이번 주 메인 프로젝트 Commit 로그

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 메인 프로젝트 클래스 컴포넌트화

기존의 Engine project에 있는 코드들을 메인 프로젝트의 코드들과 병합시켰다.

Component Class

텍스트이(가) 표시된 사진

자동 생성된 설명

컴포넌트들의 기본 클래스이며 추상 클래스로 만들어 상속하여 사용한다.

Start() – 컴포넌트가 생성될 때 처음 시작하는 함수 (ex: 초기화)

Update() – 매 프레임 마다 실행하는 함수 (실질적인 기능을 담당하는 부분이다)

텍스트이(가) 표시된 사진

자동 생성된 설명

컴포넌트 사용 예 – RotateComponent(매 프레임마다 회전하는 기능)

Object Class



오브젝트는 컴포넌트들의 리스트를 갖는다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Start() – 갖고 있는 모든 컴포넌트들의 Start()를 호출한다.

Update() - 갖고 있는 모든 컴포넌트들의 Update()를 호출한다.

AddComponent() – 컴포넌트 클래스를 템플릿으로 넣으면 오브젝트에 해당 컴포넌트를 추가한다.

GetComponent() – 컴포넌트 리스트에 해당 컴포넌트가 있는지 확인하고 반환한다.

오브젝트 추가하는 법

텍스트이(가) 표시된 사진

자동 생성된 설명

오브젝트에 생성자에 GameScene의 생성 큐에 본인의 포인터를 추가하여 처리하기 때문에

어디서든지 오브젝트를 할당하면 뒤에 설명하는 GameScene의 오브젝트 리스트에 추가되어 관리한다.

오브젝트 생성 및 컴포넌트 추가 예시

텍스트이(가) 표시된 사진

자동 생성된 설명

Scene Class

게임이 진행되는 동안 많은 오브젝트들이 생성되고 파괴될 것이다.

이때 운이 좋지 않아 같은 프레임에서 생성 또는 파괴가 되자마자 오브젝트의 갱신(update())을 시도하면 원하는 값에 행동하지 않거나 프로그램에 문제가 생길 수 있을 것이다.

때문에 생성과 파괴를 시도한 프레임이 끝나고 다음 프레임으로 넘어왔을 때부터 게임 오브젝트리스트에 포함시켜 진행해야 한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

creationQueue – 생성할 오브젝트를 담아두는 큐

gameObjects – Scene이 관리하는 오브젝트들

deletionQueue – 삭제할 오브젝트를 담아두는 큐

텍스트이(가) 표시된 사진

자동 생성된 설명

PushDelete() – 삭제할 오브젝트를 오브젝트 리스트에서 찾아 삭제 큐로 넣어주는 함수

텍스트이(가) 표시된 사진

자동 생성된 설명Update()

1. 생성 큐에서 오브젝트를 꺼내 start()를 호출하고 오브젝트 리스트에 추가한다.
2. 오브젝트 리스트에 있는 오브젝트들을 업데이트한다.
3. 삭제 큐에 있는 포인터를 꺼내 리스트에서 오브젝트를 찾고 삭제한다.