

AssetDatabase



작성자 : 김은규

Changelog

- v0.1 : 초안 작성 (22.04.03)

AssetDatabase

- AssetDatabase는 프로젝트의 에셋에 액세스할 수 있는 API입니다.
- 에셋을 찾아 로드 하는 방법이 있으며, 에셋을 생성, 삭제, 수정할 수 있는 방법 등등이 있습니다.
- Unity Editor AssetDatabase를 사용하여 내부적으로 에셋 파일을 추적하여 에셋 및 그것을 참조하는 개체의 링크를 유지합니다.
- 주의할 점은 AssetDatabase클래스는 Editor상에서만 동작하며 해당 코드를 사용하는 부분은 #if UNITY_EDITOR 조건 컴파일을 걸어야 에러가 나지 않습니다.

파일 탐색

- Unity Editor상에서 모든 에셋 관련 파일 처리를 할 때 사용하는 클래스
- 기본 파일 입출력 부터 Material, Prefab, Animationclip, CustomAsset 등 파일을 만들고 제어 가능

AssetDatabase.FindAssets

[Leave feedback](#)

Declaration

```
public static string[] FindAssets(string filter);
```

Declaration

```
public static string[] FindAssets(string filter, string[] searchInFolders);
```

Parameters

filter	The filter string can contain search data. See below for details about this string.
searchInFolders	The folders where the search will start.

파일 탐색

```
1 using UnityEngine;
2 using UnityEditor;
3 using System;
4
5 public class SearchExample : MonoBehaviour
6 {
7     [MenuItem("Search/GetTexture In ExampleAssets")]
8     public static void GetTexture()
9     {
10         string[] assetGuids = AssetDatabase.FindAssets("t:texture", new[] { "Assets/ExampleAssets" });
11         string[] assetPathList = Array.ConvertAll<string, string>(assetGuids, AssetDatabase.GUIDToAssetPath);
12
13         foreach(string assetPath in assetPathList)
14         {
15             Texture texture = AssetDatabase.LoadAssetAtPath(assetPath, typeof(Texture)) as Texture;
16             Debug.Log("에셋 이름 : " + GetAssetName(assetPath));
17             Debug.Log("에셋 경로 : " + GetAssetPath(assetPath));
18         }
19     }
20
21     // Asset path를 넣으면 name 반환
22     public static string GetAssetName(string assetPath)
23     {
24         return assetPath.Substring(assetPath.LastIndexOf("/") + 1);
25     }
26
27     // Asset path를 넣으면 경로만 반환 name 제외
28     public static string GetAssetPath(string assetPath)
29     {
30         return assetPath.Substring(0, assetPath.LastIndexOf("/"));
31     }
32 }
33
```

- FindAsset
- 프로젝트의 Assets폴더안에 있는 에셋파일을 찾아서 가져오는 함수입니다.
- 찾은 에셋의 GUID를 string 배열로 반환합니다.

MenuItem

1. 메뉴아이템 추가

가장 기본적인 메뉴아이템 추가 방법이다.

```
[MenuItem("MyMenu/Do Something")]  
static void DoSomething()  
{  
    Debug.Log("Doing Something...");  
}
```

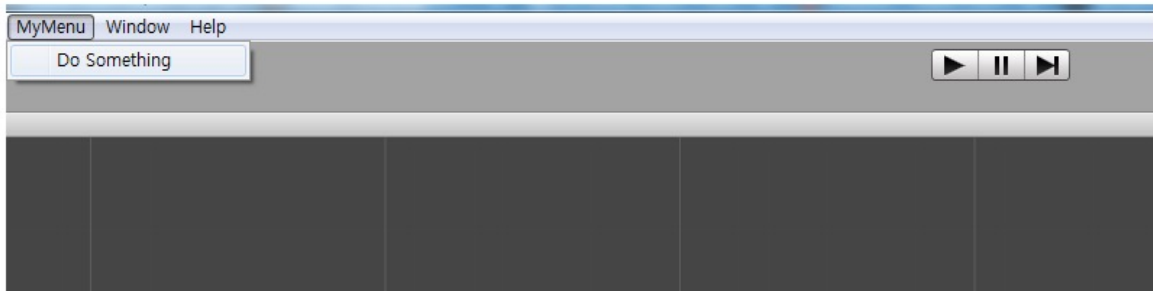


Fig1. - 클릭하면 DoSomething() 함수가 실행된다.

- MenuItem
- 상단에 메뉴를 추가 하는 방법

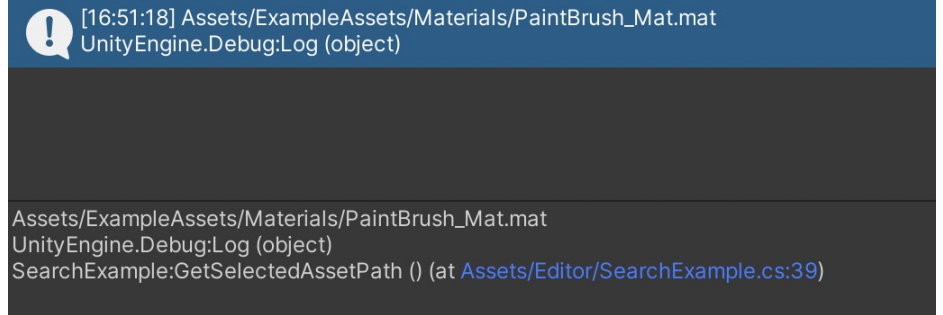
MenuItem

- itemName으로 메뉴 이름 결정

```
8 namespace UnityEditor
9 {
10     ... public sealed class MenuItem : Attribute
17     {
18         public string menuItem;
19         public bool validate;
20         public int priority;
21
22         ... public MenuItem(string itemName);
39         ... public MenuItem(string itemName, bool isValidFunction);
56         ... public MenuItem(string itemName, bool isValidFunction, int priority);
73     }
74 }
```

MenuItem

itemName으로 메뉴 이름
결정



```
// 선택된 Asset의 Path를 가져옵니다.  
[MenuItem("Search/Get Select Asset Path")]  
public static void GetSelectedAssetPath()  
{  
    UnityEngine.Object selectObj = Selection.activeObject;  
    string assetPath = AssetDatabase.GetAssetPath(selectObj.GetInstanceID());  
    Debug.Log(assetPath);  
}
```


에셋 생성

```
[MenuItem("Tools/CreateMaterialFile")]
public static void CreateMaterialFile()
{
    UnityEngine.Object selectObj = Selection.activeObject;
    bool isFolder = IsFolder(selectObj);
    if(!isFolder)
    {
        Debug.LogError("폴더를 선택하세요.");
        return;
    }

    string folderPath = AssetDatabase.GetAssetPath(selectObj.GetInstanceID());
    string assetPath = string.Format(folderPath + "/{0}.mat", "SampleMaterial");

    bool isExist = IsExist(assetPath);

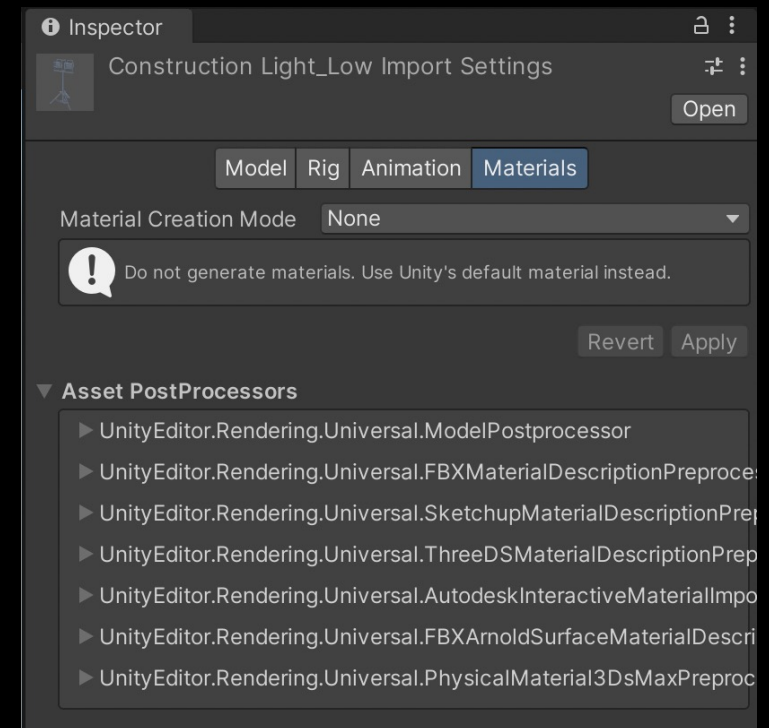
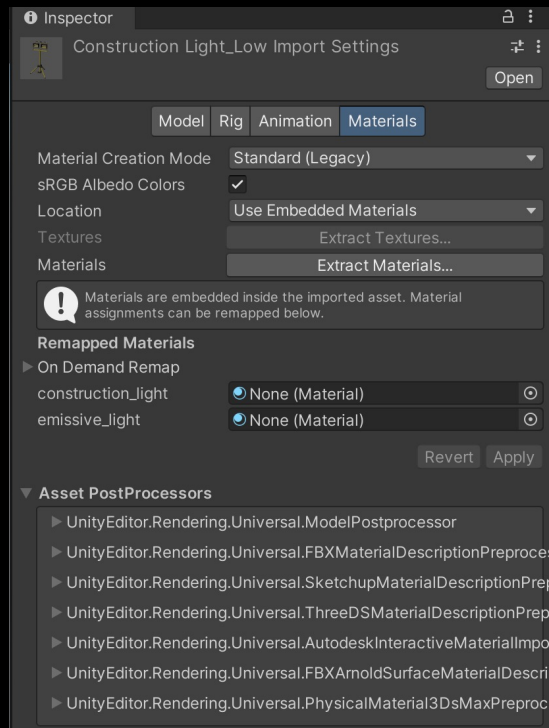
    if(isExist)
    {
        Debug.LogWarning("File is Exist");
        return;
    }

    Material mat = new Material(Shader.Find("Universal Render Pipeline/Lit"));
    AssetDatabase.CreateAsset(mat, assetPath);
    AssetDatabase.SaveAssets();
    AssetDatabase.Refresh();
}
```

에셋 수정

```
[MenuItem("Tools/EditModelFile")]
public static void EditModelFile()
{
    UnityEngine.Object selectObj = Selection.activeObject;
    string assetPath = AssetDatabase.GetAssetPath(selectObj.GetInstanceID());

    string assetName = GetAssetName(assetPath);
    if(assetName.Contains(".fbx") || assetName.Contains("FBX"))
    {
        ModelImporter importer = AssetImporter.GetAtPath(assetPath) as ModelImporter;
        importer.materialImportMode = ModelImporterMaterialImportMode.None;
        importer.SaveAndReimport();
        EditorUtility.SetDirty(selectObj);
        AssetDatabase.Refresh();
    }
    else
    {
        Debug.LogError("It is not FBX");
    }
}
```



Reference

- <https://docs.unity3d.com/kr/530/Manual/AssetDatabase.html>
- (P3)
- <https://darkcatgame.tistory.com/95>
- (P3)
- <https://darkcatgame.tistory.com/95>
- (P3 ~ 10)