

# Code Convention

작성자 : 김은규

# Code

- 더 길더라도 정확한 이름을 사용합니다.
  - 약어를 사용하지 않습니다.
  - 표준인 경우 Acronym을 사용합니다.
  - Ex : UI, IO
- 
- Method name은 동사나 동사구를 사용합니다.
  - Boolean property 앞에 "Is", "Has", "Can"을 붙일 수 있습니다.
  - 여러 속성이 동일한 항목과 관련된 경우 항목 이름을 접두사로 사용하고 속성 유형 또는 역할을 추가합니다.

# Capitalization

- Definitions
- Camel Case : 단어의 첫 문자를 대문자로 표기하는 방식
- Ex) helloWorld, world, myName
- Pascal Case : 맨 처음의 첫 문자는 소문자로 표기, 그 다음 단어는 소문자 표기 방식
- Ex) HelloWorld, World, MyName

# Convention

표기법	사용 위치	세부 사항	예시
PascalCase	Class		AudioManager
	Method		GetValue
	Enum		Title
	Namespace		UOP1.StateMachine
	public/private property		Factory
CamelCase	Local variable		previousValue, mainUI
	Method Parameter		parameterName
underscore	Private field		_inputReader
All-caps 모두 대문자	const	상수는 대문자로 단어 구분을 위해 _을 사용	GRAVITY_AMOUNT

# Programming

- Inspector에 노출 시킬 경우, public 대신 [SerializeField] private
- field가 할당되지 않으면 항상 default value를 가집니다.
- Singleton 사용을 피하도록 합니다.
- 여러 개체에서 액세스 할 수 있는 유사한 중앙 집중식 클래스에 대한 ScriptableObject의 사용법을 살펴봅니다.
- 변수를 선언할 때 var를 사용하지 마세요.
- Static 변수 사용을 피하세요. 만약 필요하다면 static 변수가,
- Fast Enter Play Mode가 호환되는지 확인하십시오.

# Programming

- 하드 코딩된 "magic number"를 사용하지 않습니다.
- ex : `xInput * 0.035f`
- Using을 사용하는 경우 코드 commit하기 전에 모두 제거합니다.
- 코드를 네임스페이스로 묶으면 "UOP1"을 사용하여 이 프로젝트와 관련된 코드 및 assemblies를 나타냅니다.

# Formatting

대괄호가 비어 있으면, 같은 선에 있어야 하며, 비어 있지 않으면 대괄호 안에 정렬되어야 합니다.

```
public class EmptyBraces() { };
```

```
public class NonEmptyBracase
```

```
{
```

```
}
```

# Comments

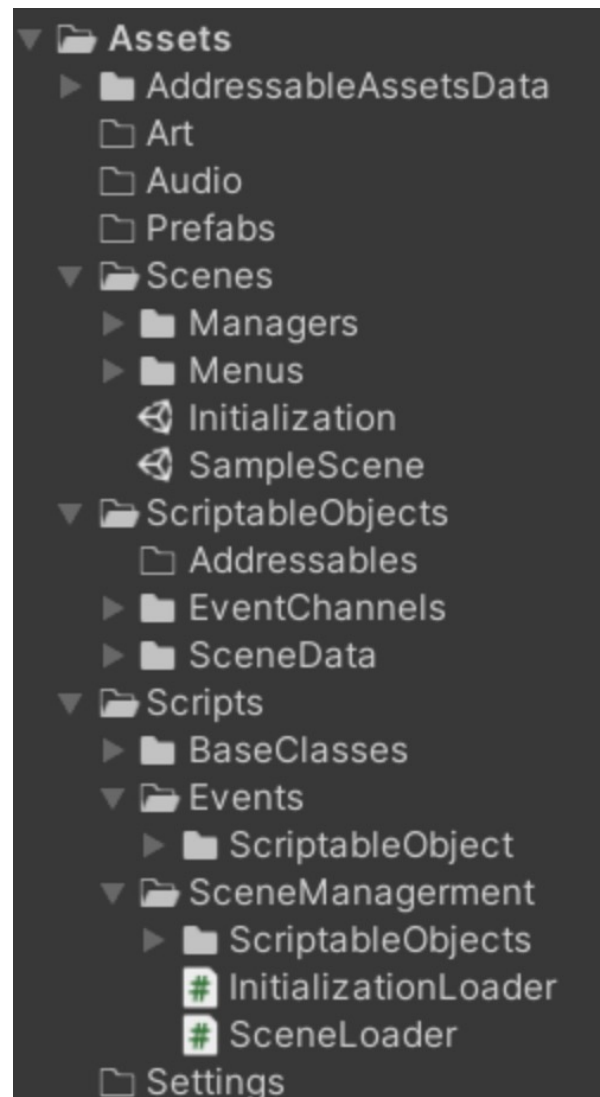
- 만약 누군가 그 코드가 무엇을 하는지 보기만 해도 이해할 수 있다고 생각하면, comment를 달지 마세요.
- 대신 변수, 클래스 및 방법을 명명하여 설명하십시오.
- Summary를 설명하는 요약은 모든 class 위에 쓰세요

```
/// <summary>  
/// This class manages save data  
/// </summary>
```



# Folder

- ScriptableObjects 하위
  - > EventChannel
  - > SceneData
- Scripts
  - > BaseClasses
  - > Events
  - > SceneManagement



# Reference

- <https://docs.google.com/document/d/1-eUWZ0IWREFu5iH-ggofwnixDDQqalOoT4Yc0NpWR3k/edit#heading=h.23fnzd1otnih>