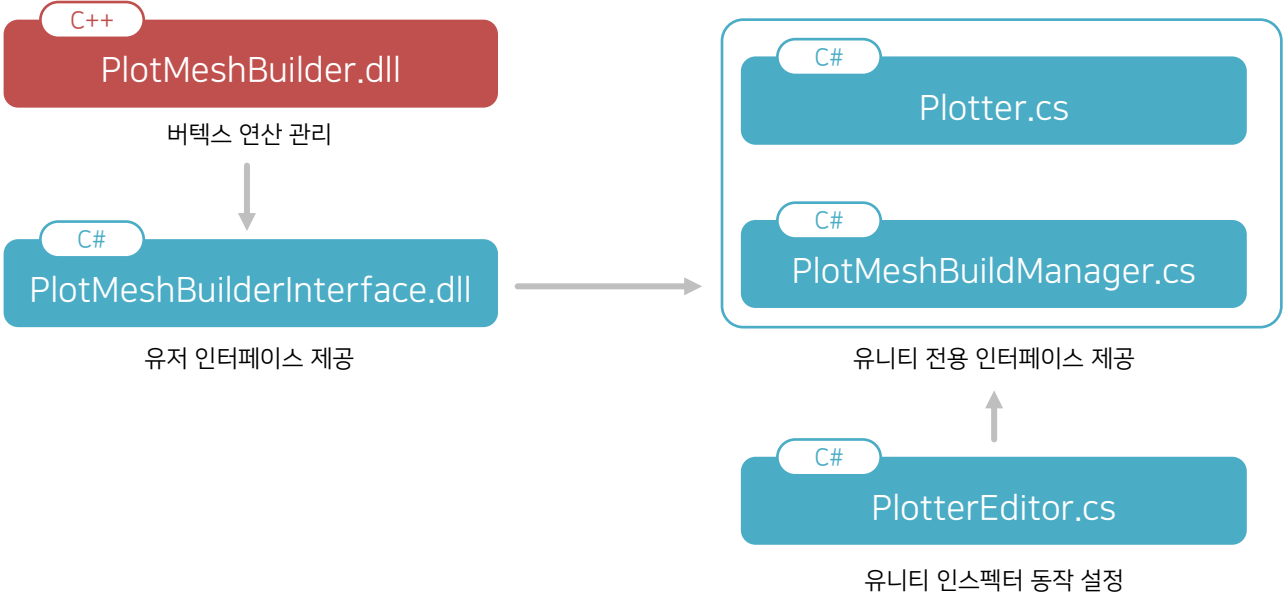


UnityUI - Plotter 사용 가이드

지원하는 프로그래밍 언어

- C++ } (Plotter.cs, PlotMeshBuildManager.cs 제외)
- C# }
- C# - 유니티

참조 체계



- PlotMeshBuilder.dll 단독 사용 시 C++에서 사용 가능
- PlotMeshBuilder.dll + PlotMeshBuilderInterface.dll 사용 시 C#에서 사용 가능

사용 가이드 (Unity)

필수 파일 목록

- PlotMeshBuilder.dll
- PlotMeshBuilderInterface.dll
- Plotter.cs
- Editor/PlotterEditor.cs
- PlotMeshBuildManager.cs

Serial Plugin 사용 가이드

Class

Plotter

- Monobehaviour : 게임오브젝트의 컴포넌트로 할당하여 사용
- 대부분의 변수는 인스펙터를 통해 조절 가능

Variable

int orderInLayer

- 렌더링 순서 결정
- Plotter가 다른 오브젝트에 가려서 출력될 경우 값을 조절

GameObject rotationHandle

- 3D 회전을 위한 핸들
- 이곳에 할당한 오브젝트를 회전시킬 경우, Plotter의 시점이 회전함

Vector2 xLimit

- x축 최소값, 최대값 설정

Vector2 yLimit

- y축 최소값, 최대값 설정

Vector2 zLimit

- z축 최소값, 최대값 설정

Vector3 origin

- Plotter의 원점 설정

bool useAxisLine

- 그래프 axis의 출력 여부 결정

If (useAxisLine)

float axisLineThickness

- 그래프 axis의 두께 설정

Color axisLineColor

- 그래프 axis의 색 설정

Serial Plugin 사용 가이드

Class

Plotter

- Monobehaviour : 게임오브젝트의 컴포넌트로 할당하여 사용
- 대부분의 변수는 인스펙터를 통해 조절 가능

Variable

bool useAxisTick

- 그래프 axis의 눈금 출력 여부 결정

If (useAxisTick)

- 눈금의 색은 axis의 색과 동일하게 설정됨

float tickLength

- 그래프 axis 눈금의 길이 설정

float tickThickness

- 그래프 axis 눈금의 두께 설정

Color backgroundColor

- 배경색 설정

Color[] plotColor

- Mesh3D, Surface3D를 제외한 플롯 메서드의 경우, 색상을 지정하지 않으면 여기서 지정된 색상을 순서대로 사용
plotColor에 설정된 색이 없을 시 하얀색 사용

ex

```
plotter.plotColor = new Color[] { Color.red, Color.green };

plotter.Plot(new float[] { 1f, 2f, 3f });
plotter.Plot(new float[] { 4f, 5f, 6f });
plotter.Plot(new float[] { 7f, 8f, 9f });
plotter.Draw();
// >> 1, 2, 3 값을 출력하는 선은 plotColor의 첫 번째 색깔로 설정됨 (예시에서는 빨간색)
// >> 4, 5, 6 값을 출력하는 선은 plotColor의 두 번째 색깔로 설정됨 (예시에서는 초록색)
// >> 7, 8, 9 값을 출력하는 선은 plotColor의 첫 번째 색깔로 설정됨 (예시에서는 빨간색)
// >> 예시에서는 plotColor에 지정된 색이 두 개 밖에 없으므로, 세 번째 부터는 다시 첫
//     색상부터 사용
```

Serial Plugin 사용 가이드

Class

SerialHandle

Method

void Plot(float[] x, float[] y, float lineWidth, Color lineColor)

- 주어진 x, y 값을 잇는 선 생성
- lineWidth, lineColor는 설정하지 않을 시 각각 1f, plotter의 plotColors로 설정됨

void Plot(float[] y, float lineWidth, Color lineColor)

- 주어진 y 값을 잇는 선 생성, x 값은 자동 생성
- 나머지는 Plot 메서드와 동일

void Plot3D(Vector3 startPoint, Vector3 endPoint, float lineWidth, Color lineColor)

- 주어진 start point와 end point를 잇는 선 생성
- lineWidth, lineColor는 설정하지 않을 시 각각 1f, plotter의 plotColors로 설정됨

void Plot3D(Vector3[] startPoints, Vector3[] endPoints, float lineWidth, Color lineColor)

- 주어진 start point와 end point를 각각 잇는 선 생성
- lineWidth, lineColor는 설정하지 않을 시 각각 1f, plotter의 plotColors로 설정됨

void Scatter(float[] x, float[] y, float scatterRadius, Color scatterColor, int scatterMeshSegment)

- 주어진 x, y 값의 위치에 맞춰 점 생성
- scatterRadius, scatterColor는 설정하지 않을 시 각각 1f, plotter의 plotColors로 설정됨
- satterMeshSegment는 최소 3, 최대 20의 값을 받는 변수, 3에 가까울 수록 삼각형, 20에 가까울 수록 원이 됨. 값이 높을 수록 컴퓨팅 시간 증가, 따로 설정하지 않을 시 12로 설정됨

void Scatter(float[] y, float scatterRadius, Color scatterColor, int scatterMeshSegment)

- 주어진 y 값의 위치에 맞춰 점 생성, x 값은 자동 생성
- 나머지는 Scatter 메서드와 동일

void Scatter3D(Vector3[] scatterPoints, float scatterRadius, Color scatterColor, int scatterMeshSegment)

- 주어진 scatter point에 맞춰 점 생성
- 나머지는 Scatter 메서드와 동일

Serial Plugin 사용 가이드

Class

SerialHandle

Method

void Bar(float[] x, float[] y, float width, Color color)

- 주어진 x, y값에 따라 Bar 그래프 생성
- width, color는 설정하지 않을 시 각각 1f, plotter의 plotColors로 설정됨

void Bar(float[] y, float width, Color color)

- 주어진 y 값을 잇는 선 생성, x 값은 자동 생성
- 나머지는 Bar 메서드와 동일

void Barh(float[] x, float[] y, float width, Color color)

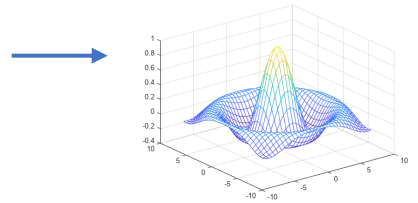
- 주어진 x, y값에 따라 Barh 그래프 (수평 Bar 그래프) 생성
- width, color는 설정하지 않을 시 각각 1f, plotter의 plotColors로 설정됨

void Barh(float[] y, float width, Color color)

- 주어진 y값에 따라 Barh 그래프 생성, x 값은 자동 생성
- 나머지는 Barh 메서드와 동일

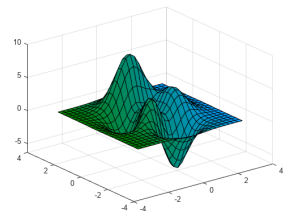
void Mesh3D(float[,] xMap, float[,] yMap, float[,] zMap, Gradient mapColor, float lineWidth)

- 주어진 x, y, z map 값에 따라 Mesh3D 그래프 생성
- mapColor로 색상 조절 가능
- lineWidth는 설정하지 않을 시 1f로 설정됨



void Surface3D(float[,] xMap, float[,] yMap, float[,] zMap, Gradient mapColor)

- 주어진 x, y, z map 값에 따라 Surface3D 그래프 생성
- mapColor로 색상 조절 가능
- 매트랩과 다르게 경계선은 출력되지 않음 (경계선 출력 필요 시 Mesh3D와 함께 사용)



Serial Plugin 사용 가이드

Class

SerialHandle

Method

void Draw()

- 누적된 Plot 명령을 기반으로 그래픽 생성
- Draw 혹은 Clear 명령 없이 계속해서 Plot 명령 생성 금지

ex

```
plotter.Plot(...);
plotter.Plot3D(...);
plotter.Surface3D(...);

plotter.Draw();
// >> 주어진 값에 따라 Plot, Plot3D, Surface3D를 출력
```

ex

```
while (true)
{
    plotter.Plot(...);
    plotter.Plot3D(...);
    plotter.Surface3D(...);
}
// >> 위와 같이 Draw 혹은 Clear 메서드 호출 없이 지속적인 Plot 데이터 누적 금지
```

void Clear()

- 누적된 Plot 명령 및 그래픽 초기화

ex

```
private void Start()
{
    plotter.Plot(...);
    plotter.Plot3D(...);
    plotter.Surface3D(...);

    plotter.Draw();
}

private void OnDestroy()
{
    plotter.Clear();
}
// >> 씬 종료 혹은 오브젝트 파괴 시 plotter 데이터 및 그래픽 초기화
```