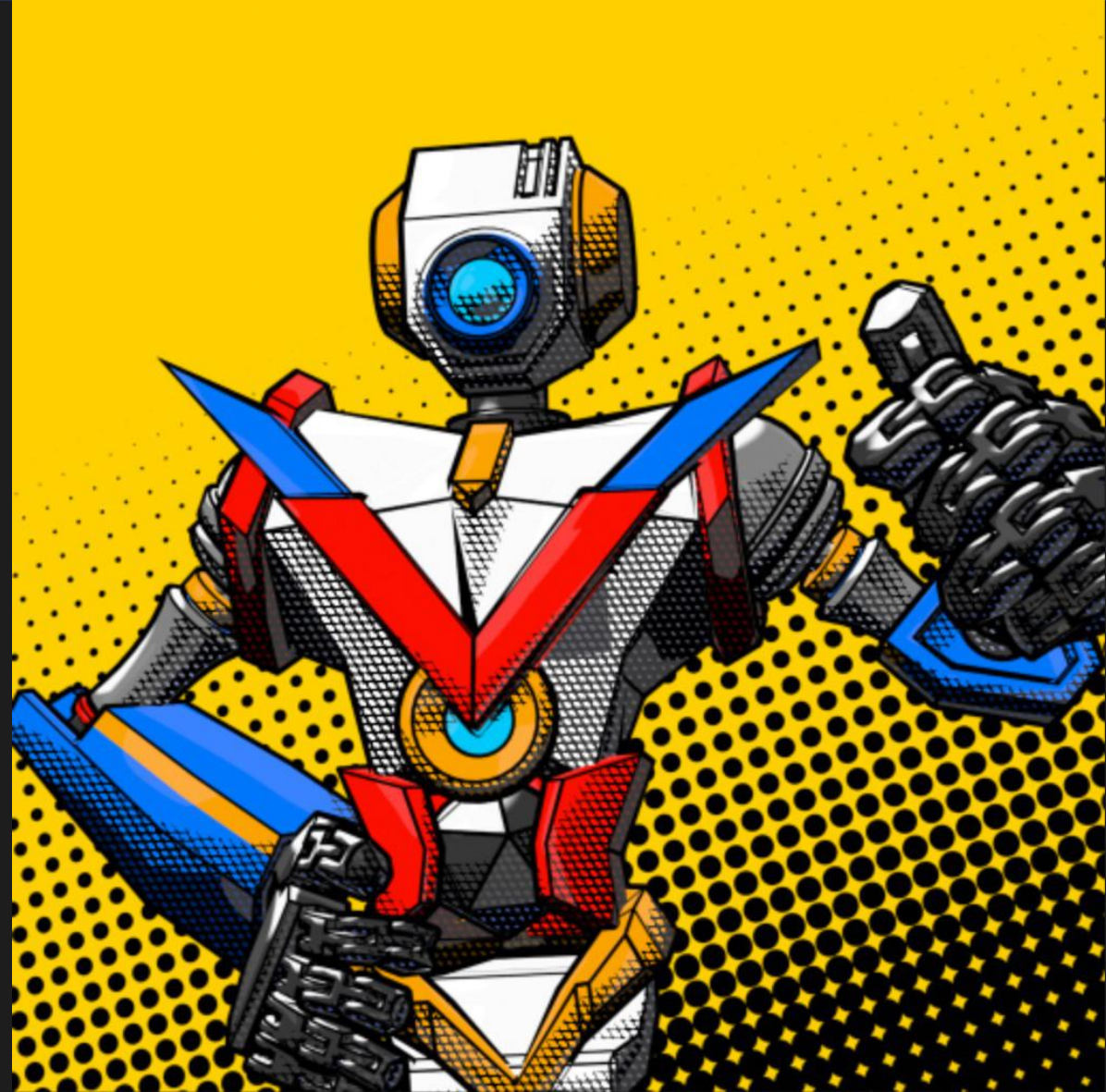


## 026강 NPR : Non-Photorealistic Rendering

카툰 렌더링  
Eevee에서 실사가 아닌 재질을 만들기



# NPR : Non-Photorealistic Rendering

## 'Stylized'

NPR은 카툰, 일러스트레이션과 같은 스타일을 지향하는 렌더링 방식입니다.

NPR은 그 특성상 추구하는 형태나 접근 방식이 무궁무진하므로 여기서 모두 다룰수는 없지만, 기본적인 접근법과 주로 사용하는 노드 중심으로 살펴보겠습니다.



# 셀 셰이딩과 음영 단계

효과적인 명도 표현을 위해서



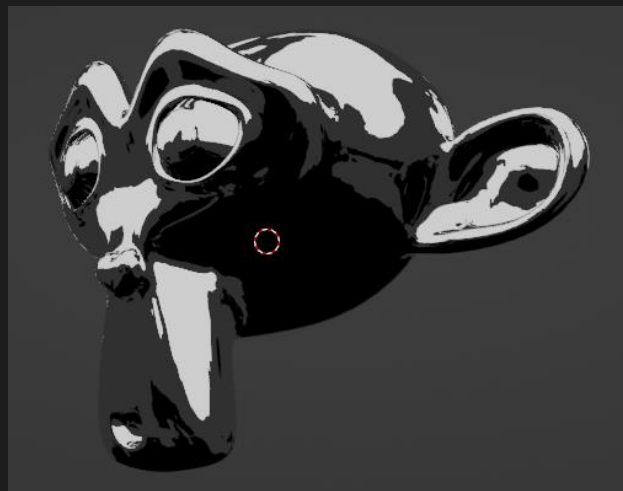
가장 흔한 연결은 Shader to RGB – ColorRamp 조합입니다.

그런데 이 연결은, 경우에 따라 왼쪽처럼 울퉁불퉁하게 표현될 수 있습니다..

물론 이런 이미지를 의도할 수도 있겠지만

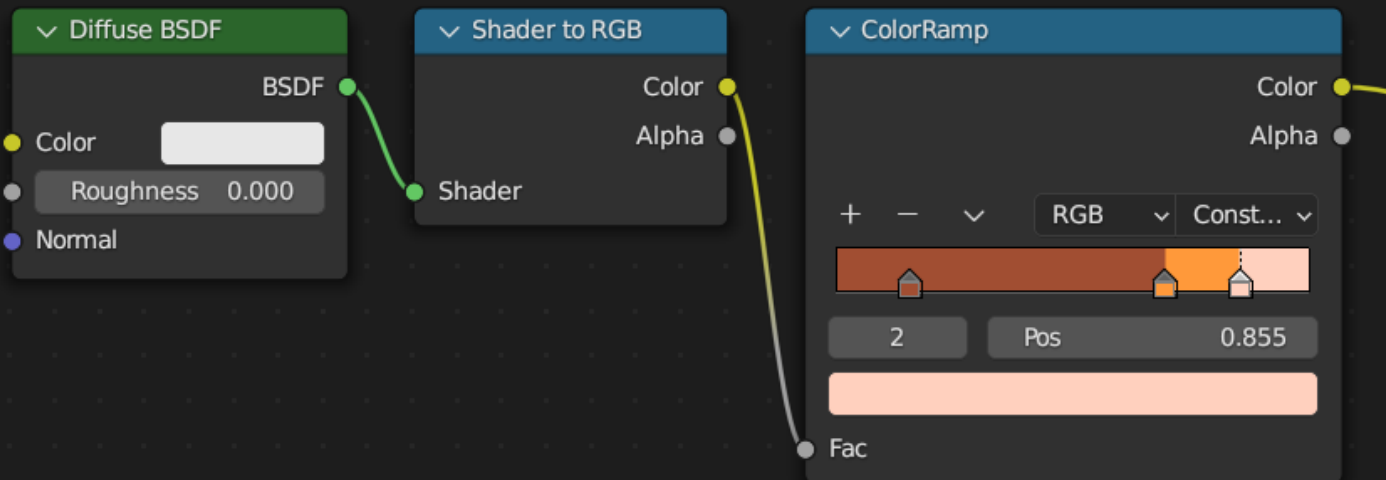
보통 셀 셰이딩은 경계가 **물체의 형태를 따라 만들어질 때**

더 효과적입니다.



# 셀 셰이딩과 음영 단계

효과적인 명도 표현을 위해서



ColorRamp를 통한 색 구분은 날카로운 편이 좋습니다.

점진적으로 부드럽게 증가하는 경우,  
마치 금속과 같은 느낌으로 번들거립니다.

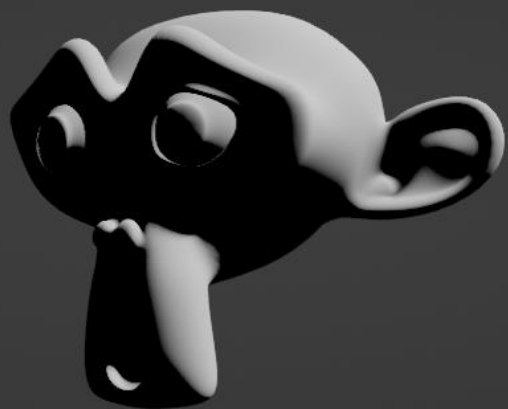
그러나 좌측처럼 단계를 단순화시키면 표면의 디테일이 제대로  
드러나지 않아 답답한 느낌이 납니다.

NPR에서는 이런 부족한 디테일을 색 구분, 외곽선, 텍스처로  
메꾸게 됩니다.



# 가짜 음영

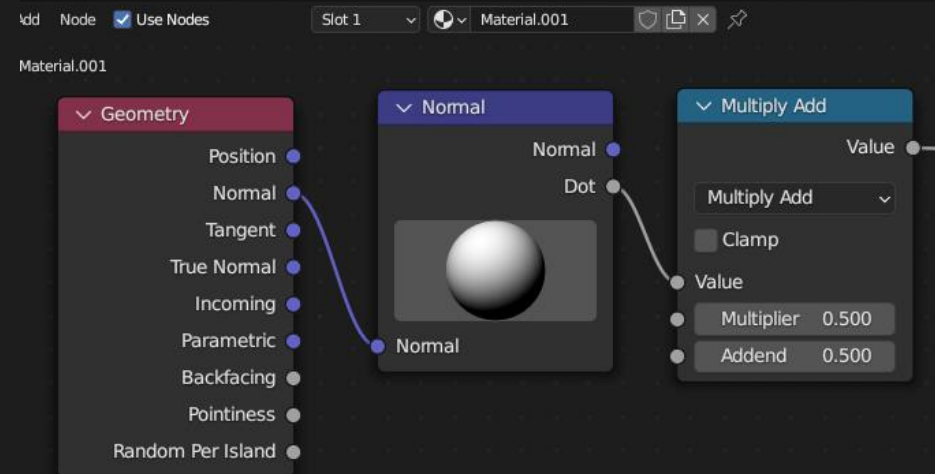
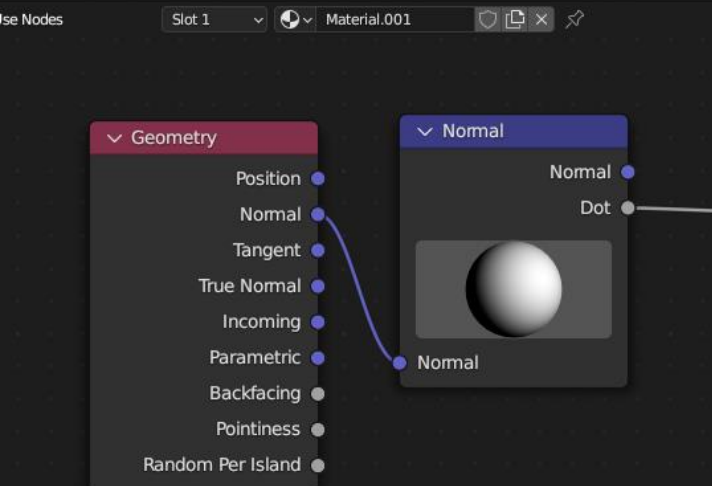
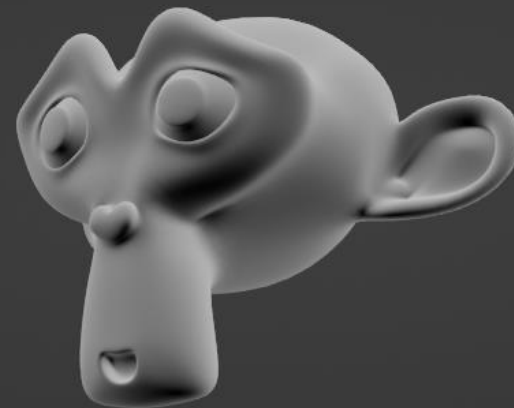
## Normal node



라이팅 효과가 아닌 물체 표면의 각도만 필요하다면 13강에서처럼 Normal 노드를 이용할 수 있습니다.

Dot출력은 -1에서 1의 범위를 갖기 때문에, 우측처럼 범위를 0에서 1로 변경하면 표면 전체에 걸쳐 부드러운 명도 단계를 얻습니다.

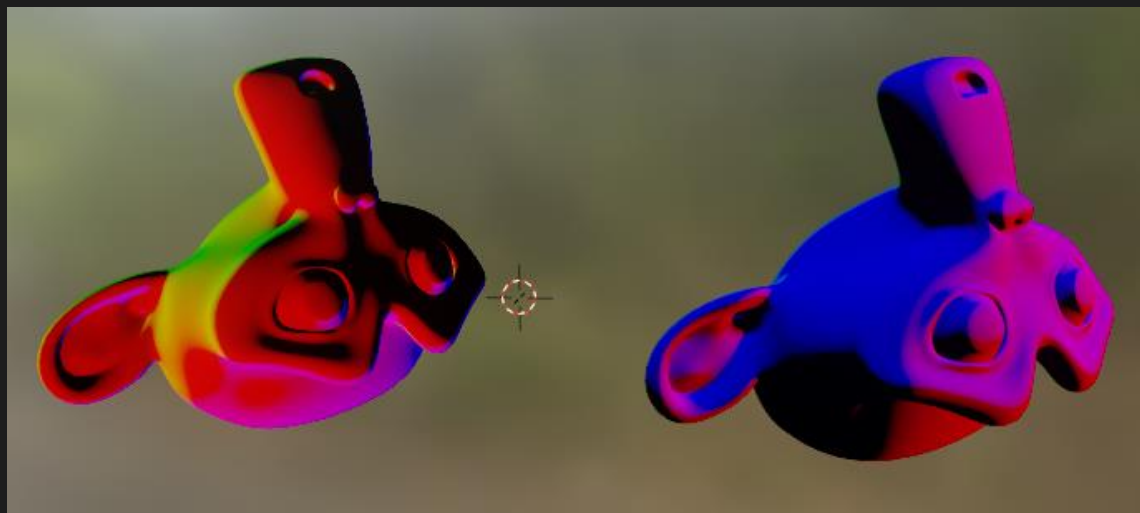
이것을 **컬러램프**에 연결하여 모든 영역을 컨트롤할 수 있습니다.



여담입니다만 2000년대 초반 게임에서 사용되던 방식 중 하나입니다.



# 두 종류의 Normal

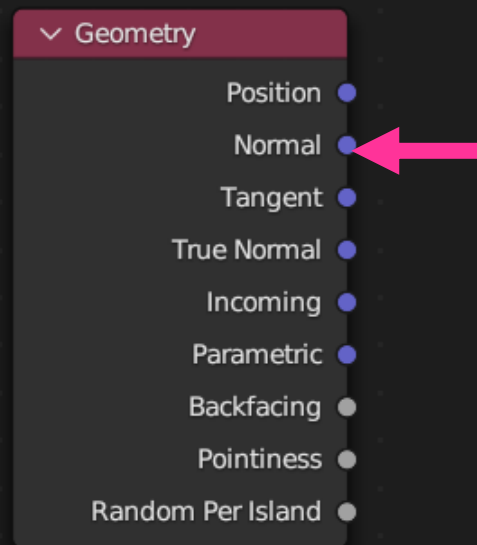
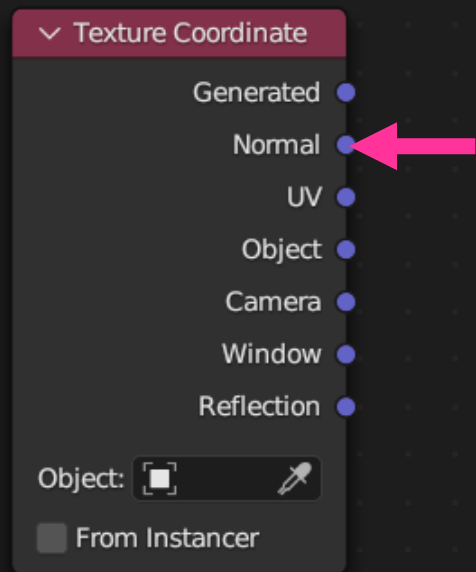


Texture Coordinate 의 Normal은 Object Space를 따라갑니다.  
즉, 오브젝트가 회전했을 때 노멀도 같이 회전합니다.

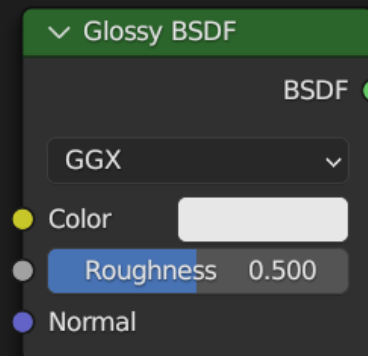
한편, Geometry의 Normal은 World Space에 속하므로  
오브젝트가 회전해도 노멀은 방향을 유지합니다.

Texture Coordinate의 Normal은 오브젝트를 따라  
같이 움직이고 싶을 때 사용하고,

Geometry의 Normal은 오브젝트가 회전해도  
계속 같은 방향을 가리키고 싶을 때 사용합니다.



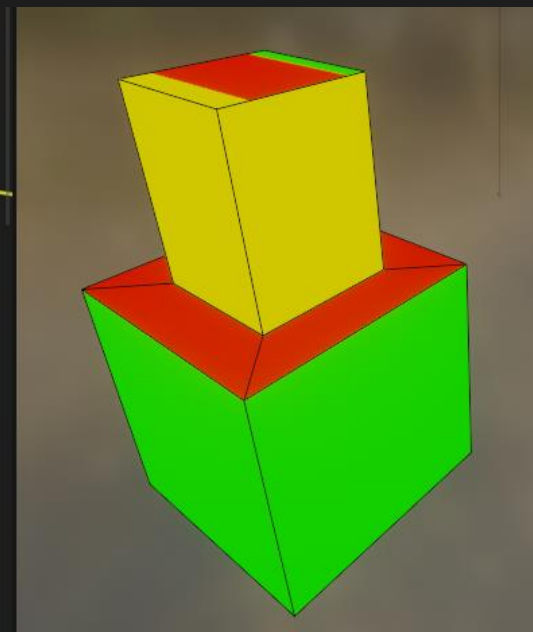
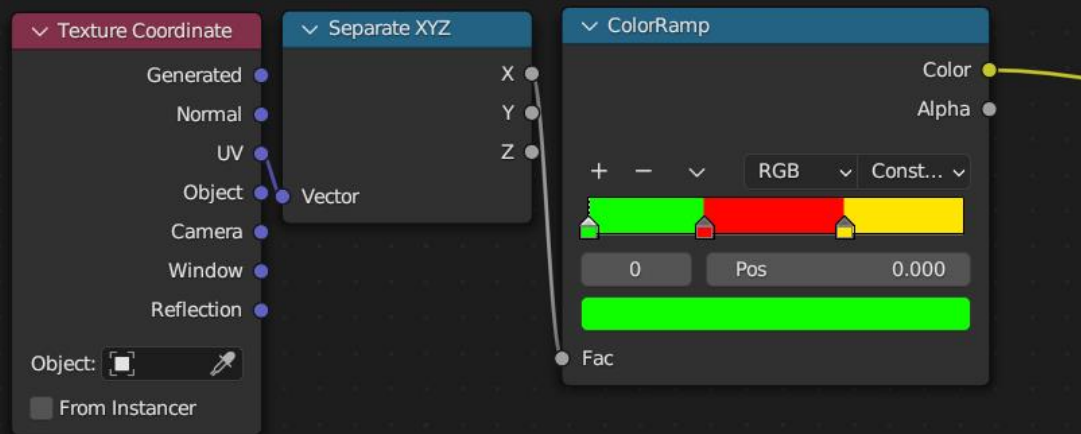
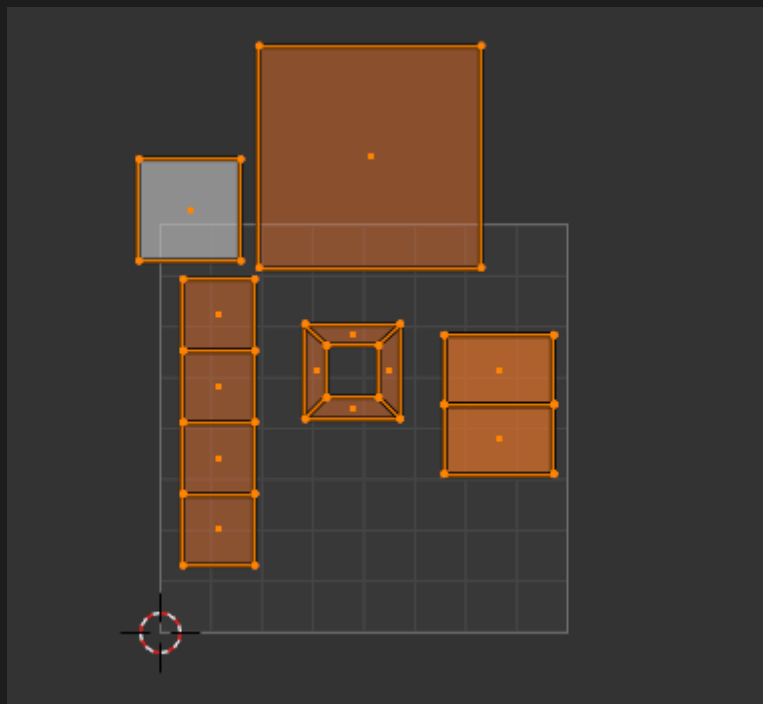
※ Shader의 Normal은  
World Space Normal입니다.



## 색 구분

## 하나의 머티리얼에서 색을 구분시키기

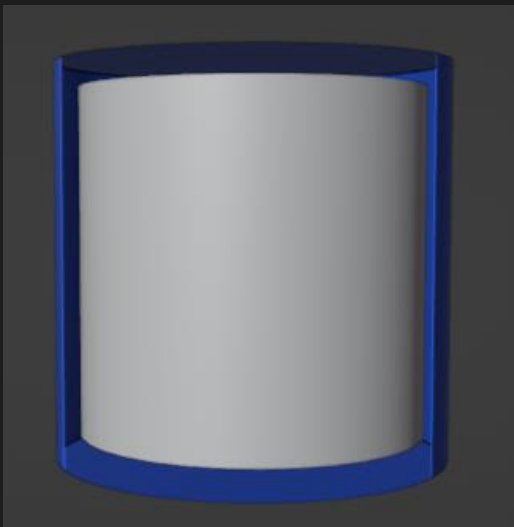
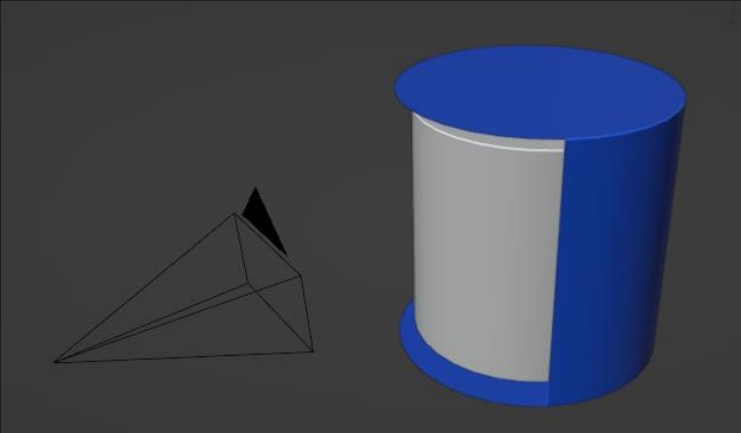
UV좌표를 이용하면 이미지없이 표면의 색을 여러 개로 구분할 수 있습니다.  
아래와 같이 연결하시고 각 면의 UV위치를 조절하세요.



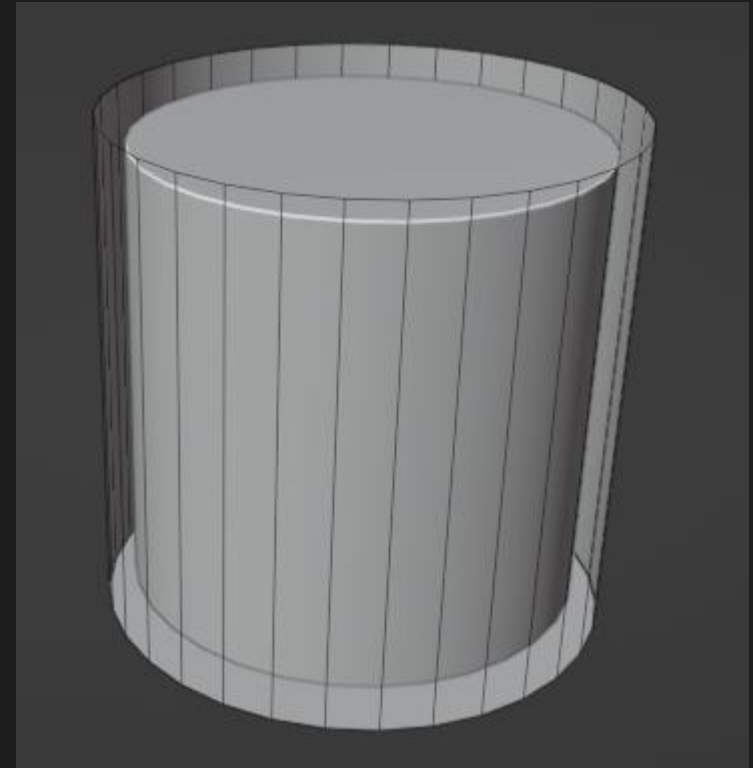
만약 UV를 이미 다른곳에서 사용하신다면 UV를 두개를 만드세요. (13강 참고)

# 외곽선(1) Backface Culling

1. 만약 오브젝트를 감싸는 좀 더 큰 오브젝트를 만들어서, 카메라 방향으로 구멍을 뚫는다면 마치 외곽선처럼 보일 것입니다.



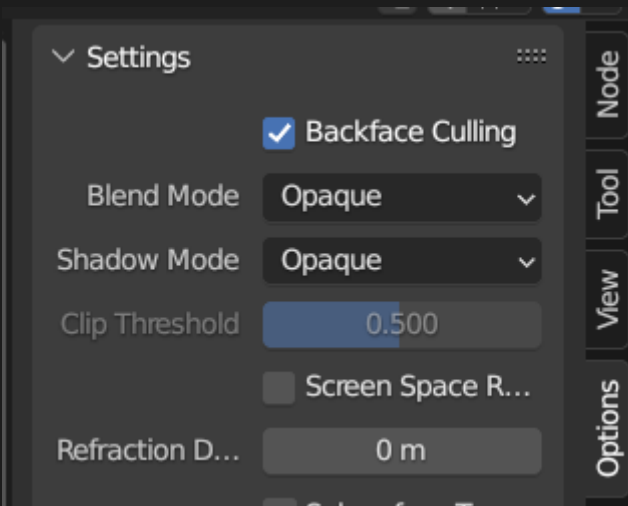
2. 한편, Backface Culling은 물체의 뒷면을 투명하게 만드는 기능입니다. 노멀 방향을 반대로 뒤집는다면, Backface Culling은 거꾸로 작용하여 물체의 앞면을 투명하게 만들어, 결과적으로 아래와 같이 물체의 뒷면만 보이게 만들 것입니다. 이것이 '자동으로 카메라 방향으로 구멍을 뚫어주는' 효과가 됩니다.



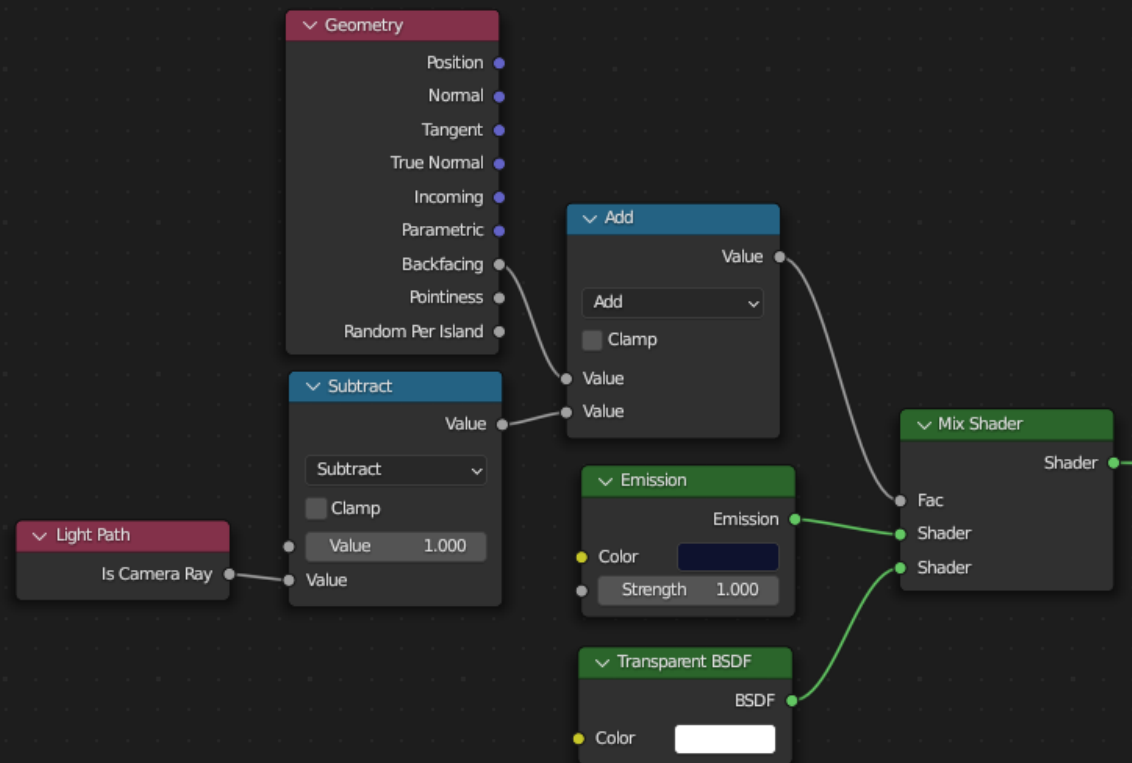


# 외곽선(1) Backface Culling

Eevee에서는 렌더 옵션에서 Backface Culling을 켜주면 작동합니다.

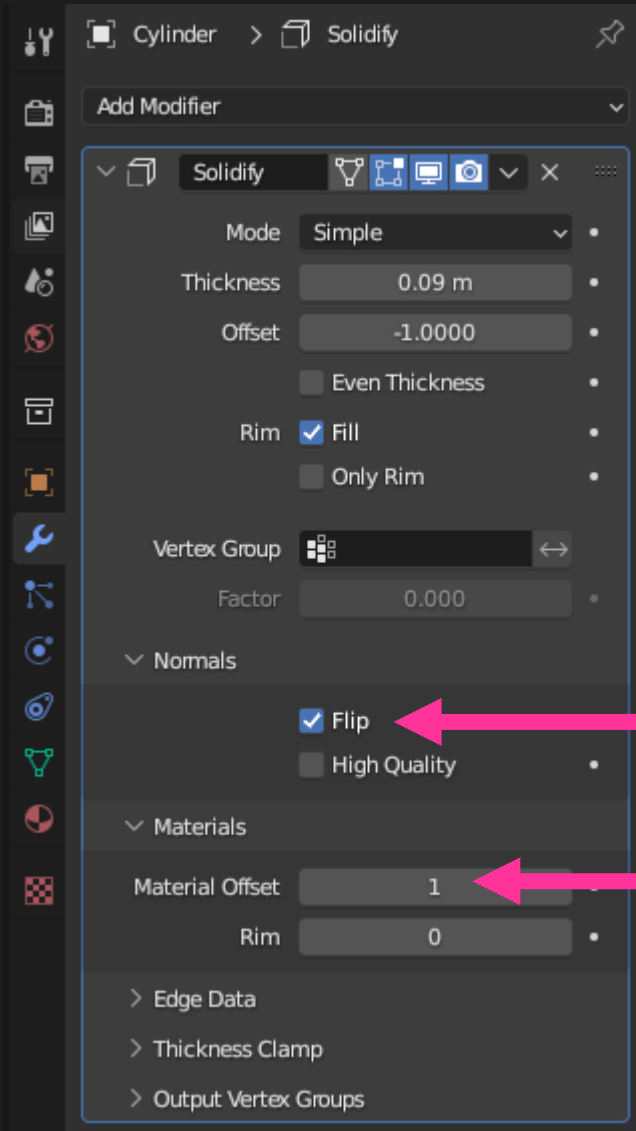


Cycles에서는 조금 복잡해집니다.. (자세한 사항은 28강 참고)  
하지만 NPR은 보통 Cycles에서 구현하지 않습니다.



# 외곽선(1) Backface Culling

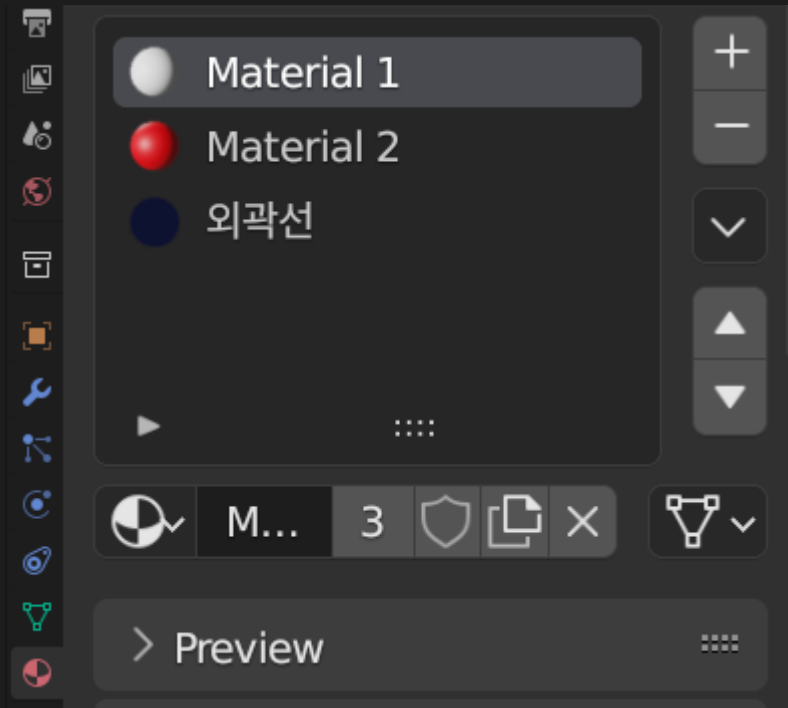
오브젝트를 감싼 외곽선이 **자동으로** 만들어지게 하려면 Solidify를 이용합니다.



Flip으로 바깥의 면을 뒤집습니다.

Material Offset으로 Material을 선택합니다. (뒷장 참고)

# 외곽선(1) Backface Culling



Solidiy의 Material Offset은 조금 특이한 방식으로 재질을 선택합니다.

Offset 이라는 이름처럼, 원래 재질에서 n번째 떨어진 재질을 선택합니다.

앞에서 Material Offset이 1이었으므로,

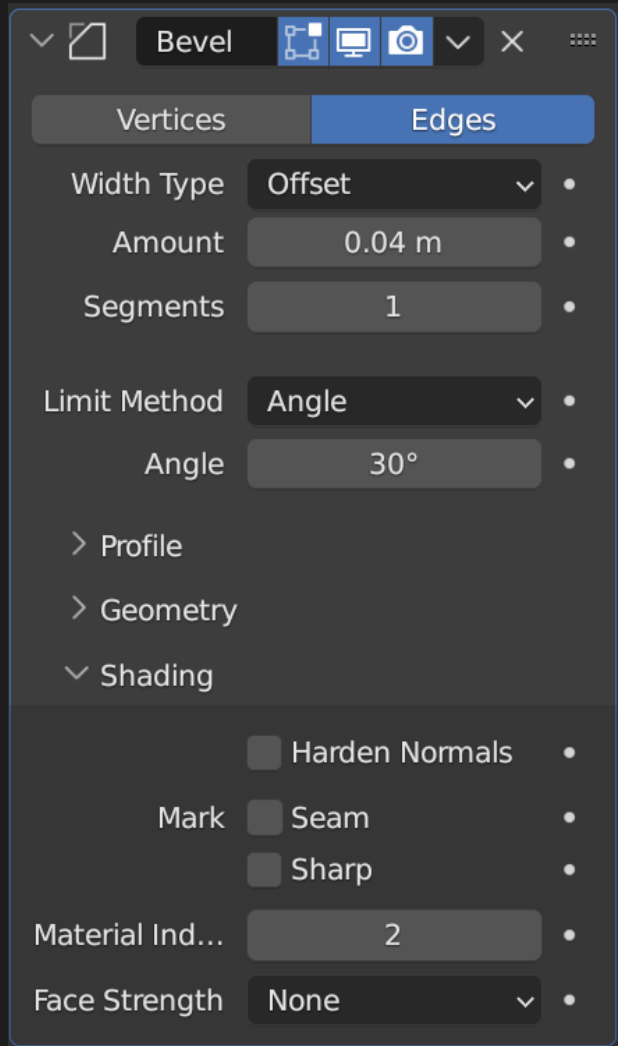
Material 1이었던 재질은 한칸 아래인 **Material 2**,

**Material 2**였던 재질은 그 한칸 아래인 **외곽선**이 됩니다.

혼란을 막기 위해 **외곽선** 재질을 가장 마지막에 놓고,

**Material offset**을 매우 큰 숫자 (예컨대 1000) 으로 두면 반드시 맨 아래 재질이 선택되므로 편리합니다.

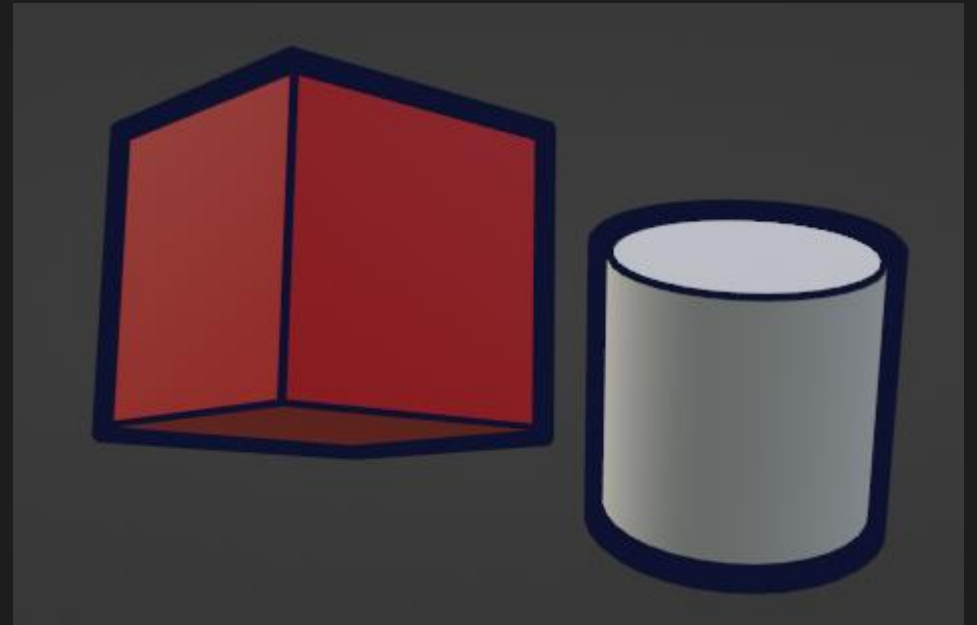
## 외곽선(2) Bevel



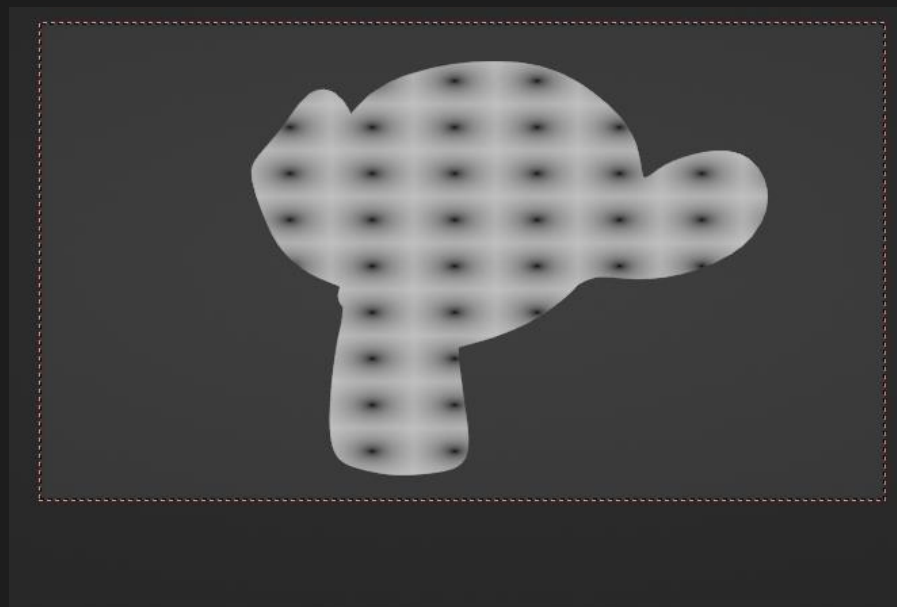
Bevel모디파이어를 사용하여 추가적으로 외곽선을 만들 수 있습니다.

Bevel은 Solidify와 달리 Material 'index'를 사용합니다. 이는 Material 슬롯의 재질을 순서대로 불러옵니다.

방식은 다르지만 이 역시 매우 큰 값으로 두면 반드시 마지막 재질을 선택하므로, 이렇게 Solidify와 같은 재질을 선택하게 만들 수 있습니다.



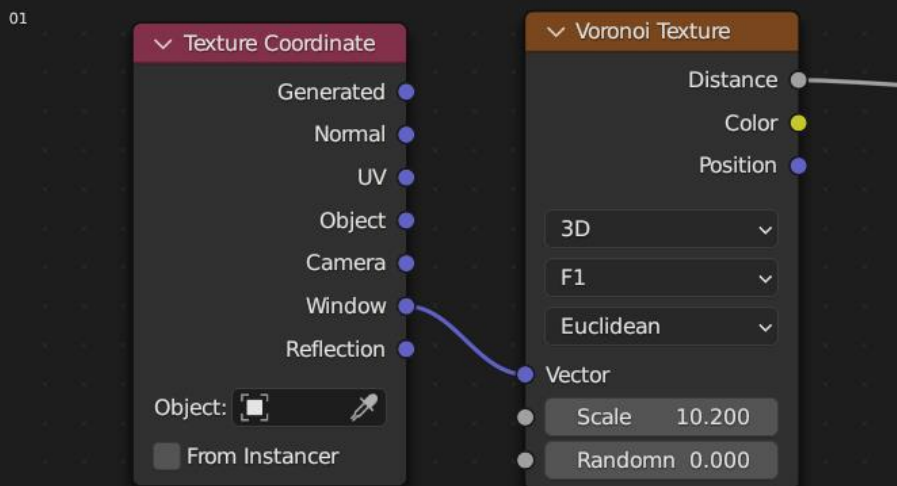
# Window 좌표



Window 좌표는 현재 보고 있는 화면을 나타냅니다.  
카메라 모드에서 카메라 왼쪽 아래부터 오른쪽 위 끝까지를  $x,y$ 축에 대응시킵니다.  
뷰포트 상태라면 '현재 뷰포트의 끝부터 끝까지'가 좌표가 됩니다.  
(뷰포트 크기에 따라 좌표가 바뀝니다!)

혼란을 막기 위해 window좌표를 쓸 때는 가급적 카메라 모드를 사용하며, 해상도 비율에 따라 미리 스케일을 조절해 둡니다.

01 Use Nodes Slot 1 Material.001





# 가짜 광택

## Reflection 좌표



Reflection 좌표는 화면이 움직임에 따라 변화하는 반사를 나타낼 수 있습니다.

Reflection좌표의 원리는 '배경 좌표를 반사한 것' 이므로, 만약 이미지 텍스처를 꺾는다면 배경으로 쓰일 수 있는 이미지여야 합니다.

즉 Hdri 같은 배경 텍스처를 Environment Texture 노드에 꺾어 사용합니다.

