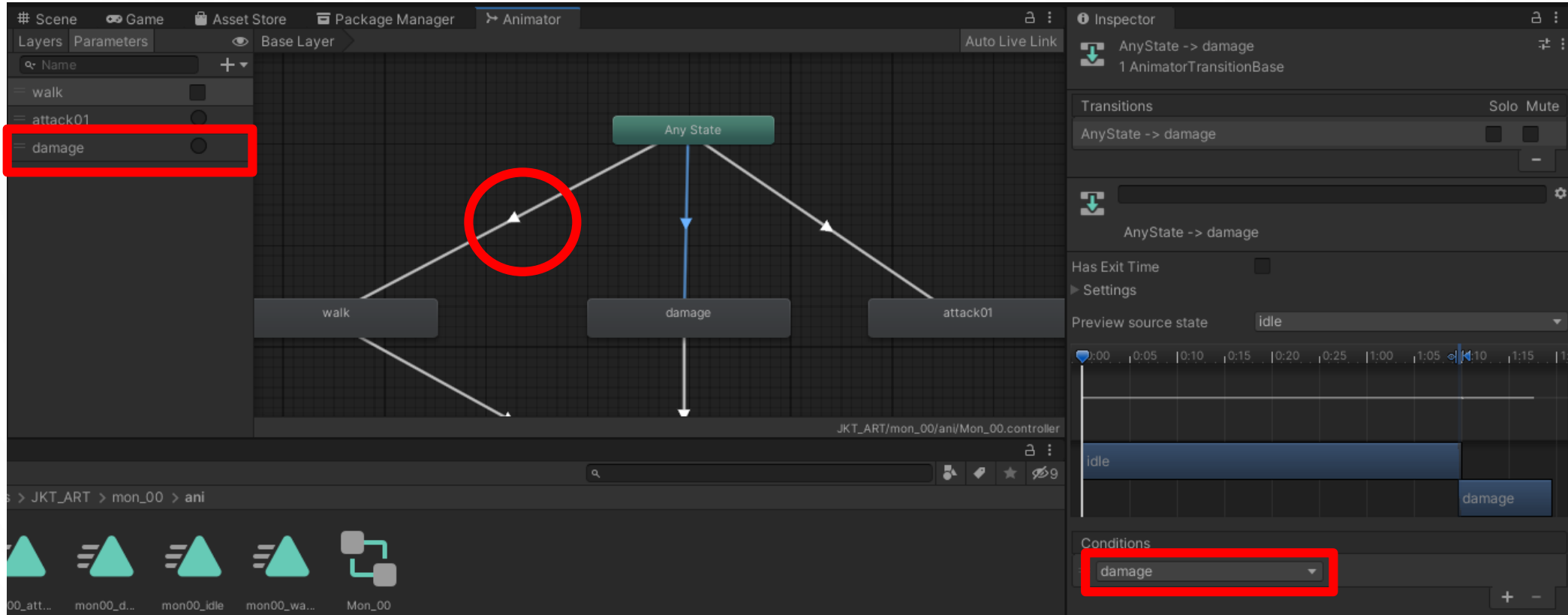


walk 애니메이션



walk로 가는 트랜지션 화살표를 살펴보면
“walk” 파라미터가 true로 되면 변한다는 것을 알 수 있다.

damage 애니메이션



damage로 가는 트랜지션 화살표를 살펴보면
“damage” 파라미터가 트리거 되면 변한다는 것을 알 수 있다.

PlayerController 스크립트

```
public class PlayerController : MonoBehaviour
{
    private float translation;
    private float rotation;

    private Animator ani;

    private void Start()
    {
        ani = GetComponent<Animator>();

    }

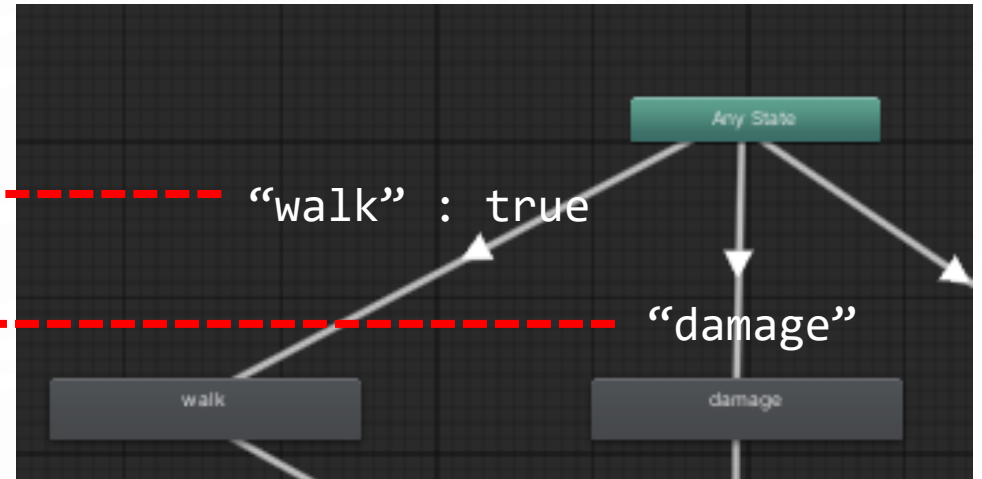
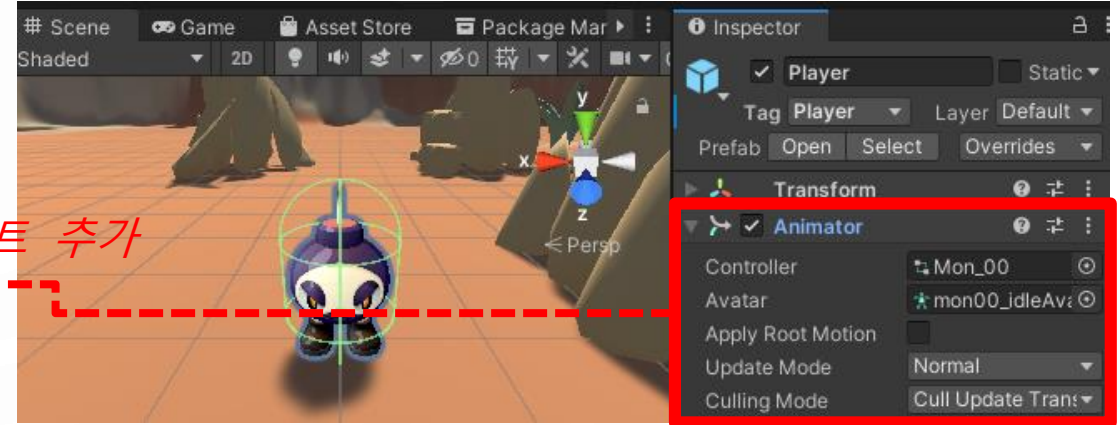
    void Update()
    {
        translation = Input.GetAxis("Vertical") * 10f;
        rotation = Input.GetAxis("Horizontal") * 100f;

        transform.Translate(0, 0, translation * Time.deltaTime);
        transform.Rotate(0, rotation * Time.deltaTime, 0);

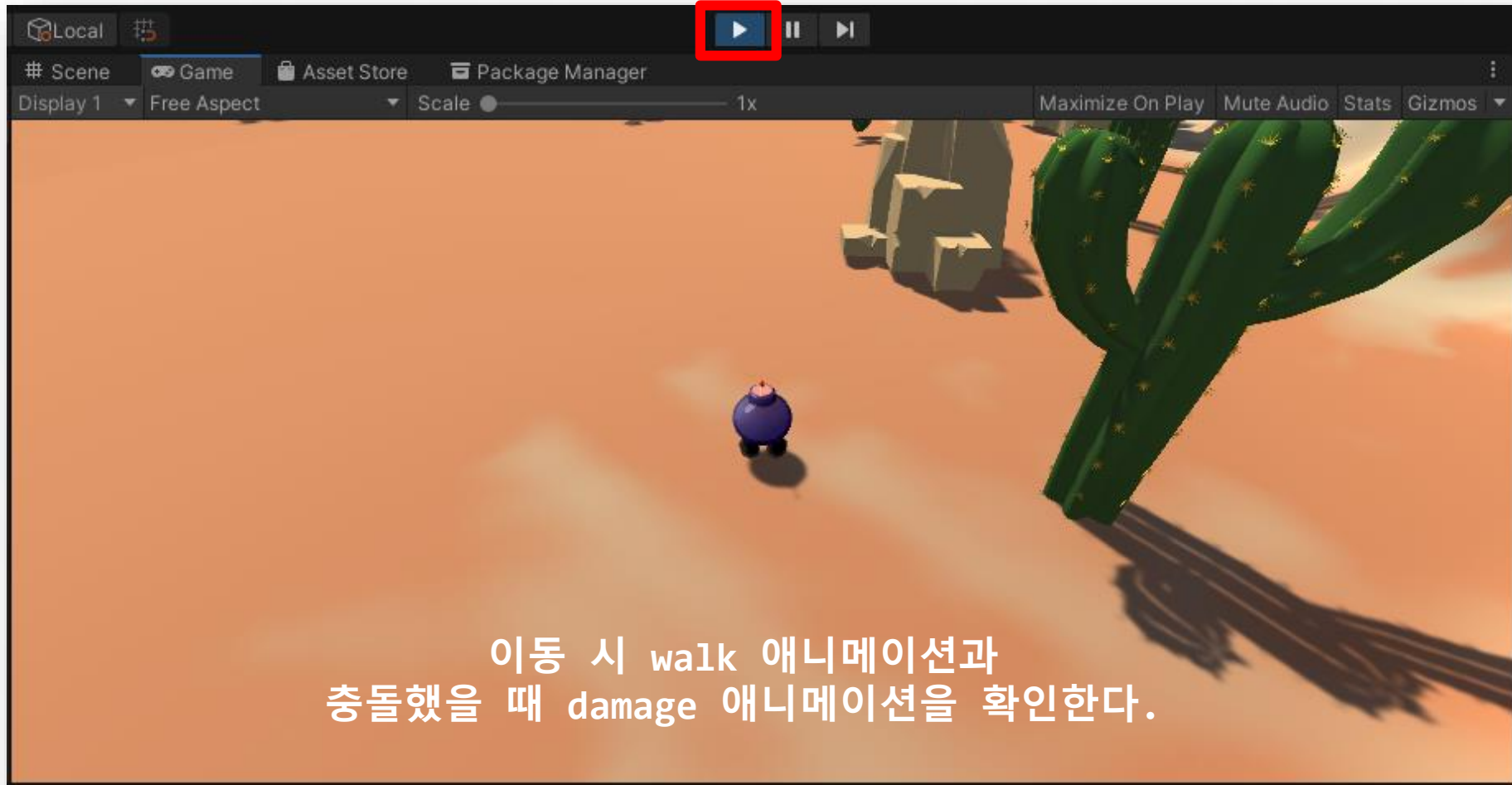
        if(translation != 0 || rotation != 0)
        {
            ani.SetBool("walk", true);
        }
        else
        {
            ani.SetBool("walk", false);
        }

        private void OnCollisionEnter(Collision collision)
        {
            if(collision.collider.tag == "Obstacle")
            {
                ani.SetTrigger("damage");
            }
        }
    }
}
```

Animator 컴포넌트 추가



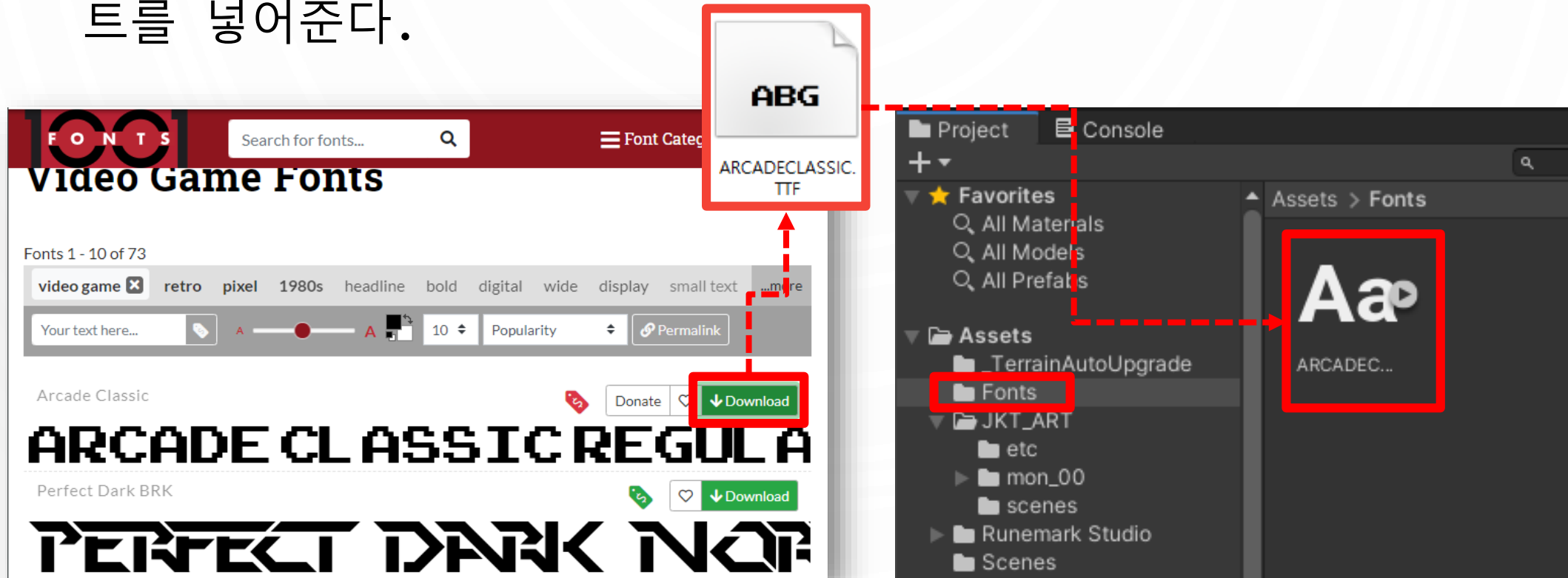
결과 확인하기



UI 만들기

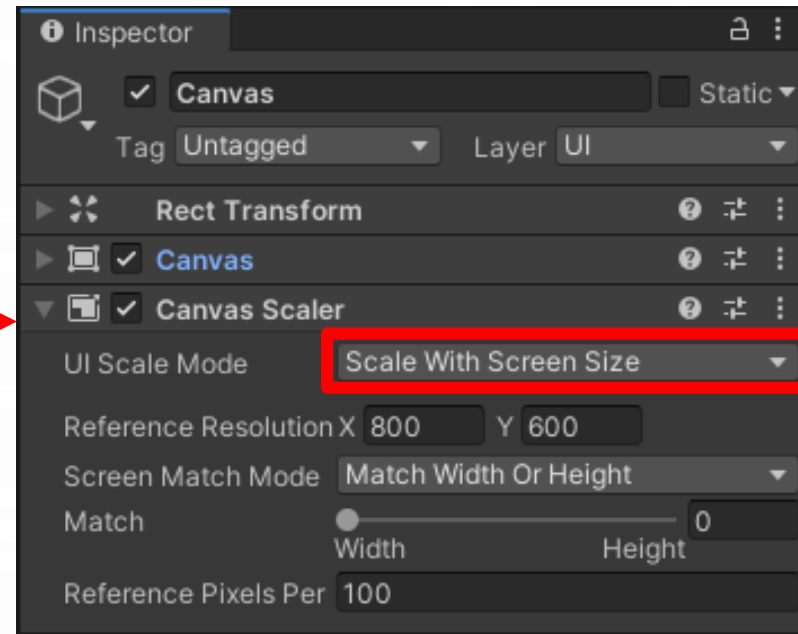
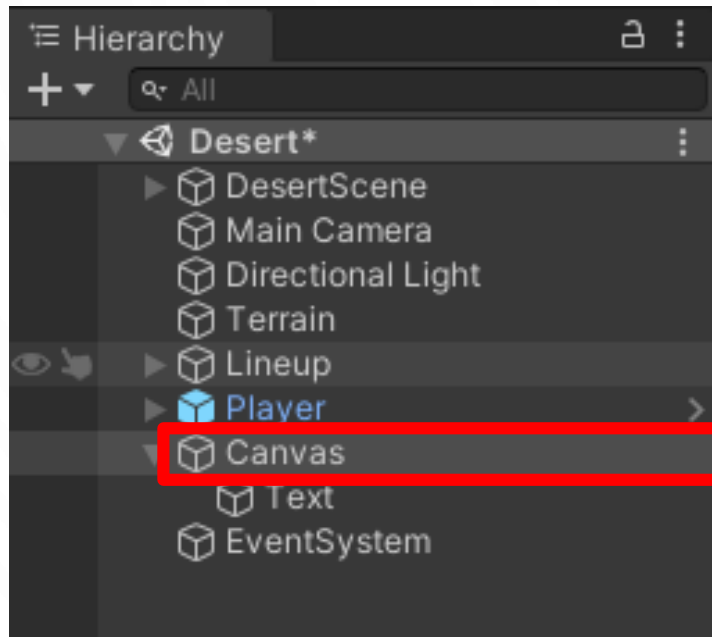
게임 폰트 정하기

- <https://www.1001fonts.com/> 에서 게임 폰트를 다운 받는다.
- Assets 폴더 하위에 Fonts 폴더를 만들고 그 안에 다운 받은 폰트를 넣어준다.



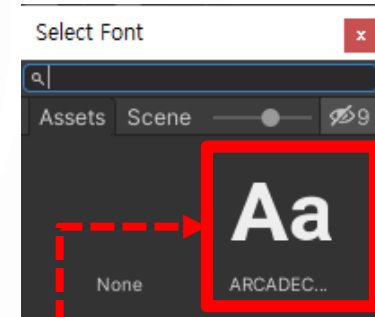
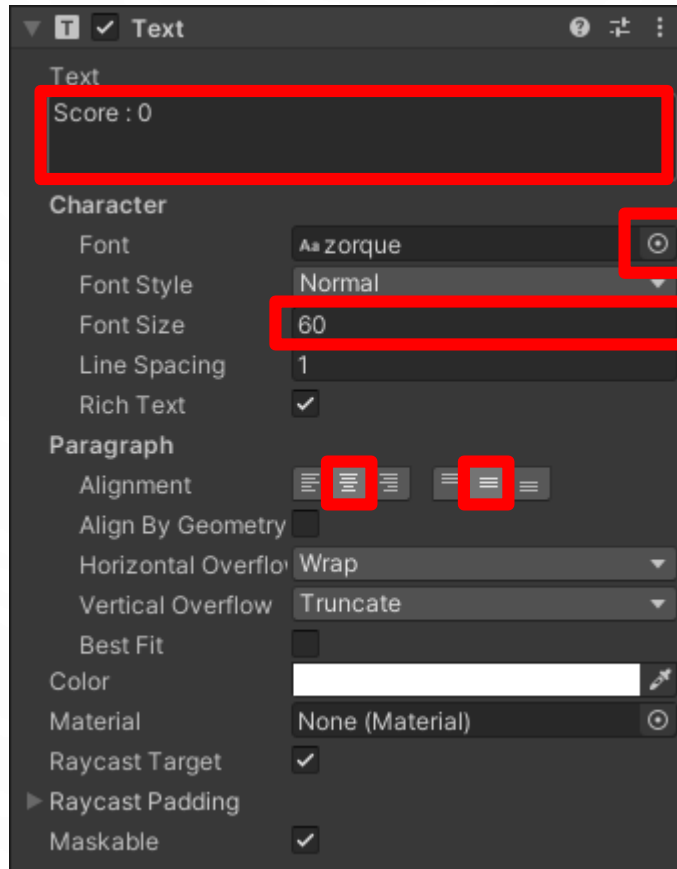
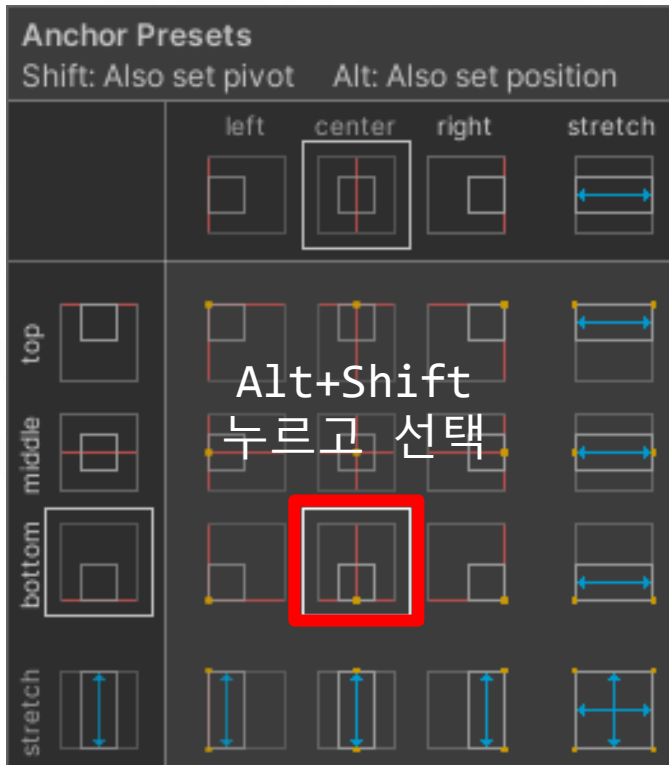
Canvas 수정

- 하이어라키뷰에서 마우스 오른쪽 버튼 > UI > Text 선택
- Canvas를 선택 > 인스펙터뷰 Canvas Scaler 에서 UI Scale Mode를 'Scale With Screen Size'로 수정한다.

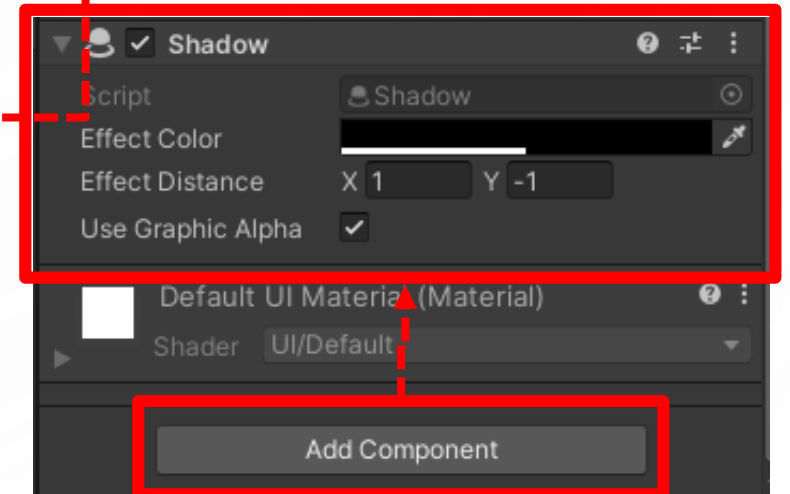


Score Text

- Text를 Score Text로 이름을 변경
- Width 400, Height 100



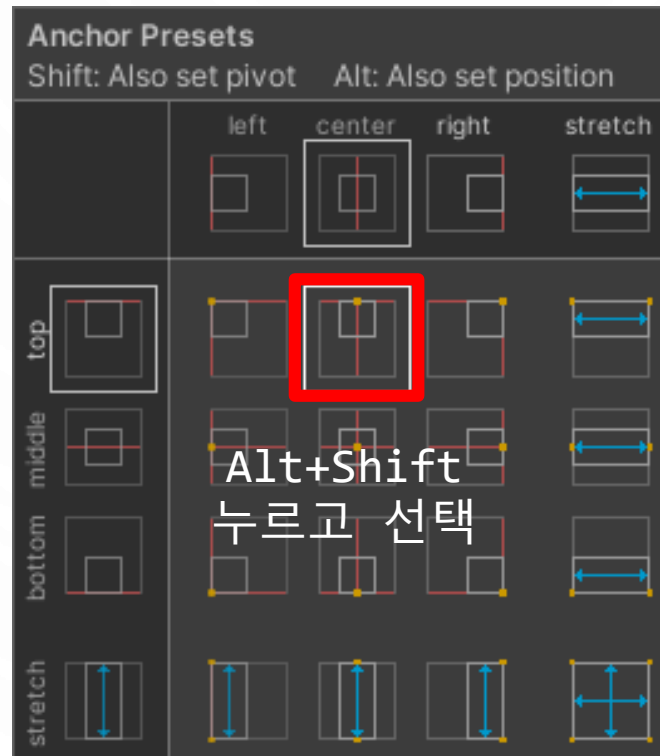
다운 받은
폰트 선택



Add Component로 Shadow
선택하여 넣어주기

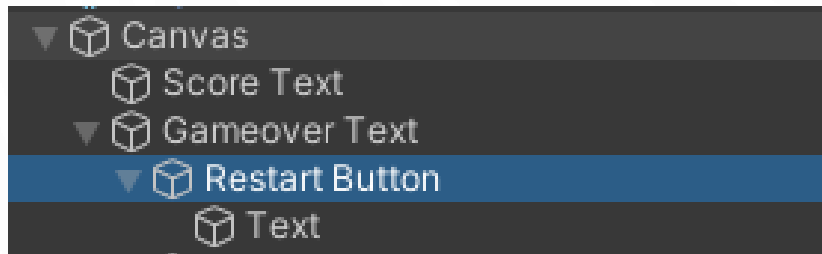
GameOver Text

- Score Text를 선택 후 Ctrl+D를 누르면 복제가 된다.
- Gameover Text를 Anchor Pivot을 수정해서 위치 수정

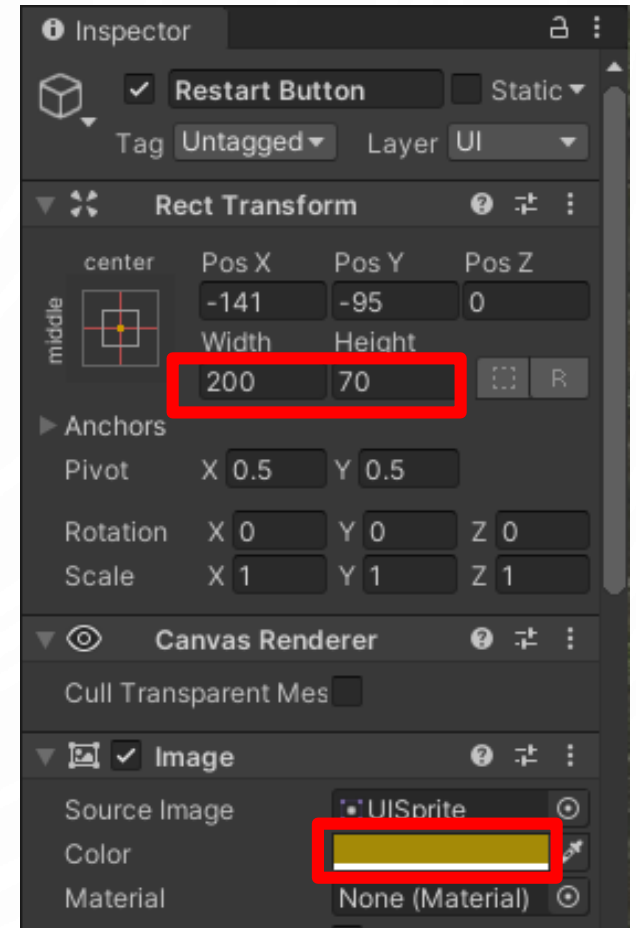


Restart Button

- Gameover Text를 선택한 후 마우스 오른쪽 버튼 > UI > Button
- Button 이름을 Restart Button으로 바꾼다.

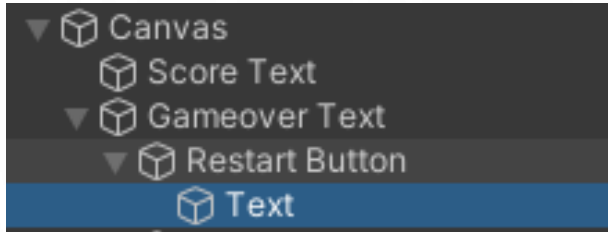


- Width 200, Height 70 (자유롭게 조정 가능)
- Button의 Color를 마음에 드는 색으로 바꾼다.

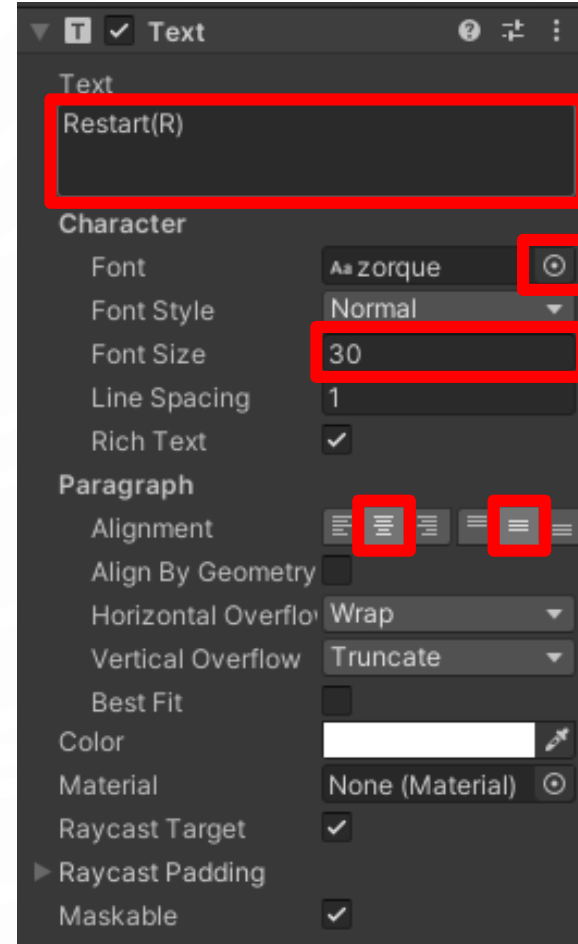


Restart Button의 Text

- 하이어라키뷰에서 Restart Button 하위의 Text를 선택한다.

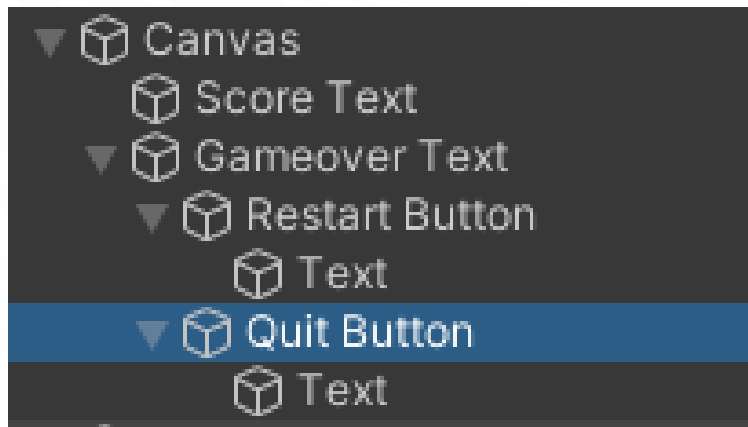


- 내용을 'Restart(R)'로 바꾼다.
- Font와 Size를 수정한다.
- Alignment를 맞춘다.
- 기타 원하는 부분을 수정한다.



Quit Button

- Restart Button을 복제(Ctrl+D)한다.
- 복제된 Button 이름을 Quit Button으로 바꾼다.
- Restart Button 옆에 적당한 위치에 배치시킨다.
- Button의 Color는 자신이 좋아하는 색으로 정한다.
- Button 하위의 Text를 'Quit(Q)'과 같이 수정한다.



버튼 완성

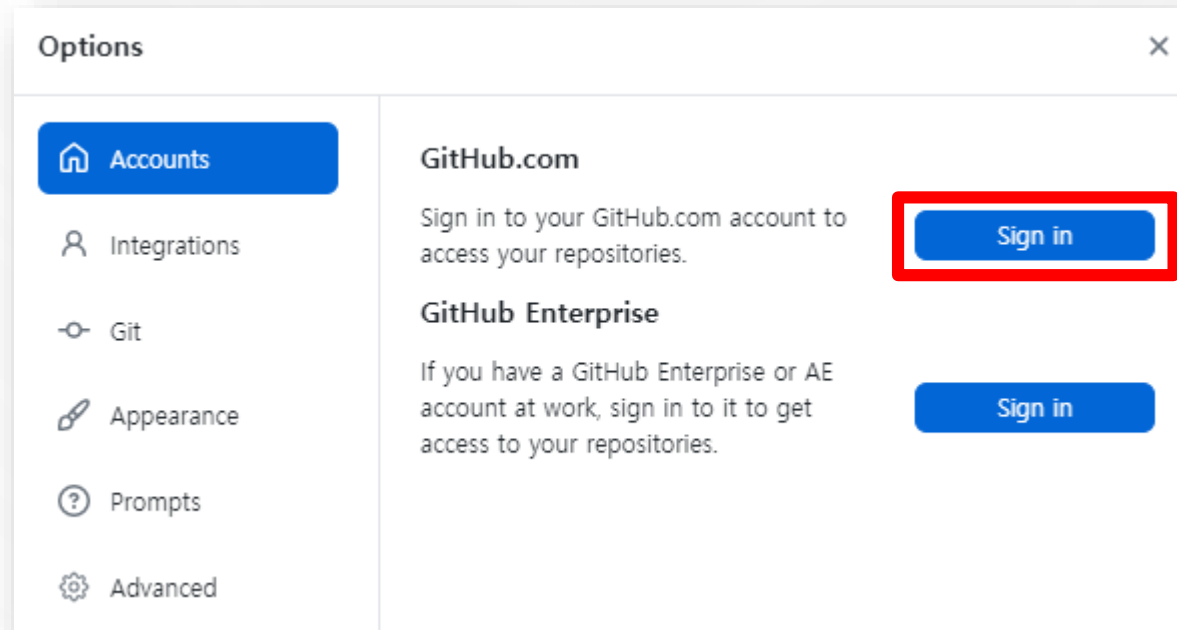
- Gameover Text에 Gameover 문구를 넣고 숨김처리 한다.



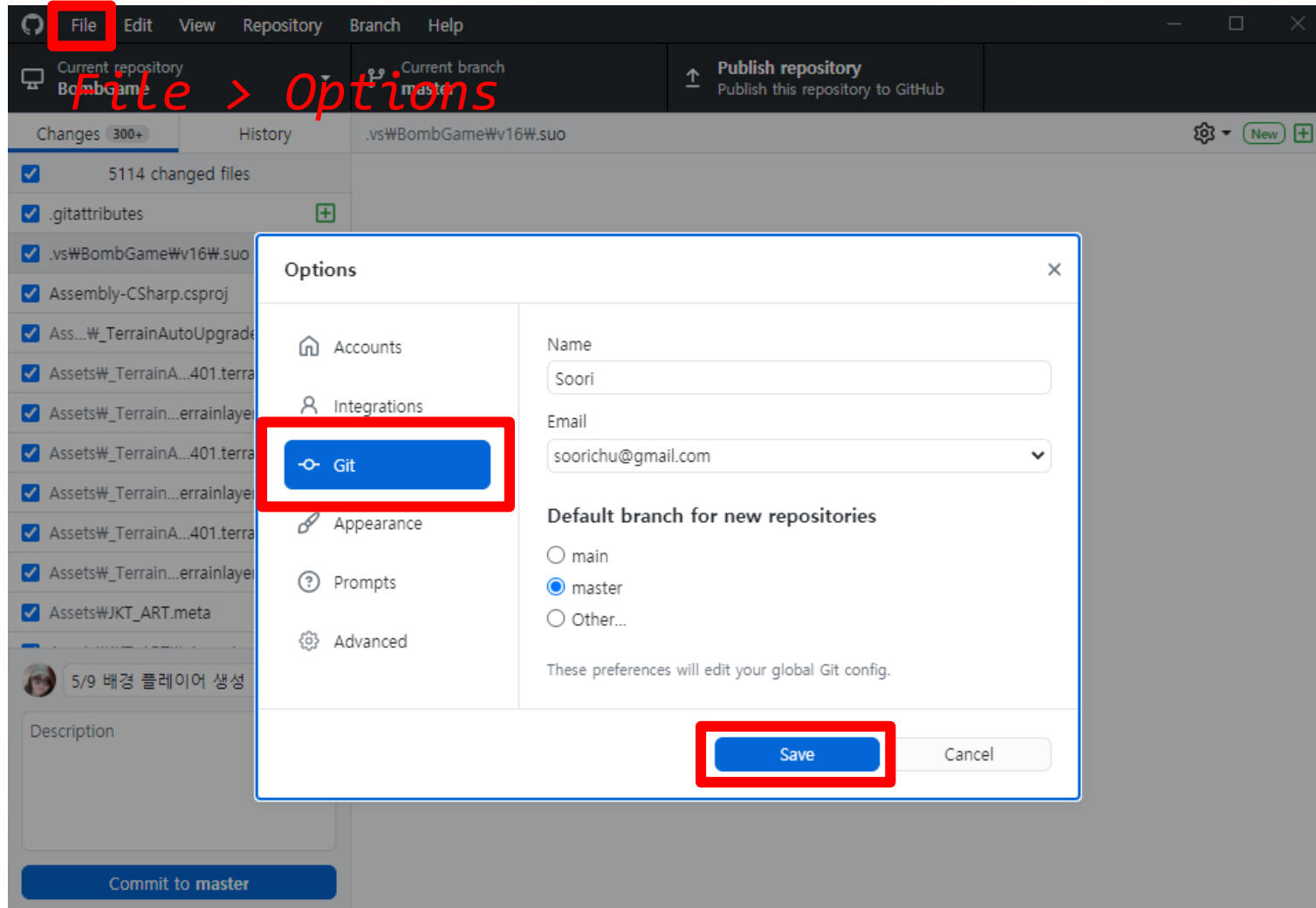
깃허브 올리기

Git Desktop

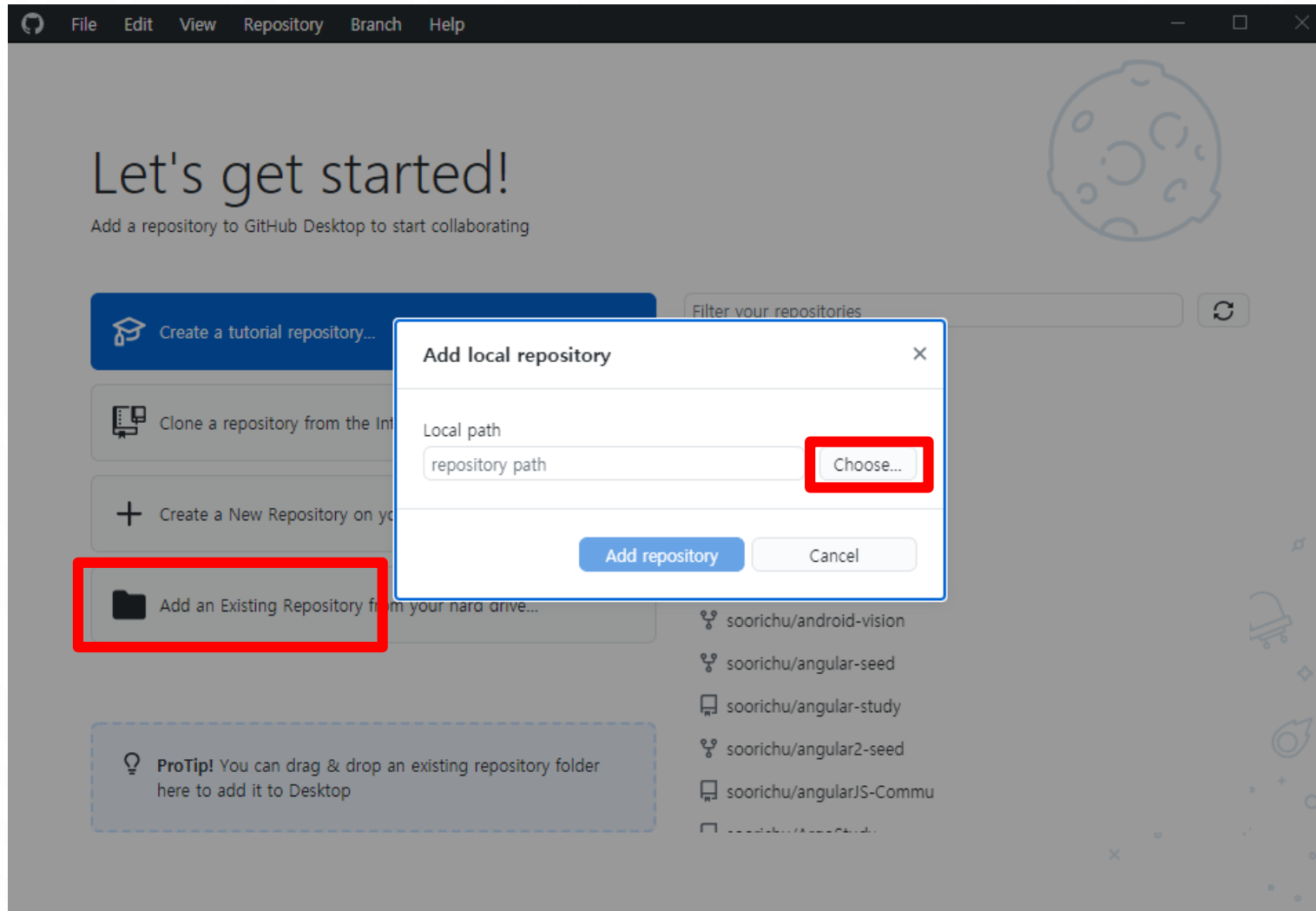
- 프로젝트를 저장(Ctrl+S)한 후 모든 창을 닫고 빠져나온다.
- Git Desktop을 실행한다.
- 상단 메뉴 File > Options에서 Github.com에 Sign In



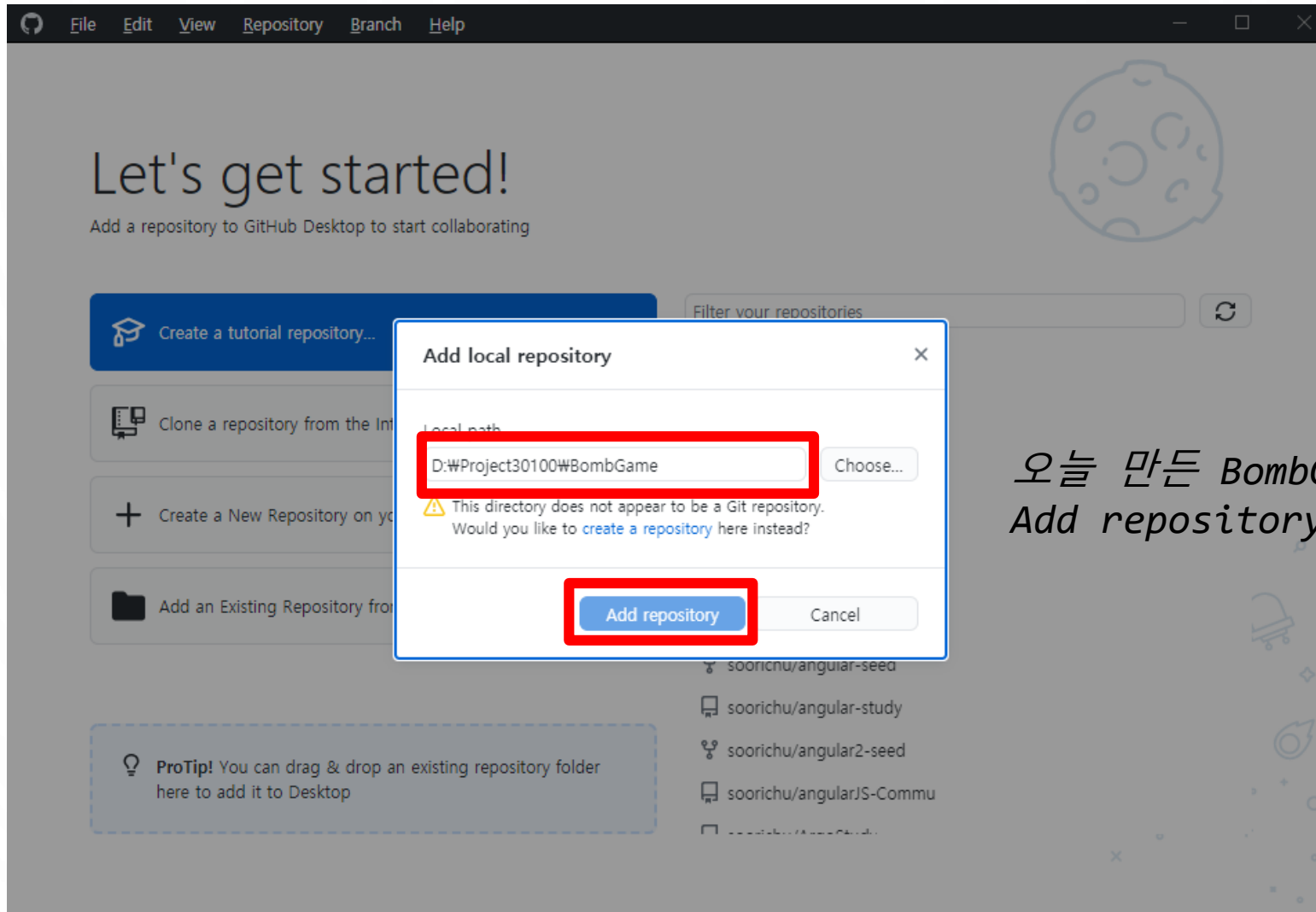
오류 방지



Add and Existing Repository ...

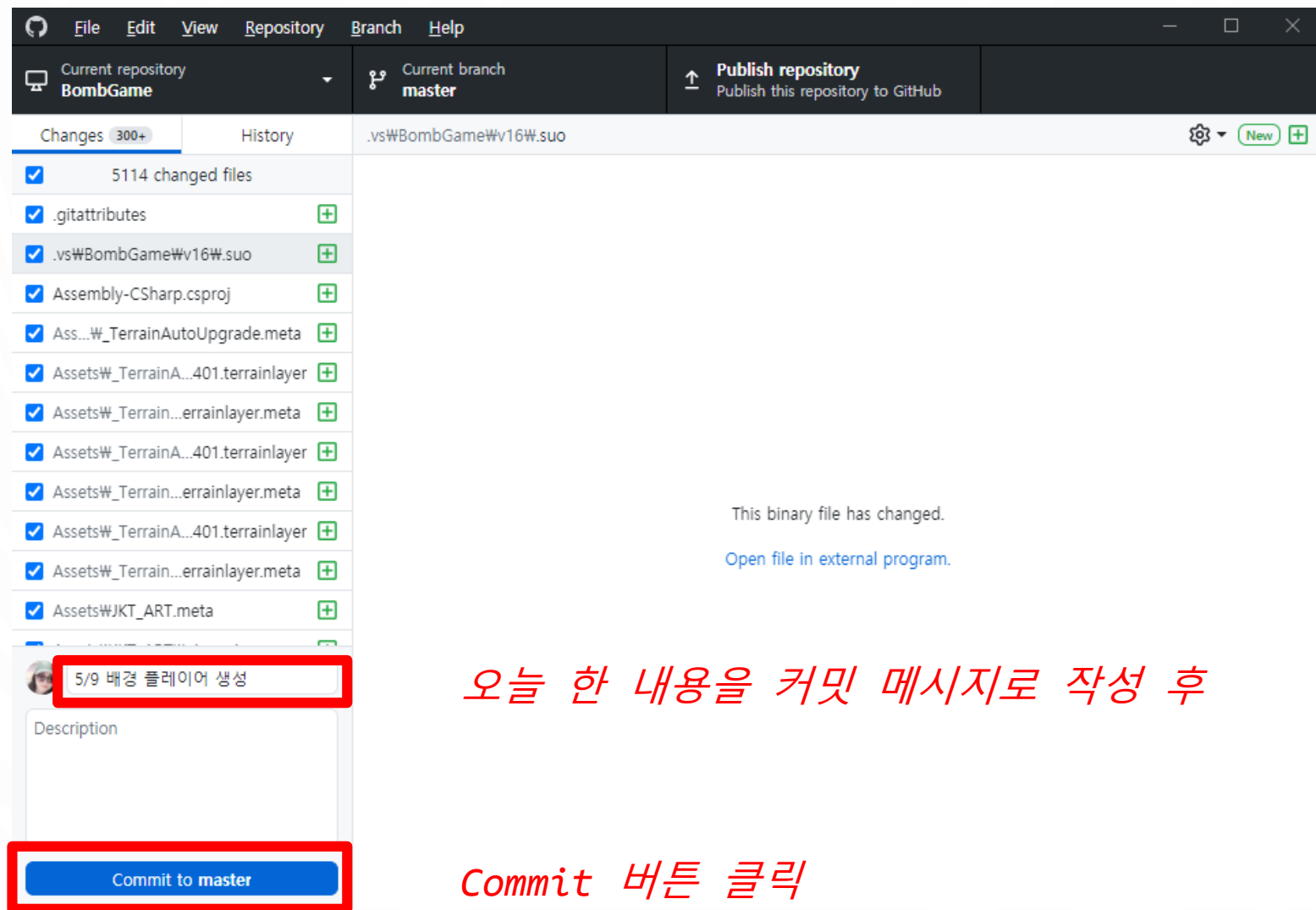


Create a repository



오늘 만든 BombGame을 가져온뒤
Add repository

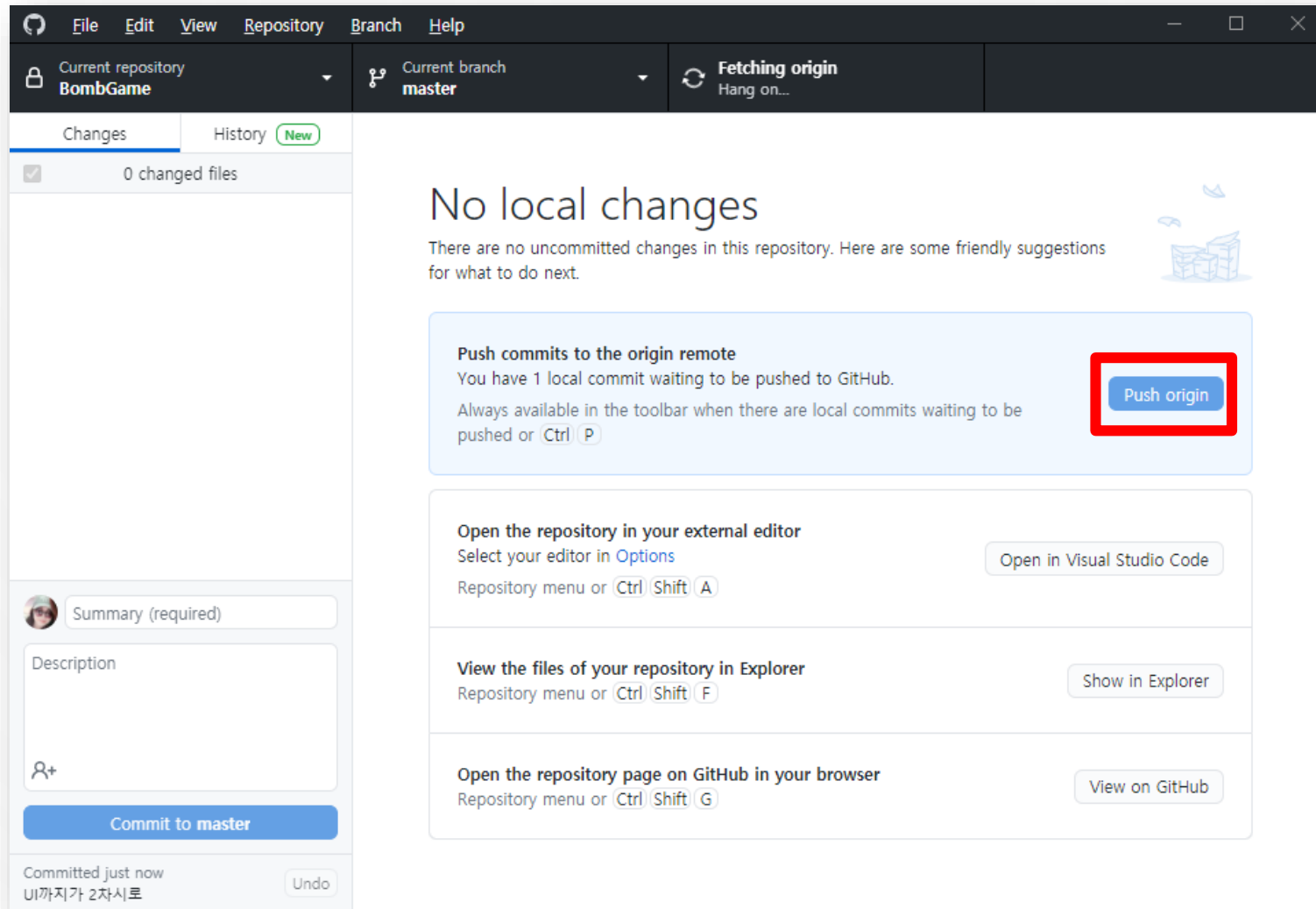
Commit



오늘 한 내용을 커밋 메시지로 작성 후

Commit 버튼 클릭

Push하기



마지막으로 Push
꼭 하세요!!

END

깃허브 올리기 전에 반드시 검사 받으시고 수행평가 종이 잊지 말고 작성해서 제출해주세요.

학습자료는 *INJUNG.SITE* 에 모아져 있습니다. 언제든지 찾아와 공부하세요!

항상 열심히 해주어서 고맙습니다. ^^