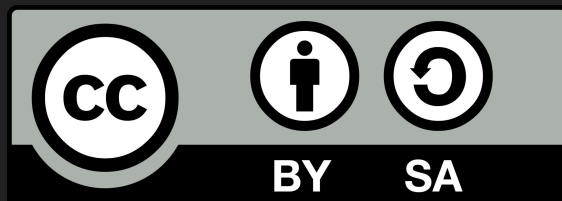


Câmera e Pós-Processamento

Slides por:
Breno M. F. Viana (brenov@usp.br)





Este material é uma criação do
Time de Ensino de Desenvolvimento de Jogos
Eletrônicos (TEDJE)

Filiado ao grupo de cultura e extensão
Fellowship of the Game (FoG), vinculado ao
ICMC - USP

Este material possui licença CC By-SA. Mais informações em:
<https://creativecommons.org/licenses/by-sa/4.0/legalcode>

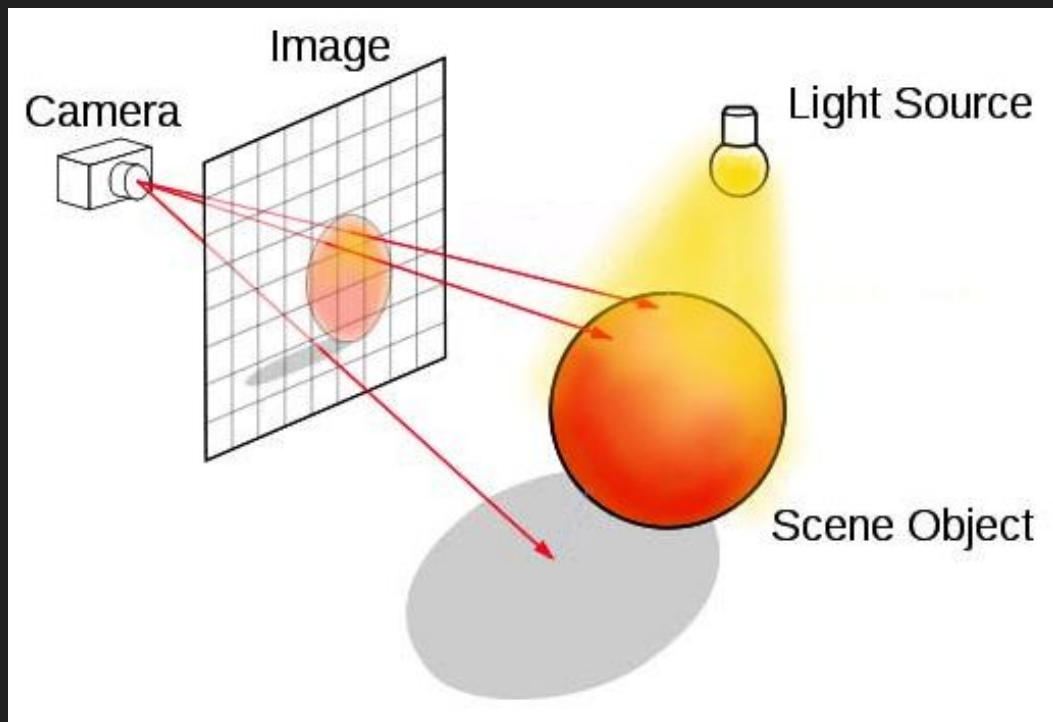


Câmera

- A câmera é responsável por filtrar o que vai ser mostrado em uma cena na tela do computador;
- Sempre é necessário ter uma câmera no jogo.



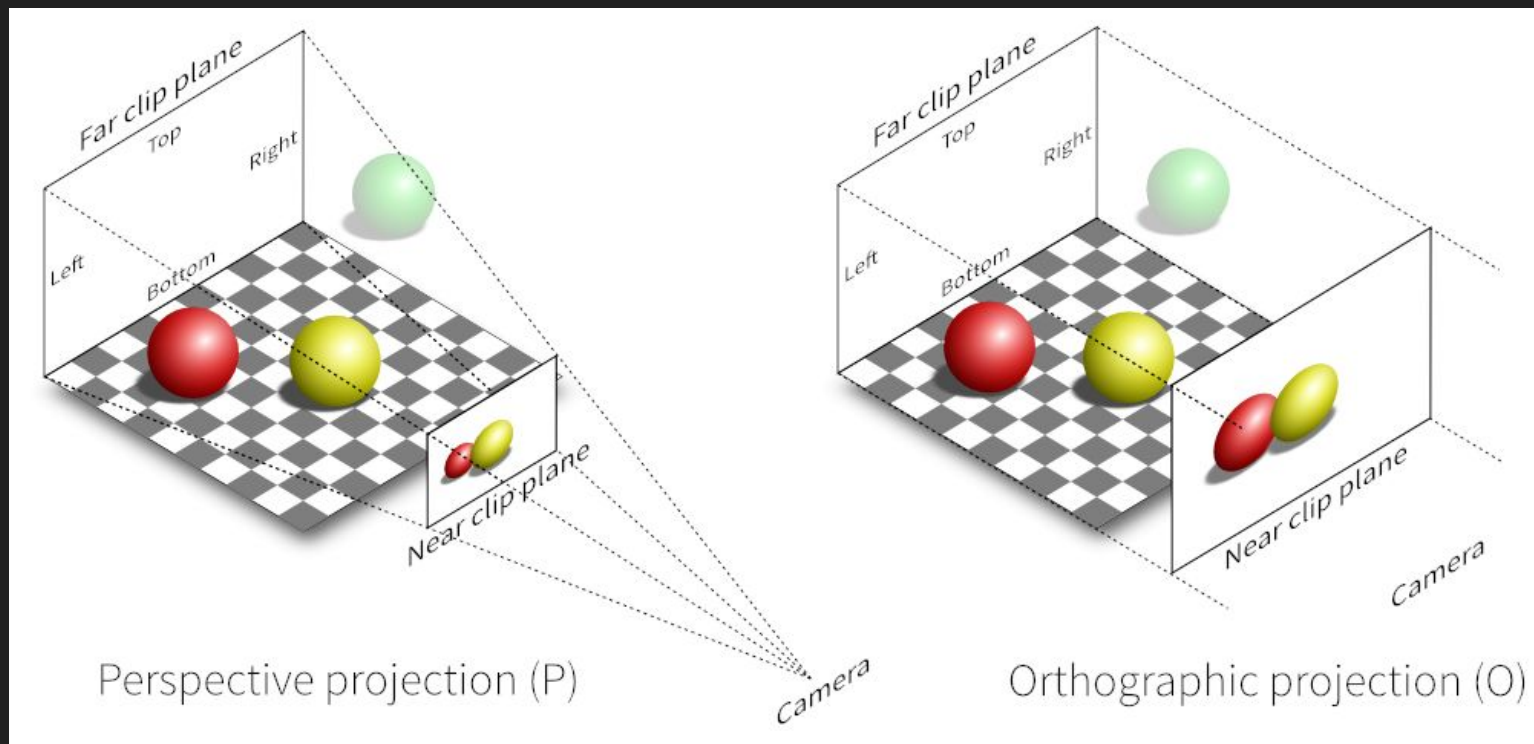
Como uma câmera funciona?



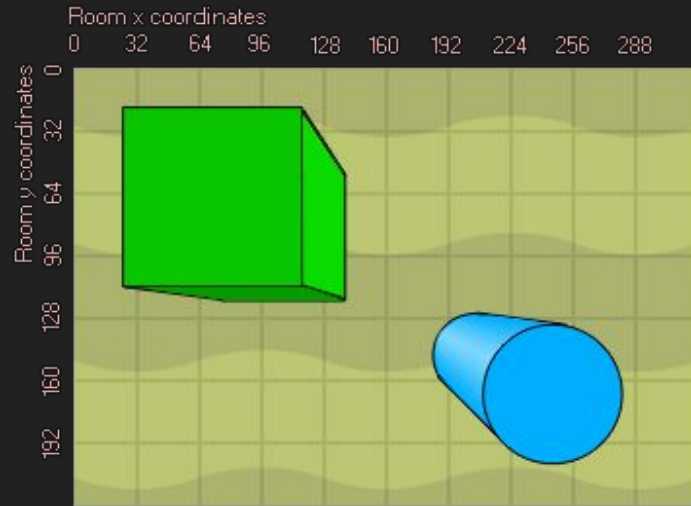
Fonte: <https://sites.google.com/site/tiffanycinglis/fun-stuff/cs-girls-ray-tracing-workshop>



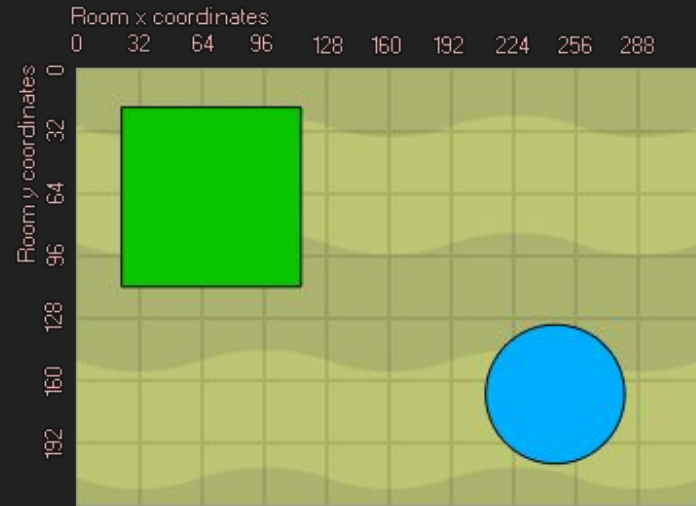
Modo de Visão: Perspectiva vs Ortográfico



Modo de Visão: Perspectiva vs Ortográfico



Perspective View In Room



Orthographic View In Room

Fonte: https://docs.yoyogames.com/source/dadiospice/002_reference/drawing/drawing%203d/3d%20projections/d3d_set_projection_ortho.html

Modo de Visão: Perspectiva vs Ortográfico

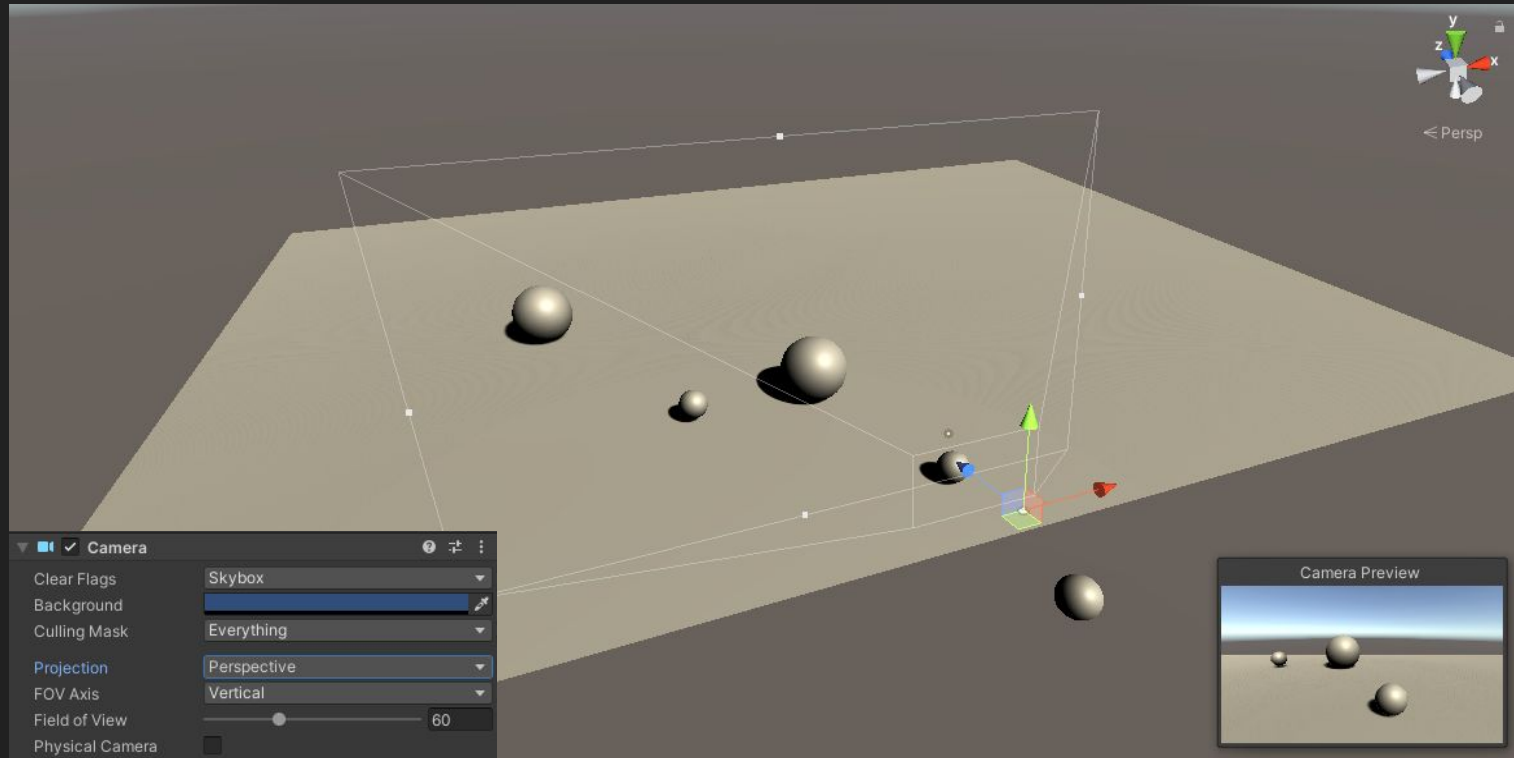


Fonte: https://docs.yoyogames.com/source/dadiospice/002_reference/drawing/drawing%203d/3d%20projections/d3d_set_projection_ortho.html

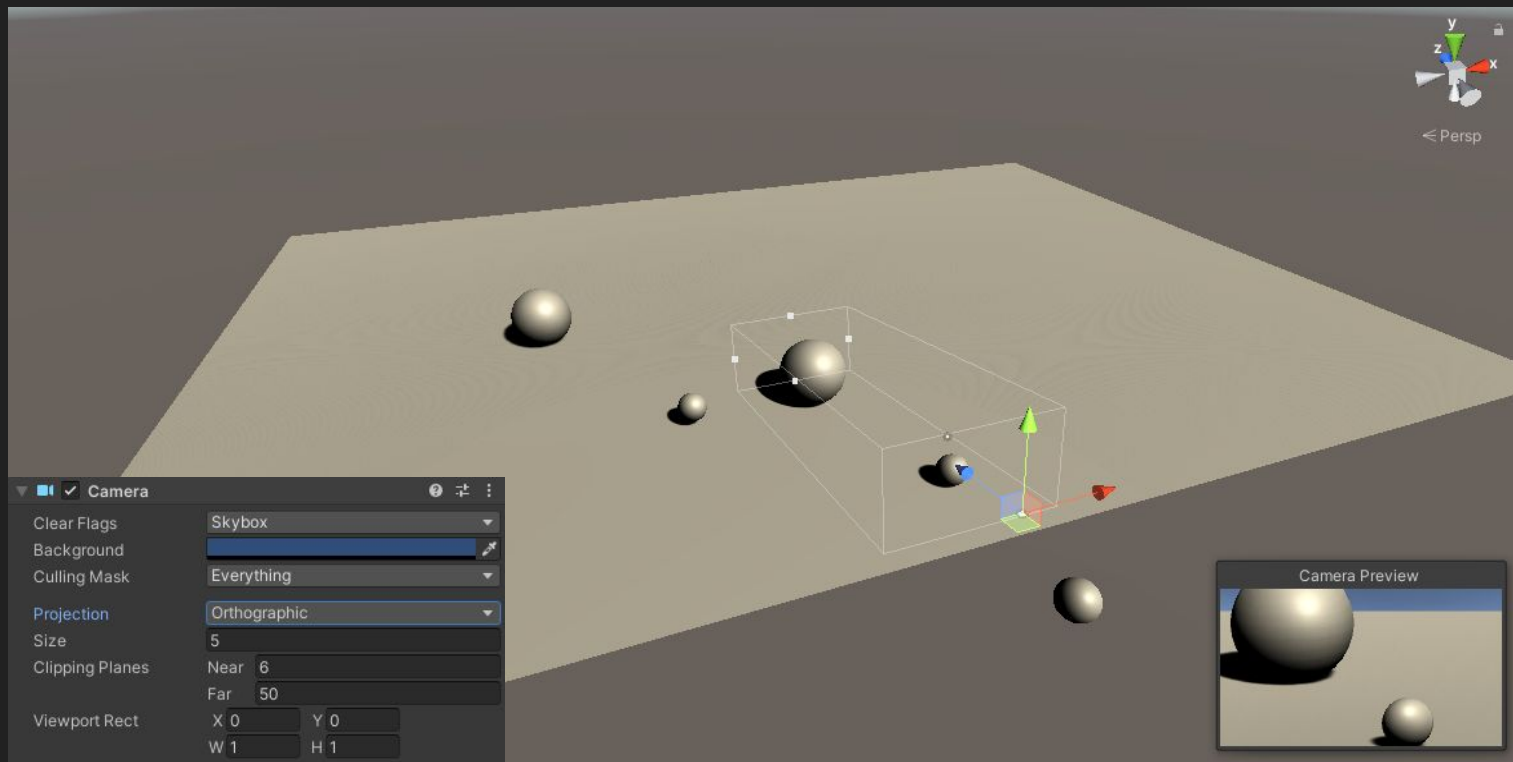
Modos de Visão na Unity



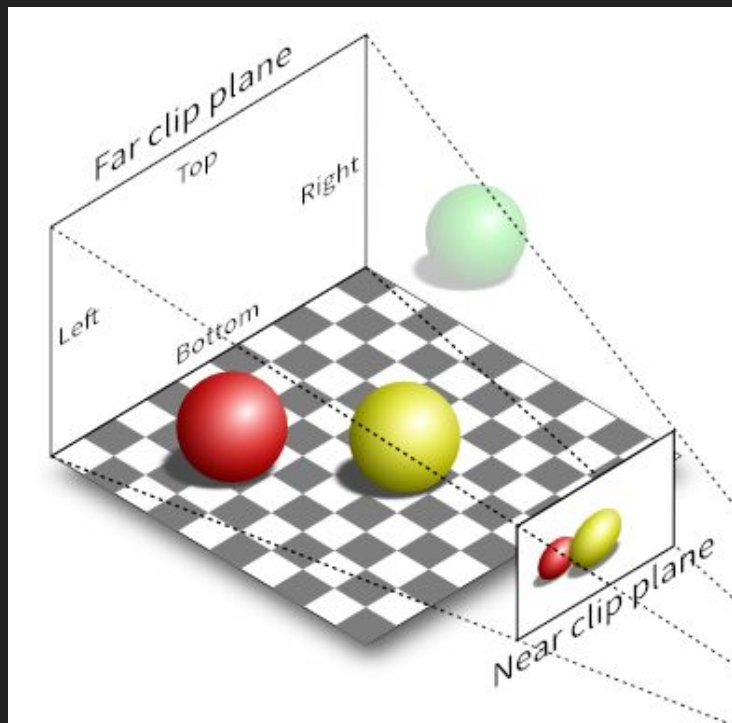
Modo Perspectiva na Unity3D



Modo Ortográfico na Unity3D



Far e Near Clip Planes (Planos de Corte)

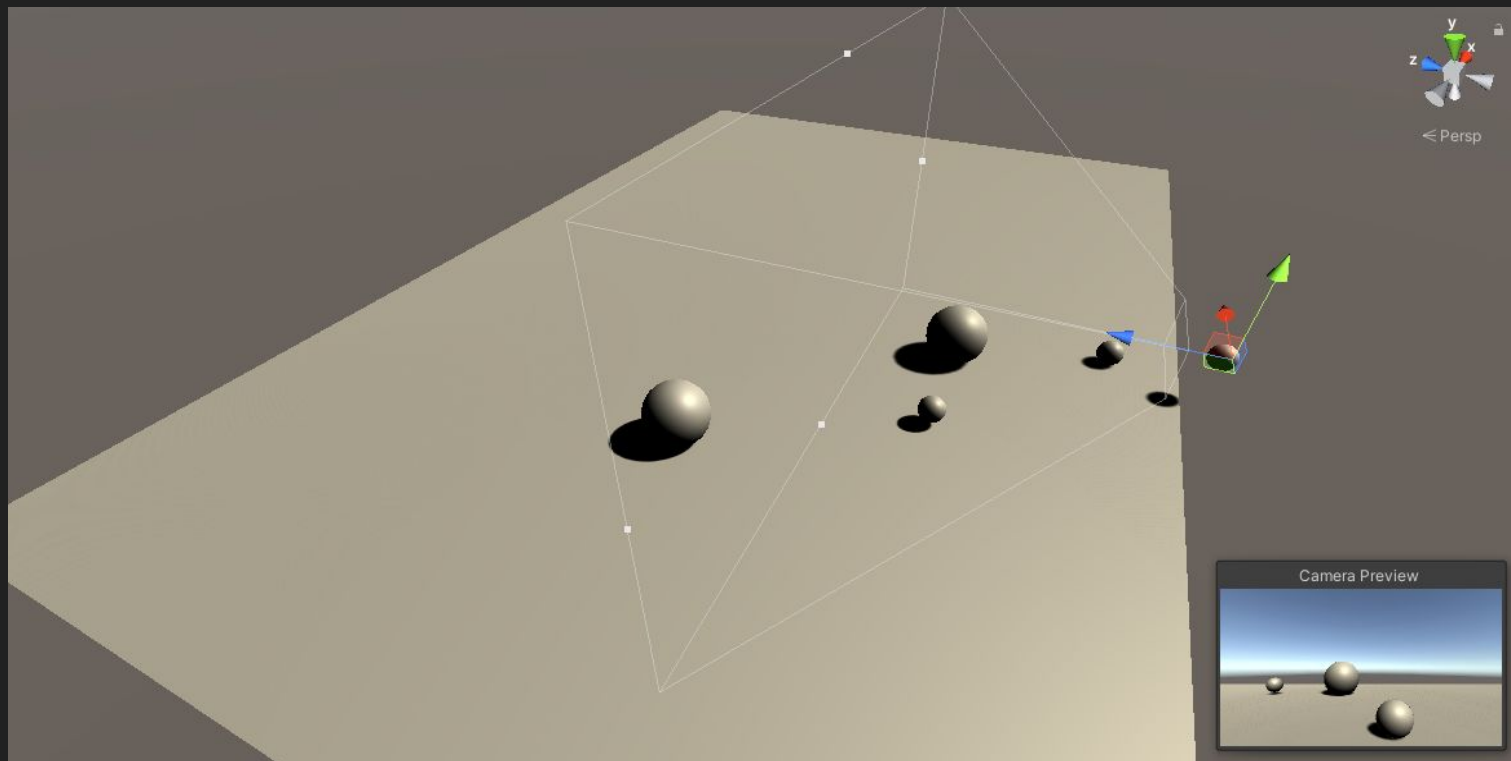


Fonte: Adaptado de
<https://www.geofx.com/graphics/nehe-three-js/lessons17-24/lesson21/lesson21.html>

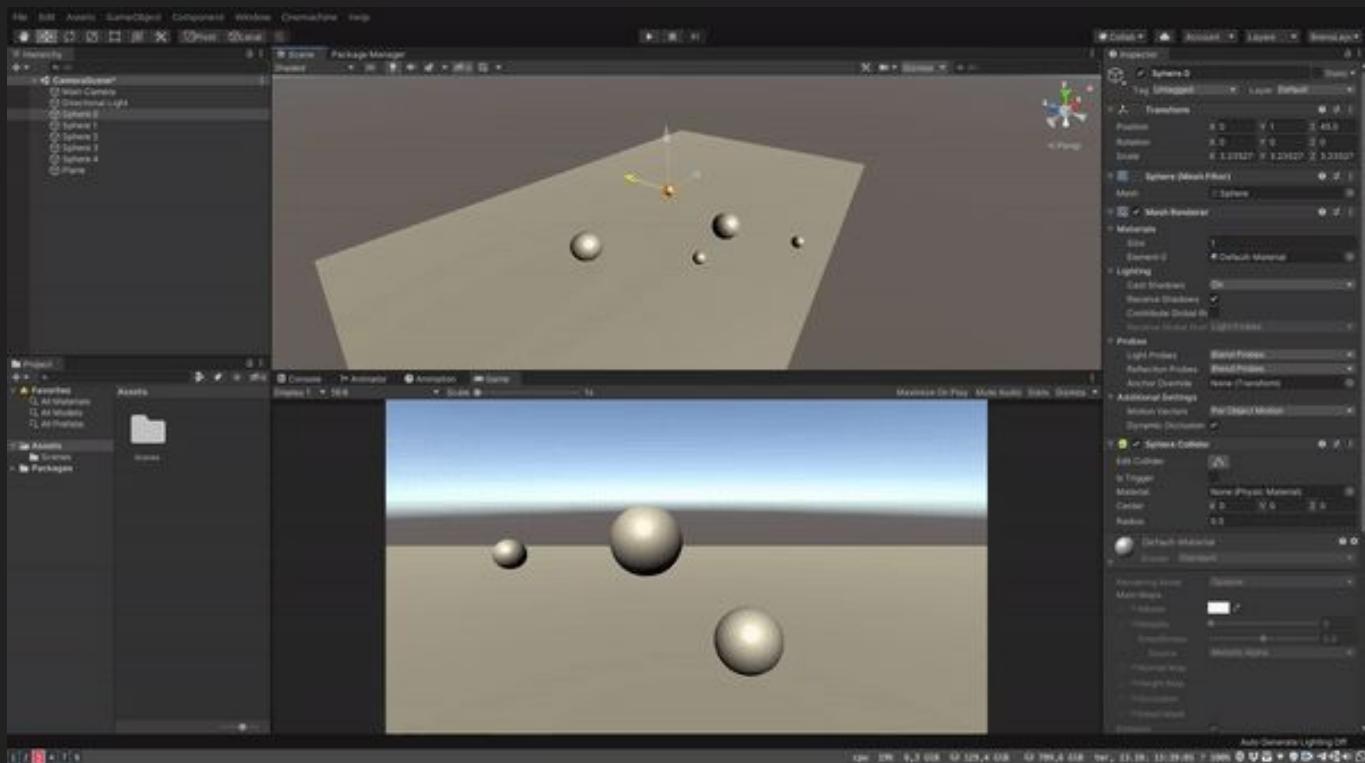
Clip Planes na Unity



Far e Near Clip Planes na Unity3D

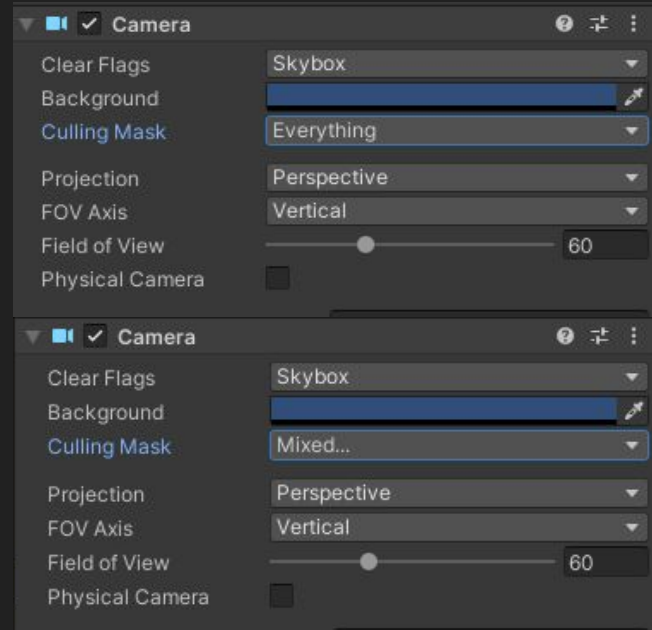


Far e Near Clip Planes na Unity3D



Culling Mask

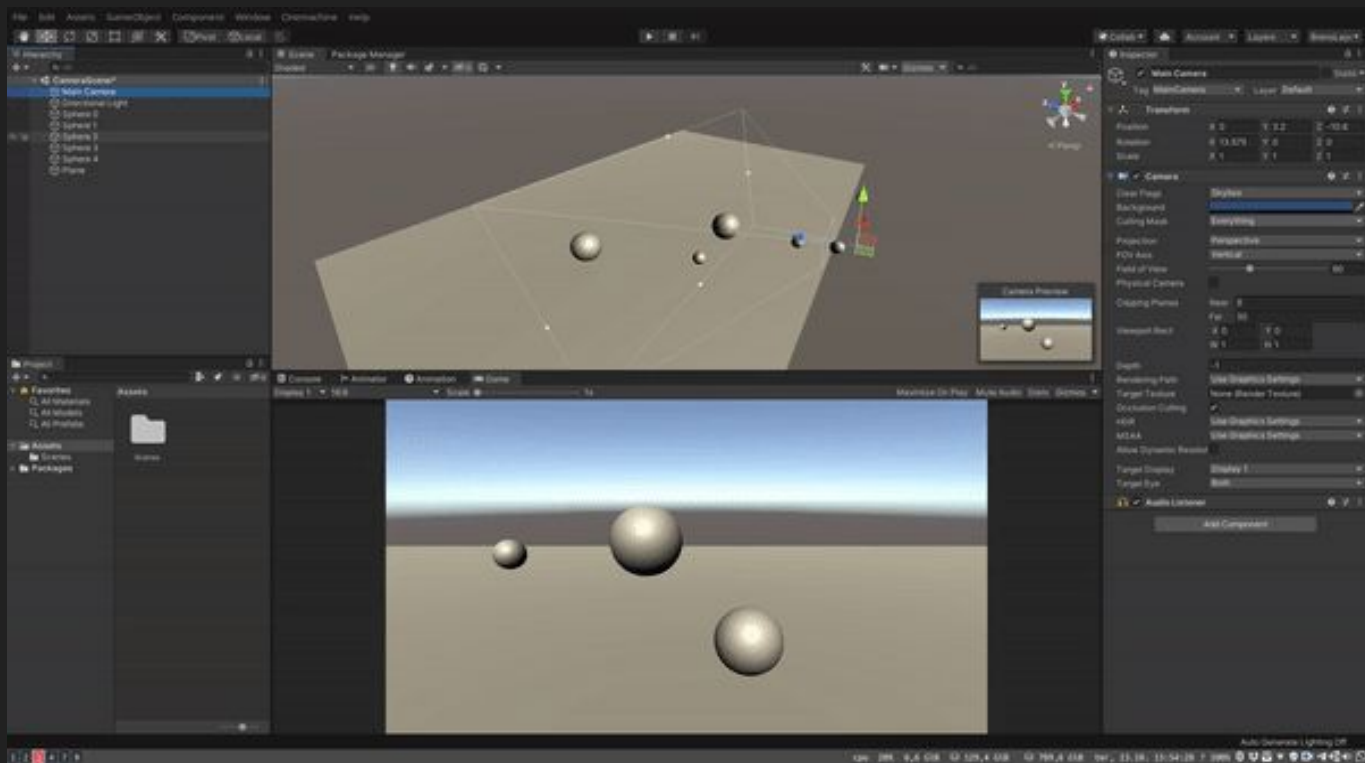
- Por padrão, a câmera enxerga todos os objetos entre os Clip Planes;
- Designar, a partir de camadas (layers), quais objetos serão renderizados pela câmera;
- O objeto não deixa de existir.



Culling Mask na Unity



Culling Mask

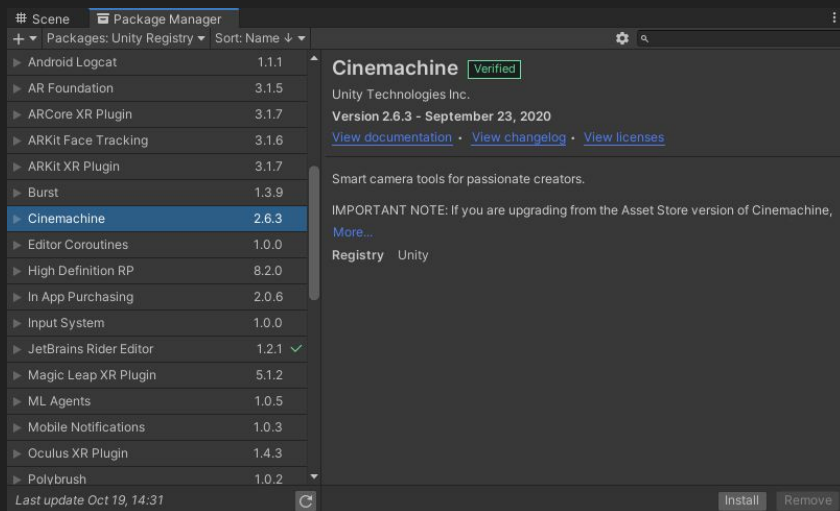


Câmera

- Mas afinal, como fazer a câmera seguir um personagem?
 - Para que a câmera siga o jogador, basta adicioná-la como um objeto filho do jogador.

Cinemachine

- Certos comportamentos de câmera são difíceis de implementar;
- Solução: Cinemachine;
- Instalação via *Package Manager > Unity Registry*.



Cinemachine

- Funciona tanto para jogos 2D quanto para jogos 3D;
- Tipos de Cinemachine:
 - 2D Camera;
 - Virtual Camera;
 - FreeLook Camera;
 - Etc.

Cinemachine

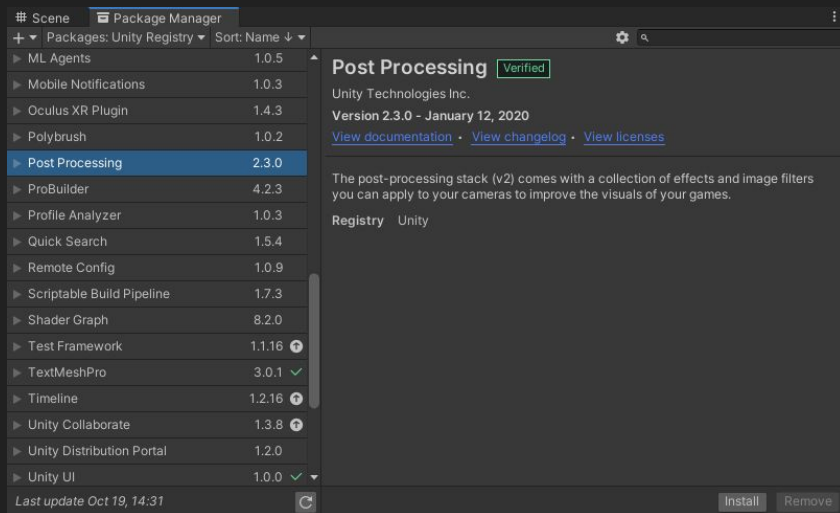
- Funciona tanto para jogos 2D quanto para jogos 3D;
- Tipos de Cinemachine:
 - 2D Camera;
 - Virtual Camera;
 - FreeLook Camera;
 - Etc.
- A cinemachine resolve tudo? Não.

Cinemachine 2D na Unity



Pós-Processamento

- Pós-Processamento consiste em utilizar técnicas de manipulação de imagens em busca de melhorias visuais.



Pós-Processamento

Sem Color Grading

Com Color Grading (Azul)

Sem Vignette



Com Vignette



Pós-Processamento

- Post-Process Layer:
 - Define onde os efeitos vão se aplicados.
- Post-Process Volume:
 - Define quais efeitos serão aplicados.

Pós-Processamento

- Efeitos
 - Color Grading: filtro de cor;
 - Bloom: filtro de fluorescência;
 - Vignette: filtro de enquadramento;
 - Chromatic Aberration: filtro de falha de câmera;
 - Auto exposure: filtro de correção de exposição de luz;
 - Depth of field: filtro de foco de câmera;
 - Motion blur: filtro de borrão de movimento;
 - Ambient Occlusion: filtro de sombreamento.

Pós-Processamento

- Post-Process Debug:
 - Histograma;
 - Debug Overlay > Colorblind mode.

Referências

- Camera: <https://docs.unity3d.com/ScriptReference/Camera.html>
- Cinemachine: <https://unity.com/pt/unity/features/editor/art-and-design/cinemachine>
- Cinemachine 2D games: <https://www.youtube.com/watch?v=2jTY11Am0lg>
- Cinemachine Third Person Game: <https://www.youtube.com/watch?v=537B1kJp9YQ>
- Post-Processing: <https://docs.unity3d.com/Manual/PostProcessingOverview.html>
- Post-Processing na Unity: <https://www.youtube.com/watch?v=pPar4bWpHfK&t=0s>