
Materials:

Lighting and Shading
A brief intro

What color is something?

- specify pixel value (2D)

real world

- material
- geometry
- light

standard 3D programming

- compute color from material, geometry, light

Material and Lighting

The material responds to lights

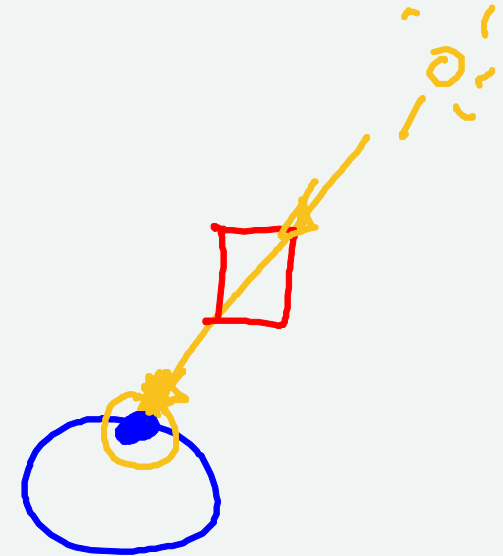
How a point (pixel) appears depends on:

- the surface properties
- the surface orientation
- the color/intensity of the light
- the direction of the light

For now, light travels direct from source to point

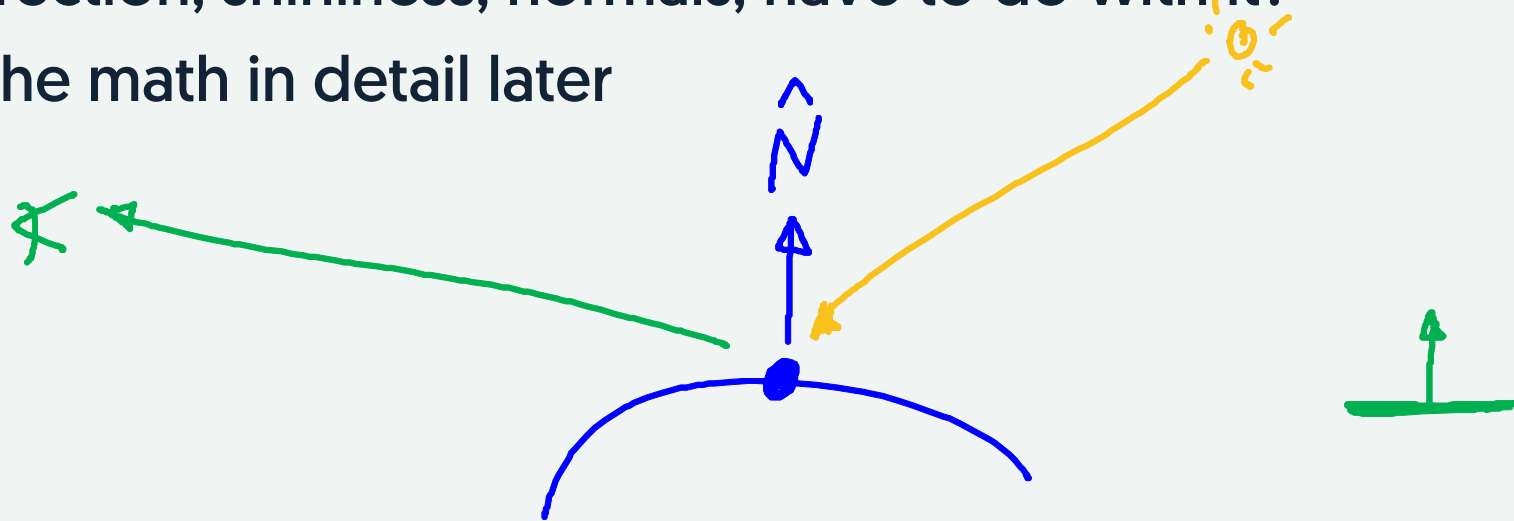
- Local Lighting - no shadows / reflections / spill

→ 1 piece or point of geometry



Shading Intuitions

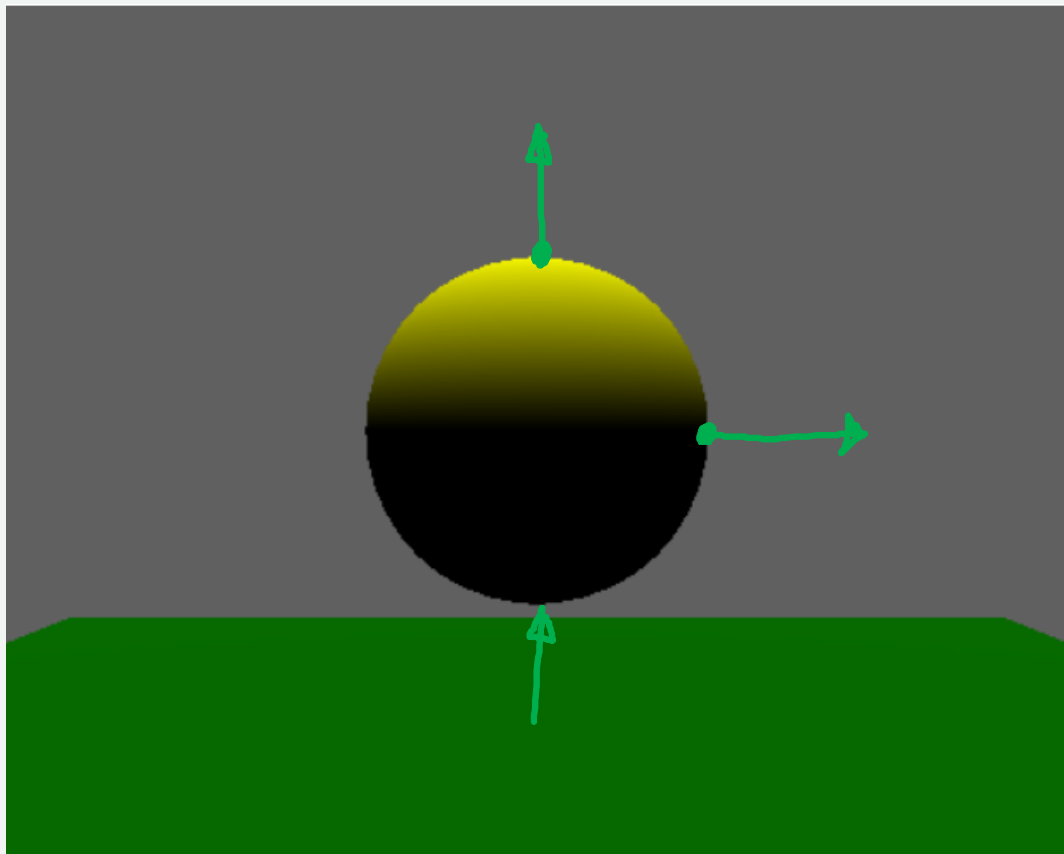
What does direction, shininess, normals, have to do with it?
We'll look at the math in detail later



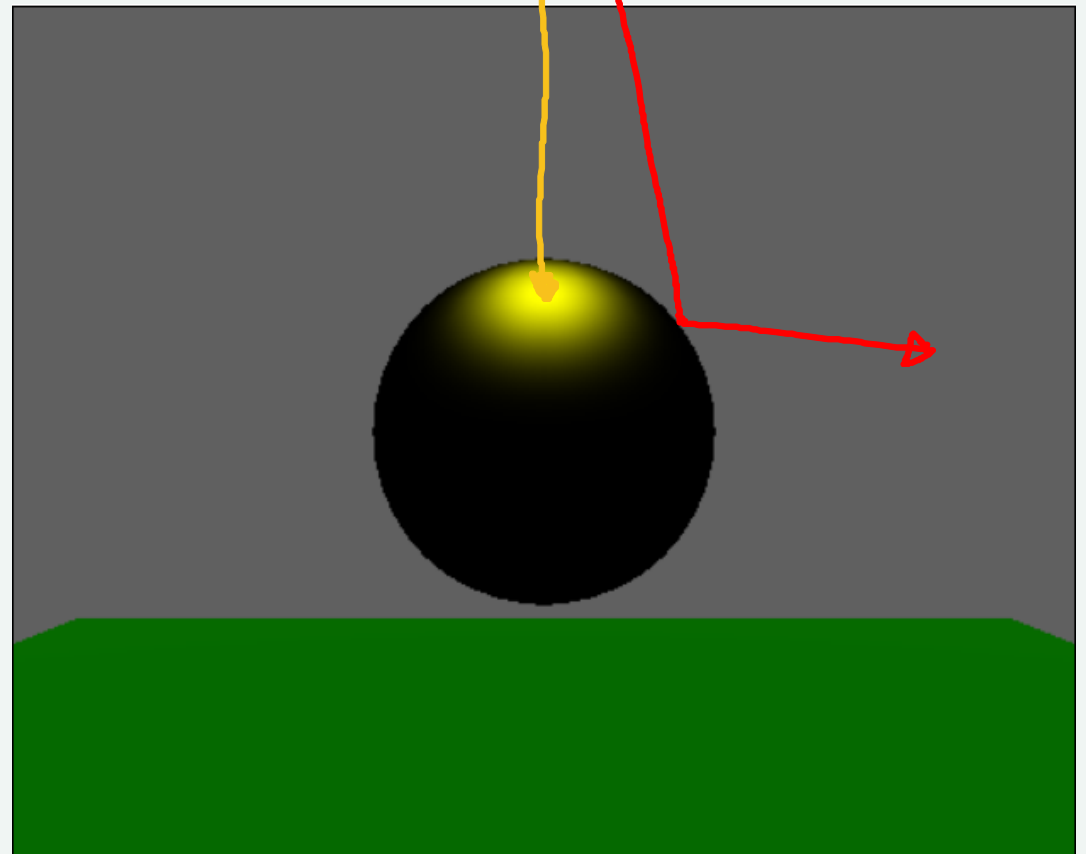
Simple Surface Model

Light in picture is downward

Diffuse



Specular



Colors

- Surfaces have colors
 - per material
 - per vertex (triangle?)
 - more colors later
- Lights have color
- Red light on white object = red
- White light on red object = red
- Red light on blue object? - nothing

Add lights

```
let ambientLight = new T.AmbientLight ("white", 0.5);
scene.add( ambientLight );
let pointLight = new T.PointLight( "white", 1 );
pointLight.position.set( 25, 50, 25 );
scene.add( pointLight );
```

The lights are objects in the world

We control their transformation to place them

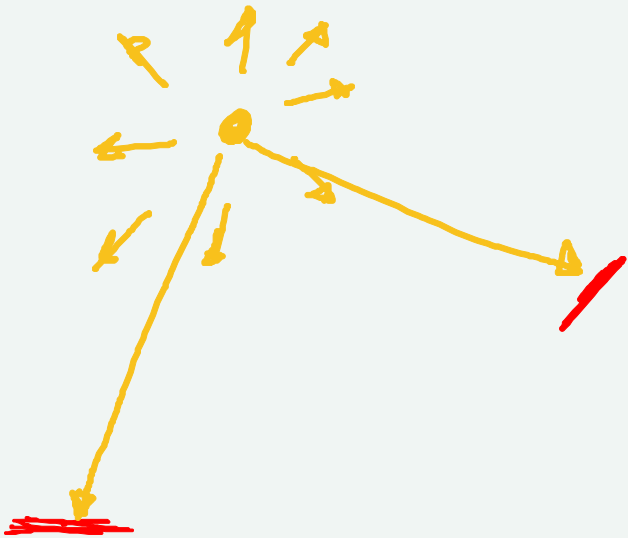
Types of Lights

Ambient Light

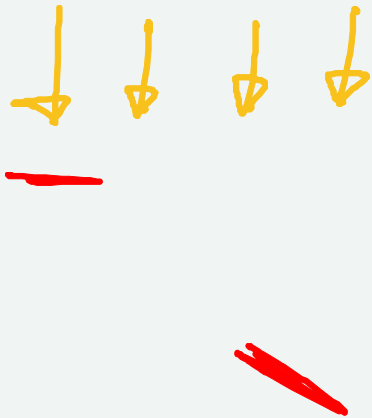


Types of Lights

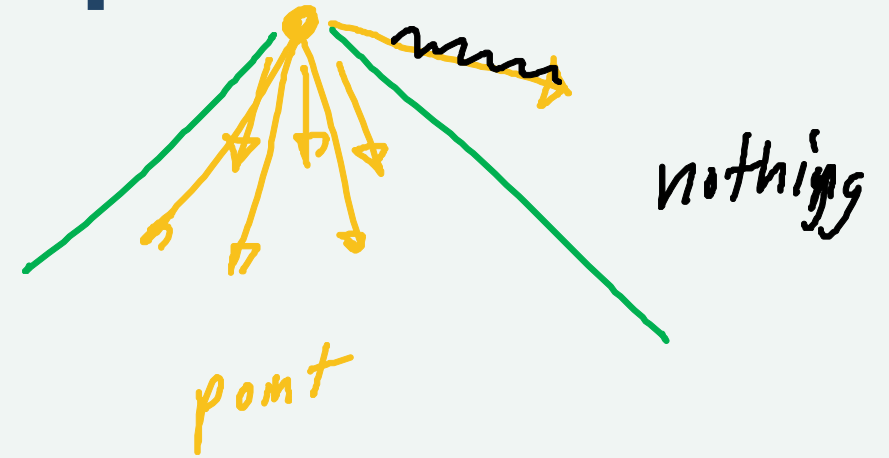
Point



Directional



Spot



Lights in THREE

- They are just Objects!
- You can position and orient them
- THREE's materials know to look for them

Summary

1. Use Materials and Lights to create appearance
2. Color depends on geometry, material, and lighting
3. Specular and Diffuse material properties
4. Local lighting
5. Lights with different geometries