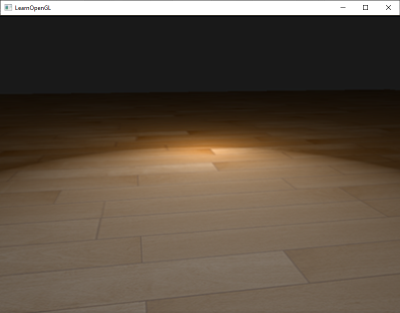
Advanced Lighting // Advanced Lighting

-조명을 공부할 때, Phong 모델을 사용해서 기본적인 빛 표현을 배웠음

Blinn-Phong

-Phong는 특정 조건(반사율이 낮은 표면이 거친)의 shininess가 낮은 영역인 경우에는 정반사가 고장이 남

-shininess가 1.0, 평평만 질감의 평면에서

-빛의 가장자리에서 반사 영역이 바로 끝나는 모습이 보임

-view 벡터와 반사 벡터 사이의 각도가 90도가 넘어가서 나타나는 현상, 이는 내적이 음수가 되고 반사 지수가 0이 되기 때문임

-view벡터와 반사 벡터 사이의 각도가 90도 보다 크면 반사 지수가 0이 되어 90도보다 큰 값의 빛은 보이지 않는 것이 맞는가? // 틀린 이야기임

-normal이랑 광원 사이의 각도가 90도 보다 높으면 광원이 표면보다 아래에 있음을 의미함, 따라서 light diffuse값이 0.0이어야 함, 그러나 우리는 광원과 normal 사이의 각도를 측정하는 것이 아니라 view와 반사 벡터 사이의 각을 측정하기에 오류가 발생함



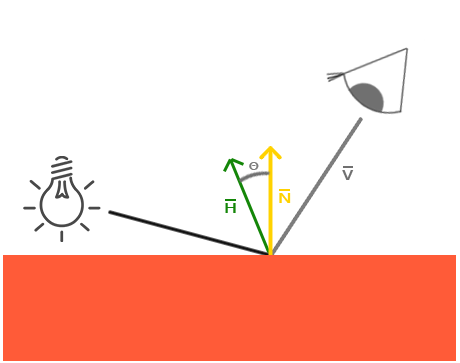
-왼쪽 이미지는 세타 각도가 90도 미만인 Phong반사를 보여줌, 오른쪽 이미지에서는 세타 각도가 90도가 넘는 것을 볼 수 있음

-세타 각도가 90도가 넘어 specular를 무시할 수 있음

-이는 보통 view방향이 반사 방향보다 멀 경우에는 문제가 되지 않으나, 낮은 반사 지수를 사용하면 충분히 영향을 받을 수 있음

-Blinn-Phong모델은 거의 비슷하나, specular에 약간 다른 접근법을 가짐

-반사 벡터를 사용하는 대신, view방향과 조명 방향의 중간인 Halfway벡터를 사용함

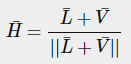
-view방향이 반사 방향과 완벽하게 정렬하게 되면 H는 N과 완벽하게 같아짐

-따라서 H, N의 사이 각 세타가 작을수록 반사되는 빛의 값이 강해짐

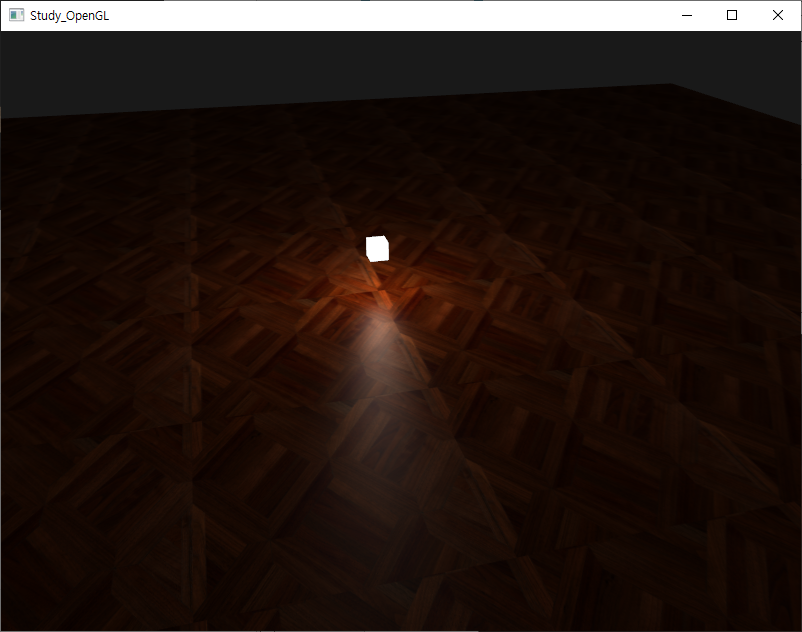
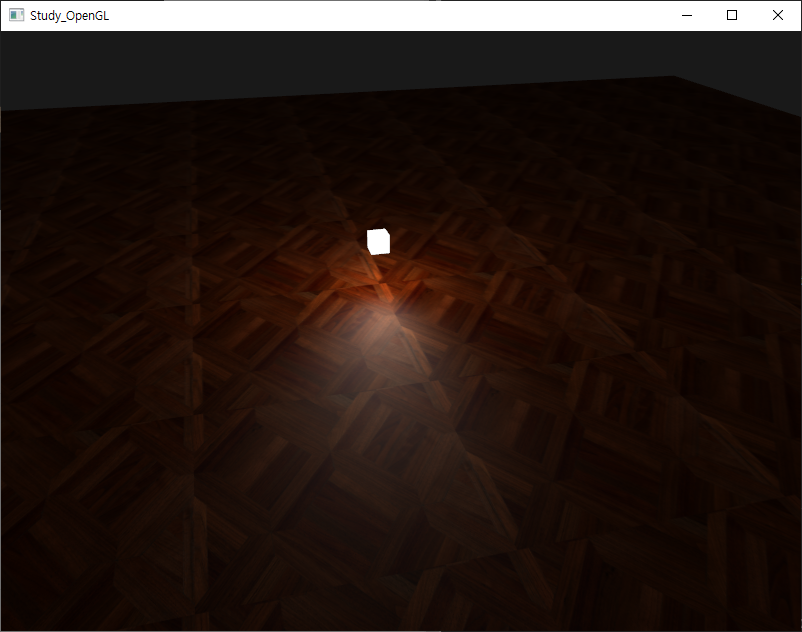
-또한 view가 어느 방향이던 간에 H, N 사이 각은 90도를 넘지 않음

-Blinn-Phong 모델은 OpenGL의 이전 fixed function pipeline에서 사용된 쉐이딩 모델임

-Halfway 벡터를 구하는 법

-GLSL코드

|  |
| --- |
| vec3 lightDir = normalize(lightPos - FragPos);  vec3 viewDir = normalize(viewPos - FragPos);  vec3 halfwayDir = normalize(lightDir + viewDir); |
| float spec = pow(max(dot(normal, halfwayDir), 0.0), shininess);  vec3 specular = lightColor \* spec; |



-왼쪽이 phong, 오른쪽이 blinn-phong