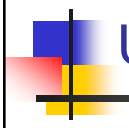


Unity Game Engine

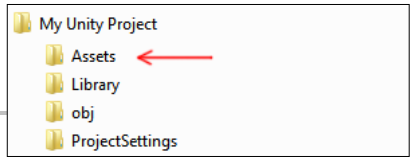
Introduction to Unity – Basic Game Graphics

https://docs.unity3d.com/Manual/Graphics.html

Graphics

- To understand how to work on and improve the visuals.

Assets



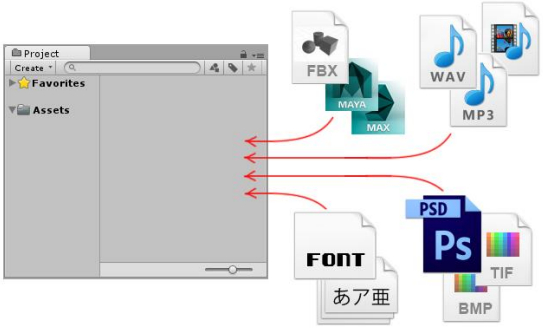
- Your project folder contains the “Assets” subfolder.
- The contents of the **Project Window** in Unity shows the items in your Assets folder.
  - The items you see in your Project window represent (in most cases) actual files on your computer, and if you delete them within Unity, you are deleting them from your computer too.

Assets

- An asset is representation of any item that can be used in your game or project.
  - An asset may come from a file created outside of Unity, such as a 3D model, an audio file, an image, or any of the other types of file that Unity supports.
  - Some asset types can be created within Unity, such as a Prefab, an Animator Controller, an Audio Mixer or a Render Texture.

Adding Assets

- Drag a file into Unity's Project Window from your computer (eg, from Explorer on Windows). It will be *copied* into your Assets folder.
- Or Assets> Import New Asset.



To safely move or rename your assets is to always do it from within Unity's project folder. This way, Unity will automatically move or rename the corresponding meta file.

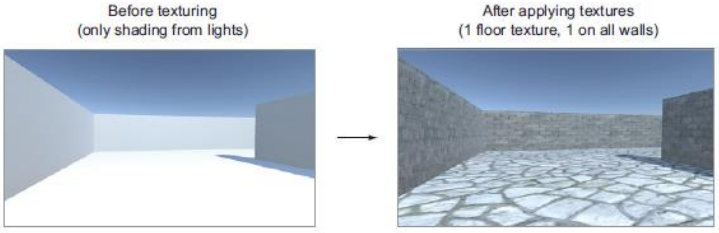
Art Assets & Types

- An art asset is an individual unit of visual information (usually a file) used by the game.

Type of art asset	Definition of this type
2D image	Flat pictures. To make a real-world analogy, 2D images are like paintings and photographs.
3D model	3D virtual objects (almost a synonym for "mesh objects"). To make a real-world analogy, 3D models are like sculptures.
Material	A packet of information that defines the surface properties of any object that the material is attached to. These surface properties can include color, shininess, and even subtle roughness.
Animation	A packet of information that defines movement of the associated object. These are detailed movement sequences created ahead of time, as opposed to code that calculates positions on the fly.
Particle system	An orderly mechanism for creating and controlling large numbers of small moving objects. Many visual effects are done this way, such as fire, smoke, or spraying water.

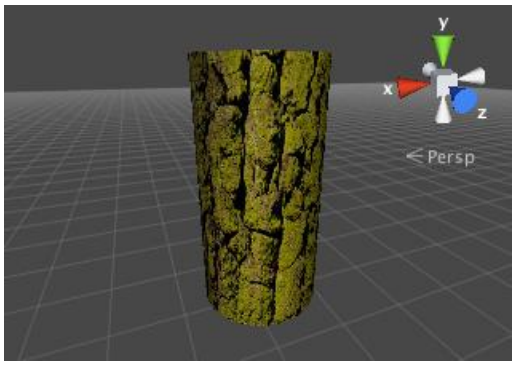
Texture the scene

- A *texture* is a 2D image being used to enhance 3D graphics.
- It improves the look of 3D model and gives details without altering actual geometry of the model.



Textures

- The mesh geometry of an object (e.g a cylinder blow) only gives a rough approximation of the shape while most of the fine detail is supplied by **Textures**.



Texture Image File Formats

- Unity supports the use of many different file formats.
- These days PNG is more widespread, because it's useful both on the web and as a texture.

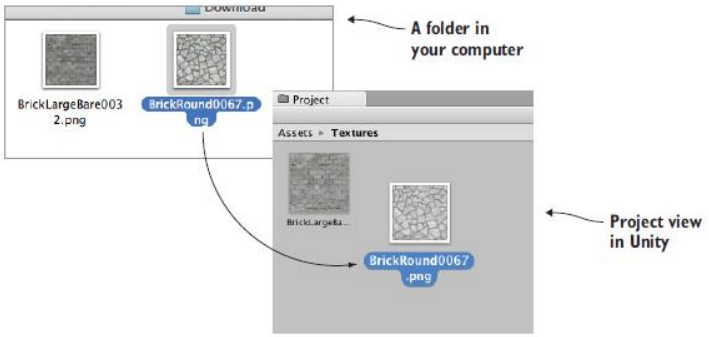
File type	Pros and cons
PNG	Commonly used on the web. Lossless compression; has an alpha channel.
JPG	Commonly used on the web. Lossy compression; no alpha channel.
GIF	Commonly used on the web. Lossy compression; no alpha channel. (Technically the loss isn't from compression; rather, data is lost when the image is converted to 8-bit. Ultimately it amounts to the same thing.)
BMP	Default image format on Windows. No compression; no alpha channel.
TGA	Commonly used for 3D graphics; obscure everywhere else. No or lossless compression; has an alpha channel.
TIFF	Commonly used for digital photography and publishing. No or lossless compression; no alpha channel.
PICT	Default image format on old Macs. Lossy compression; no alpha channel.
PSD	Native file format for Photoshop. No compression; has an alpha channel. The main reason to use this file format would be the advantage of using Photoshop files directly.

Get Texture Images

- Get tileable images from one of the many websites (Eg. www.textures.com)
- Textures should be sized in powers of 2 ((e.g. 32x32, 64x64, 128x128, 256x256, etc.).
- In your image editor (Photoshop, GIMP, or whatever;) scale the downloaded image to 256x256, and save it as a PNG.

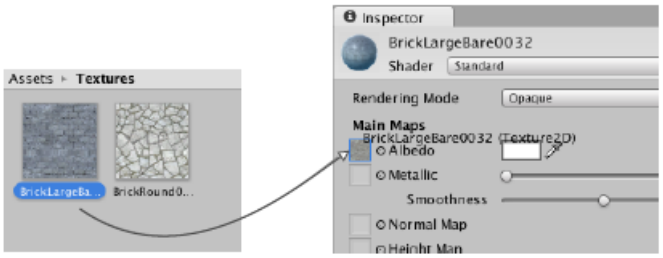
Import Image File to Project

- Drag the texture files from a folder in your computer into the Project view in Unity or right-click in Project and select “Import New Asset”.



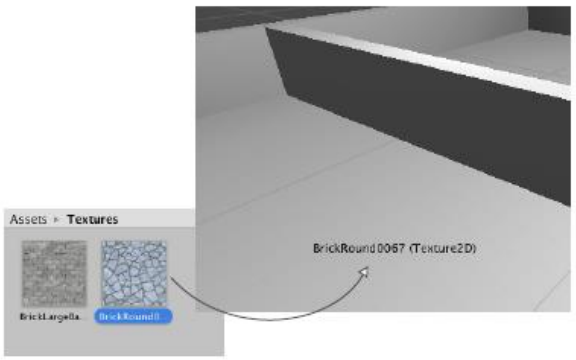
Apply the Texture Image to an Object

- Step 1: Create a Material (Assets > Create > Material.) In the Inspector, drag a texture to the main texture slot (named Albedo).
- Step 2: Drag the material up from Project onto an object in the scene to apply the material to that object.



Apply the Texture Image to an Object (shortcut)

- Drag a texture from Project view onto an object in the scene. *Unity will create a new material and apply the new material to the object.*

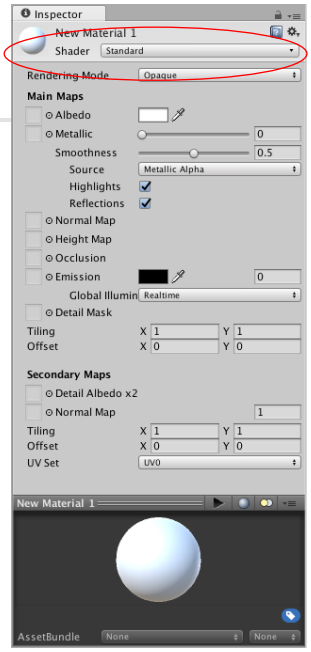


Materials and Textures

- There is a close relationship between Materials and Textures.
- **Materials** are definitions of how a surface should be rendered, including references to textures used, tiling information, colour tints and more.
- **Textures** are bitmap images. A Material may contain references to textures. In addition to basic colour (albedo) of an object's surface, textures can represent many other aspects of a material's surface such as its reflectivity or roughness.

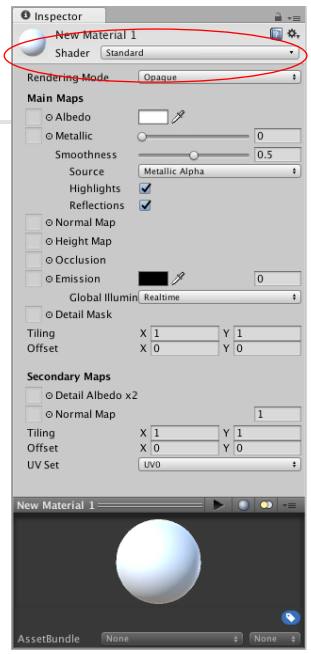
Creating Materials

- To create a new Material,
  - use **Assets->Create->Material** from the main menu or the Project View context menu.



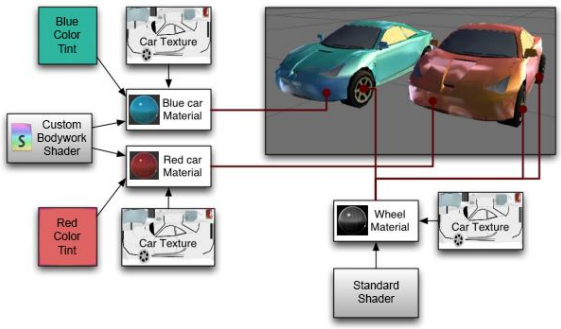
Materials

- A Material defines:
  - Which shader to use for rendering this material.
  - The specific values for the shader's parameters - such as which texture maps, the colour and numeric values to use.
- By default, new materials are assigned the "Standard" Shader, with all map properties empty.
  - You can select a different Shader. Simply expand the **Shader** drop-down in the Inspector, and choose your new Shader.



Shader

- A **shader** is a method (program) that outlines instructions for how to draw a surface, including whether to use any textures. Every material has a shader that controls it. Standard shader is the default shader.



Standard Shader

- The Unity Standard Shader is a built-in shader with a comprehensive set of features. It can be used to render “real-world” objects such as stone, wood, glass, plastic and metal, and supports a wide range of shader types and combinations.

A scene rendered using the standard shader on all models

Other Built-in Shaders

In addition to the **Standard Shader,** there are

- **FX**: Lighting and glass effects.
- **GUI** and **UI**: For user interface graphics.
- **Mobile**: Simplified high-performance shader for mobile devices.
- **Nature**: For trees and terrain.
- **Particles**: Particle system effects.
- **Skybox**: For rendering background environments behind all geometry
- **Sprites**: For use with the 2D sprite system
- **Unlit**: For rendering that entirely bypasses all light & shadowing
- **Legacy**: The large collection of older shaders which were superseded by the Standard Shader

Other Built-in Shaders

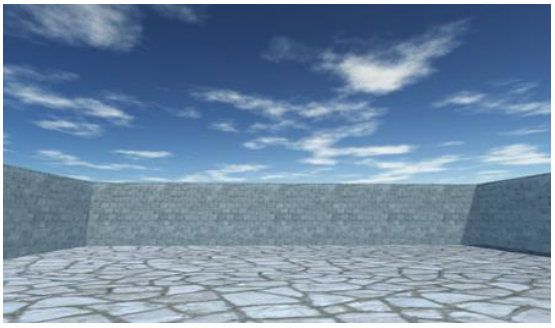
In addition to the **Standard Shader,** there are

- **FX**: Lighting and glass effects.
- **GUI** and **UI**: For user interface graphics.
- **Mobile**: Simplified high-performance shader for mobile devices.
- **Nature**: For trees and terrain.
- **Particles**: Particle system effects.
- **Skybox**: For rendering background environments behind all geometry
- **Sprites**: For use with the 2D sprite system
- **Unlit**: For rendering that entirely bypasses all light & shadowing
- **Legacy**: The large collection of older shaders which were superseded by the Standard Shader

3D Computer Game Programming

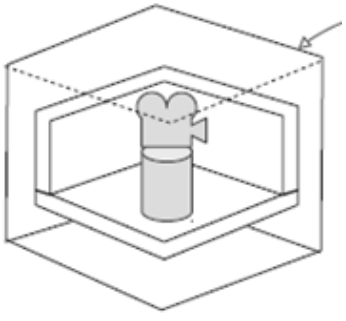
Sky Visuals by Skybox

- A realistic look for the sky - the most common approach is a special kind of texturing using pictures of the sky.



Skybox

- A *skybox* is a method of creating backgrounds by projecting textures onto the cube's faces, thus creating the illusion of distant three-dimensional surroundings.

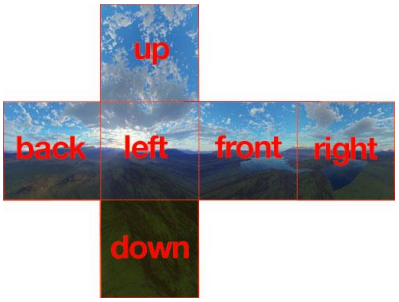


Sky Visuals by Skybox

- Step 1: Obtain Skybox texture images
- Step 2: Create **a material with Skybox shader**
- Step 3: Set Environment Lighting to Skybox.

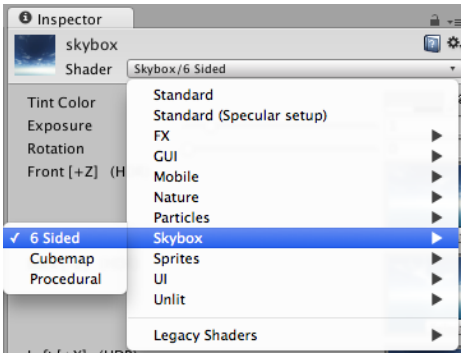
Step 1 - Skybox Texture

- Obtain skybox textures (6 images: up, down, left, right, front, back) and import the images to the project.
  - Change the “Wrap Mode” from “Repeat” to “Clamp” ***for each texture.***



Step 2 - Material with Skybox Shader

- Create a new material (Assets > Create > Material) and choose ***Skybox shader > 6 sided*** in Inspector.

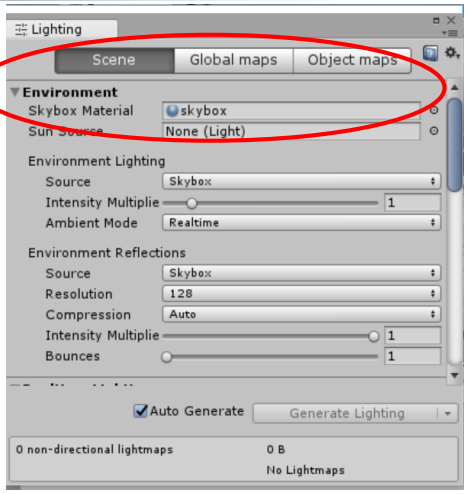


Step 2 - Material with Skybox Shader (cont'd)

- Assign 6 texture images to the corresponding slots.

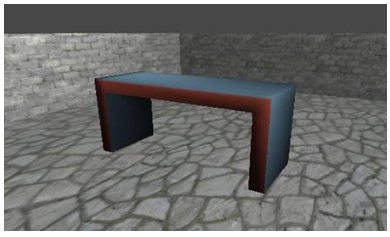
Step 3 - Environment Lighting

- Navigation menu: Window > Settings
- Choose Scene tab.
- Assign your skybox material to Skybox Material of Environment.



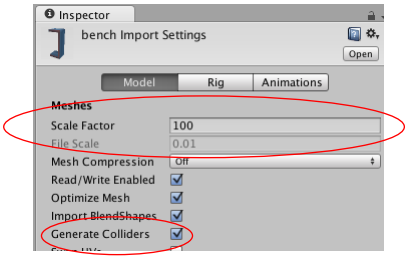
Custom 3D Models

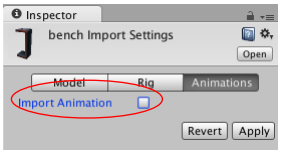
- We can build 3D models in external 3D art apps such as Autodesk's Maya, Blender, and 3ds Max.
- 3D file formats that Unity supports:
  - Mesh and animation: FBX, Collada (DXE)
  - Mesh only: FBX, OBJ, 3DS, DXF
  - FBX for Maya, Blender and 3ds Max



Import a Model in FBX

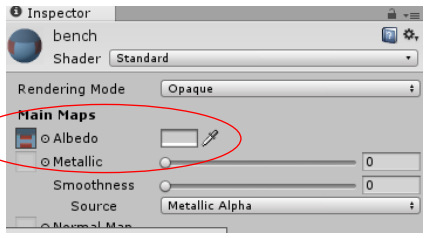
- Drag the FBX file from the computer into Unity's Project view or Import New Asset.
  - Change Scale
  - Click the "Generate Colliders" check box under the "Model" tab.
  - Deselect "Import Animation" under the Animation tab if it is a static model.





Assign a Texture Material to a Model

- When Unity imports the FBX file, it also creates a material for the bench.
- Assign a texture:
  - drag the texture image into Project to import it into Unity
  - then drag the imported texture onto the texture slot of the bench material



Resources

- Textures
  - Bricks
  - Bench
  - Skybox
- 3D model (bench.fbx)

Unity's Asset Store

- Many 3D models are available for download from Unity's Asset Store https://www.assetstore.unity3d.com.