2 – Using MakeHuman Characters

A series on Unity 3D introductory tutorials.

In Unity's "Standard Assets" resource, once installed, we have one character, ***ethan***, already configured to be used in scenes: it's managed as a **3rd Person Controller** prefab with the **standard gesture animations** (Walk, run, jump, crouch, idle). We can get other characters/avatars in public 3D repositories like Unity's Asset Store (lots of free stuff in there) or by making them ourselves in programs like Makehuman.

Although in this tutorial we are referring to MakeHuman, an easy and intuitive software that can export characters in .fbx, properly configured with a "Game Engine Rig" compatible with Unity3D, the procedures referred (after point 1) are valid to import similarly configured characters made by other reference software (Blender, 3DS Max, Daz Studio, Mixamo, …)

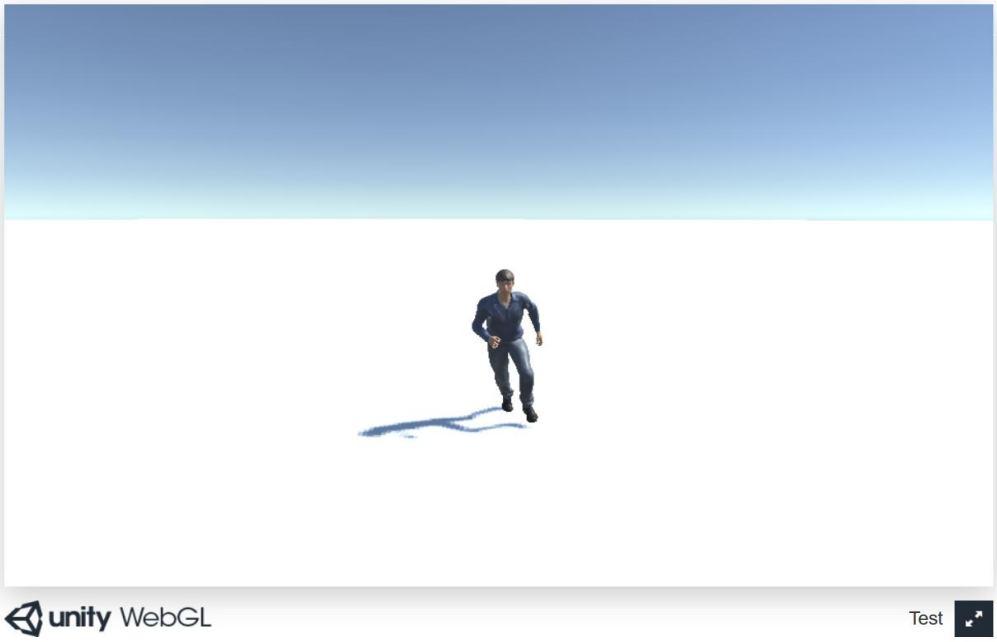


Figure 1 – An imported MakeHuman character performing in a WebGL basic scene created by Unity3D

Install Makehuman: download from http://www.makehuman.org/

Some terminology:

- **Character** designates a dynamic entity that performs in the scene, can be a humanoid, non-humanoid or other.
- **Avatar or Player** is a character that represents the user in the scene.
- **Third Person View** - when the avatar is visible in the scene.

1. Create a Character in Makehuman

The creation process is intuitive and we quickly get a character, named here **avatarMH1**, as we see in Figure 2. If you need some hints on designing the humanoid look at tutorial: MakeHuman 1.1 -- A Completely Free 3D Character Creator

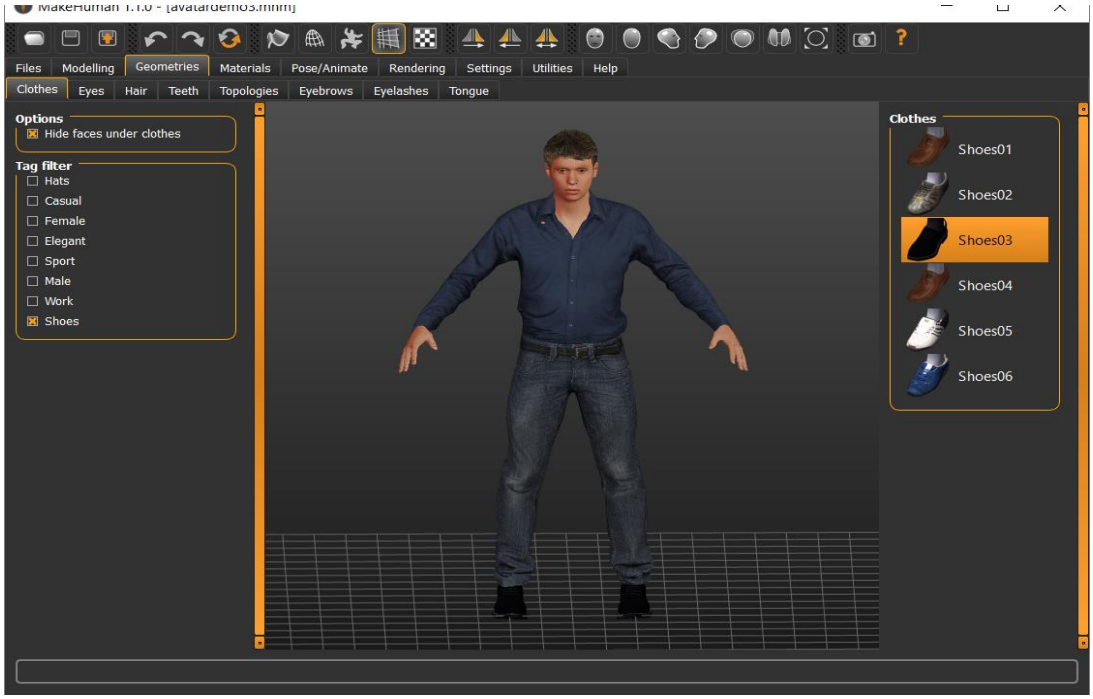


Figure 2 – Caracter creation in Makehuman

When the character shape, the modeling, is finished:

1. **Add a Game Engine Rig** - add to the character *avatarMH1* a "Game Engine" Rig on Pose/Animate>>Skeleton (Figure 3). This is essential in order to be recognized correctly in Unity3D and use unity's standard avatar gesture animations.

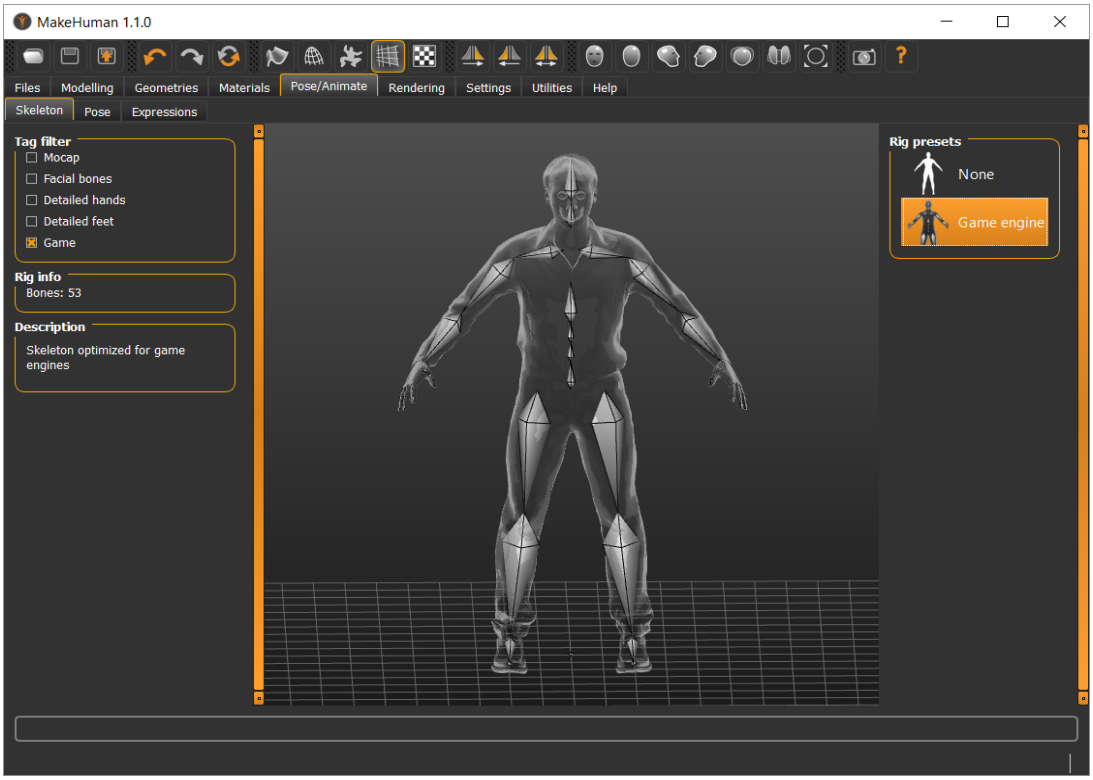
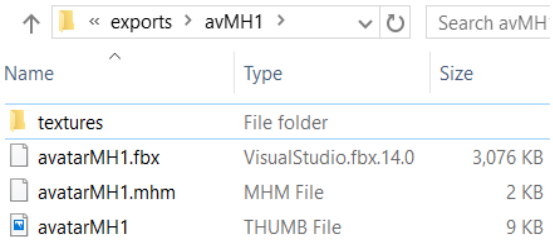


Figure 3 – Configure the character with a "Game Engine" Rig

2. **Save the character in a folder** created specifically for this *avatarMH1* character on your PC. Our folder is named "avMH1" and everything related to this character will be saved in there. This **good practice** will assure that we always have the correct path to all character's resources, textures and components. This is important to avoid disconnected resources like textures not appearing, parts missing, etc.



3. **Export the character in .fbx mesh format** (Figure 4), a format recognized by Unity3D, to "avMH1" folder. Now you have in the same folder the *avatarMH1* character in both mhm (makehuman) format and in .fbx, so it's ready to be imported into Unity3D.

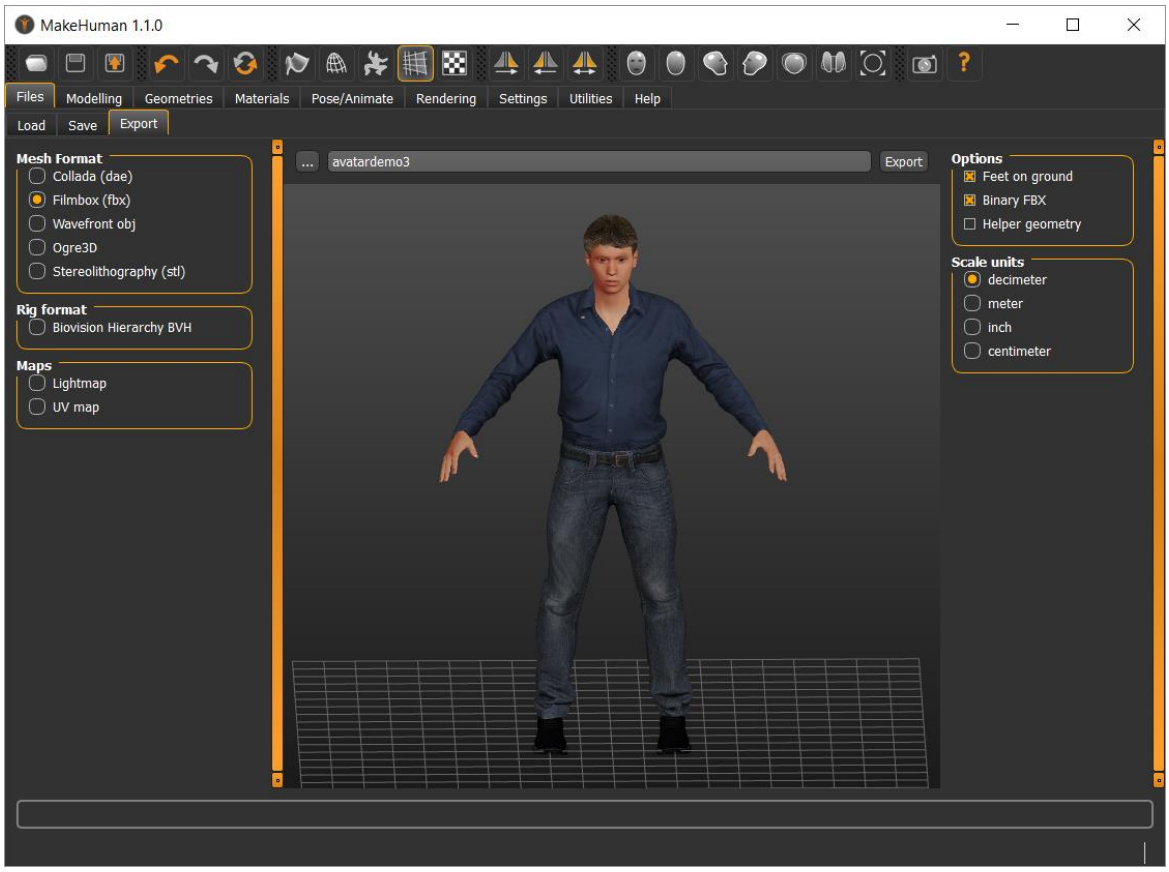


Figure 4 – Export the caracter as .fbx

Now that we have what we need from MakeHuman, it's time to start working with Unity3D.

2. Starting with a basic pre-built scene with the "default" character ethan

We start here with a pre-built basic scene (Figure 5) with ethan character configured and working as was explained in point 6 of our Unity's tutorial "1 - Easy first Scene".

In the remain of this tutorial we will import the new avatar to Unity, configure the rig to Humanoid and replace ethan in our pre-built scene.

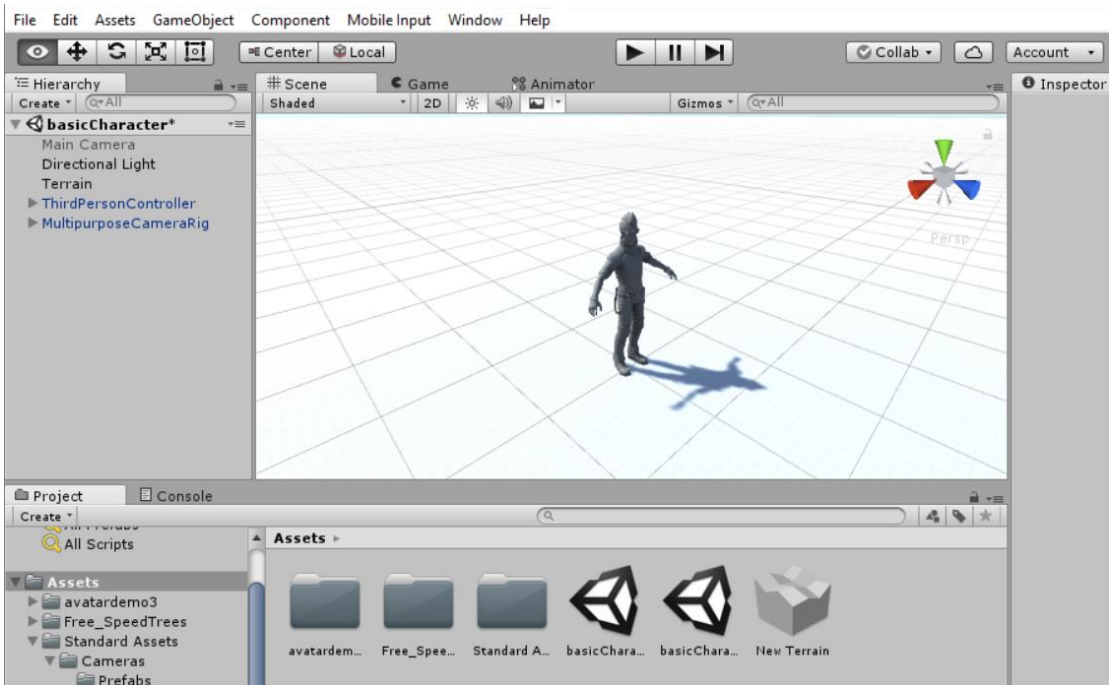


Figure 5 - A basic Unity3D scene with the "default" character ethan

2.1. Importing Makehuman files

Drag the entire folder "avMH1" created in MakeHuman to Assets Project folder in Unity's Interface (Figure 6). Unity will automatically make the necessary file conversions. Note that the "entire folder" should be dragged, otherwise the *path* can be lost and images or components may be missing on the character's body.

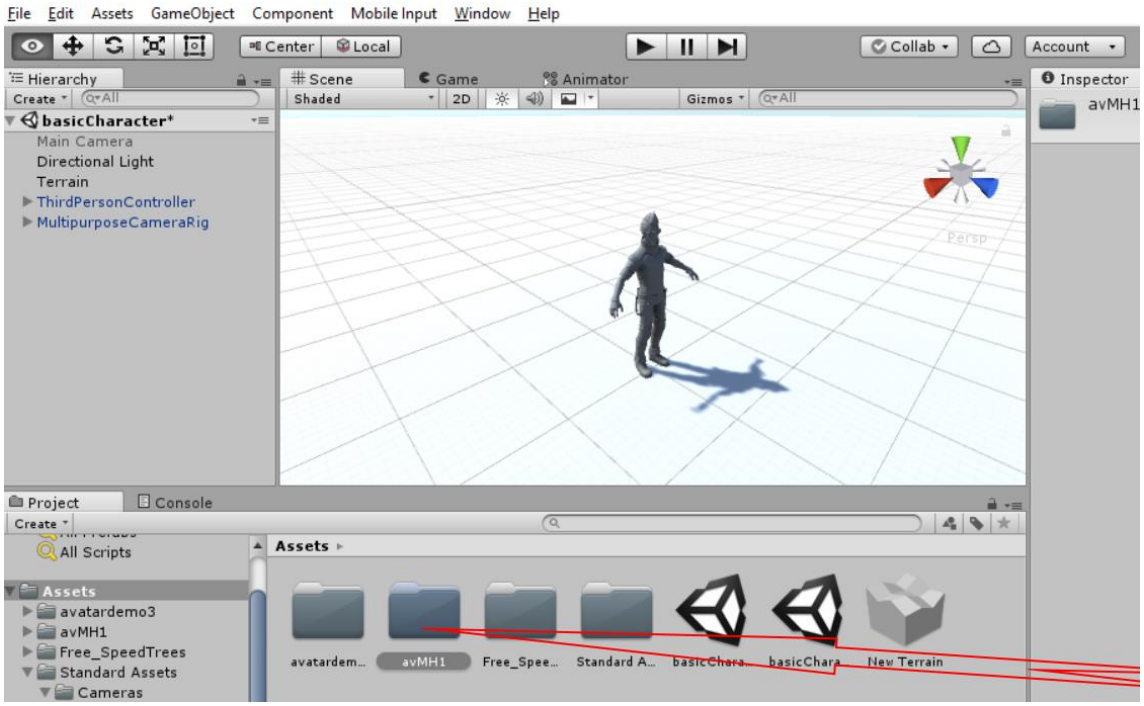


Figure 6 – Importing external files by dragging them into Unity's interface

2.2. Configure Avatar Rig to Unity's Humanoid and get Gestures

Since our makehuman character (avatarMH1) already has a bone rig compatible with unity we only have to inform unity that this is a **humanoid** character so the standard animation gestures are assigned to avatarMH1. This is done in 5 steps marked over Figure 7.

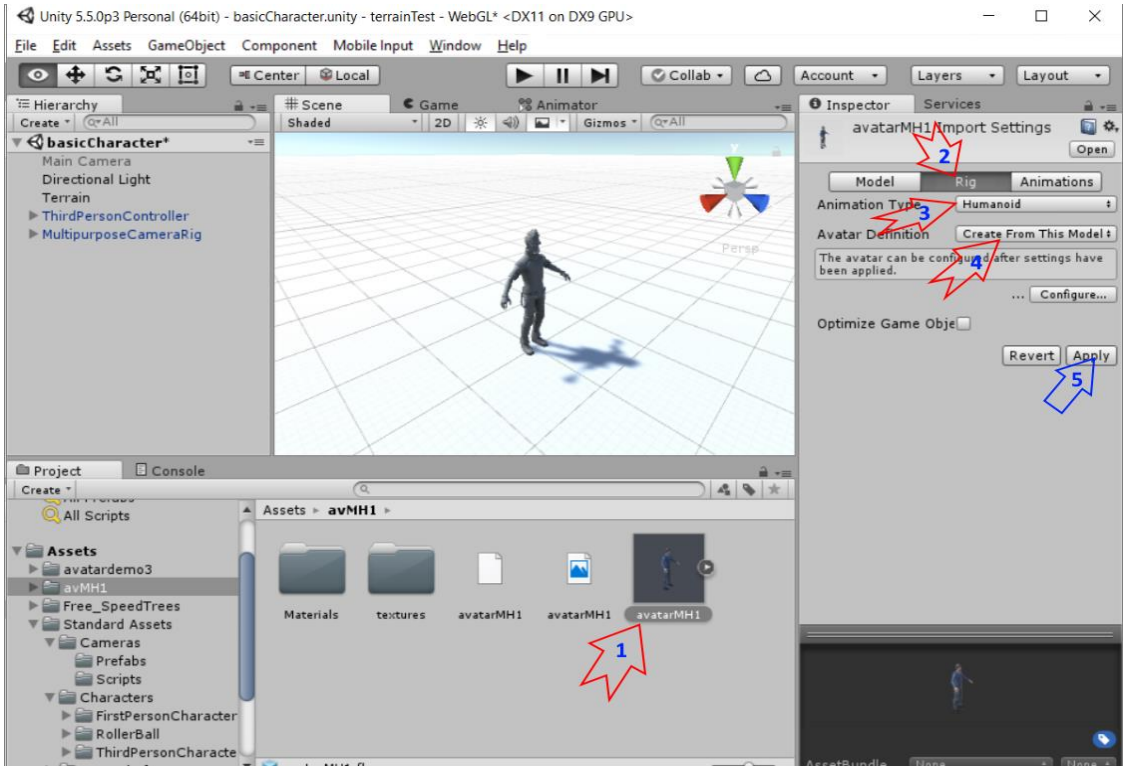


Figure 7 - Define Avatar Rig to humanoid and get Unity's standard animation gestures

1. Open avMH1 folder and click on "avatarMH1" character file to open the inspector tab.
2. On inspector choose "Rig", then
3. On Animation type choose "Humanoid"
4. On Avatar Definition choose "Create From This Model"
5. Finally click on "Apply" to finish the configuration.

After this our character avatarMH1 is configured with animation gestures and is ready to be used as an **avatar** replacing ethan.

3. Replacing ethan

We are three steps away from replacing ethan and using a new avatar.

1. Move avatarMH1 to the Hierarchy inside ThirdPersonController. Position it below ethan elements (take care not to move it inside any of ethan elements) (Figure 8). Now we see both characters superimposed on the 3D scene (If avatarMH1 looks tiny, a 1/10 in size, correct it in Inspector by increasing to 10 the X,Y,Z scale in transform).

2. To disable ethan uncheck its three elements in Inspector (EthanBody, EthanGlasses and EthanSkeleton) (Figure 9). Now only avatarMH1 is visible but its animation movements are still not under our control, a final step is needed.

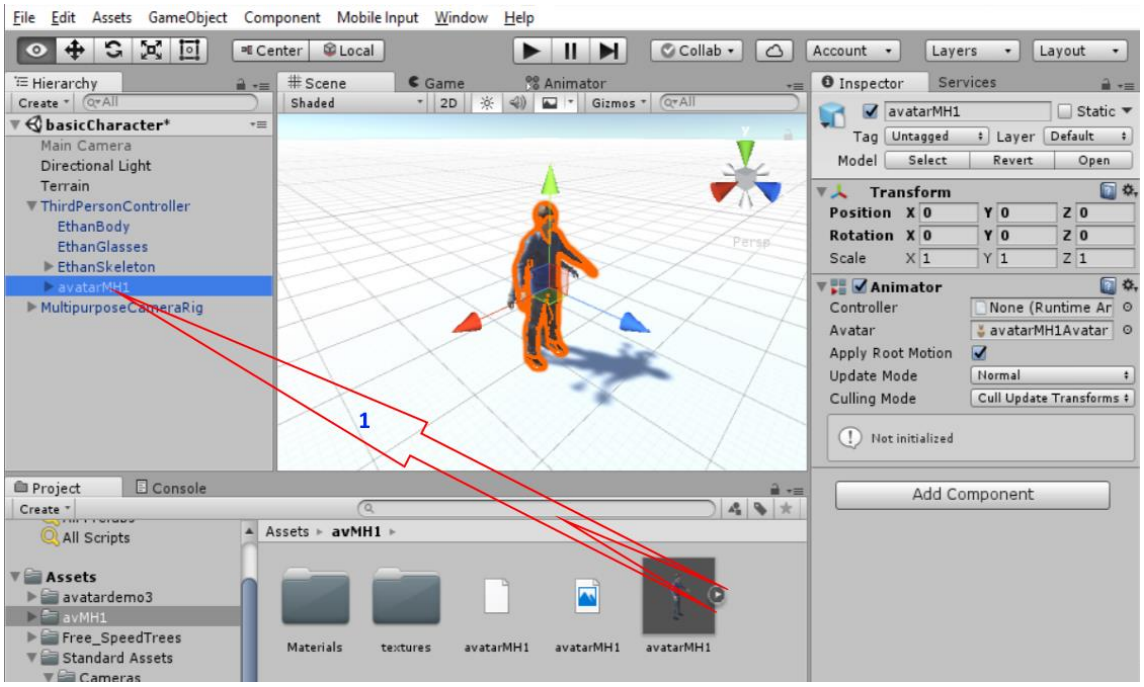


Figure 8 – Adding the new character

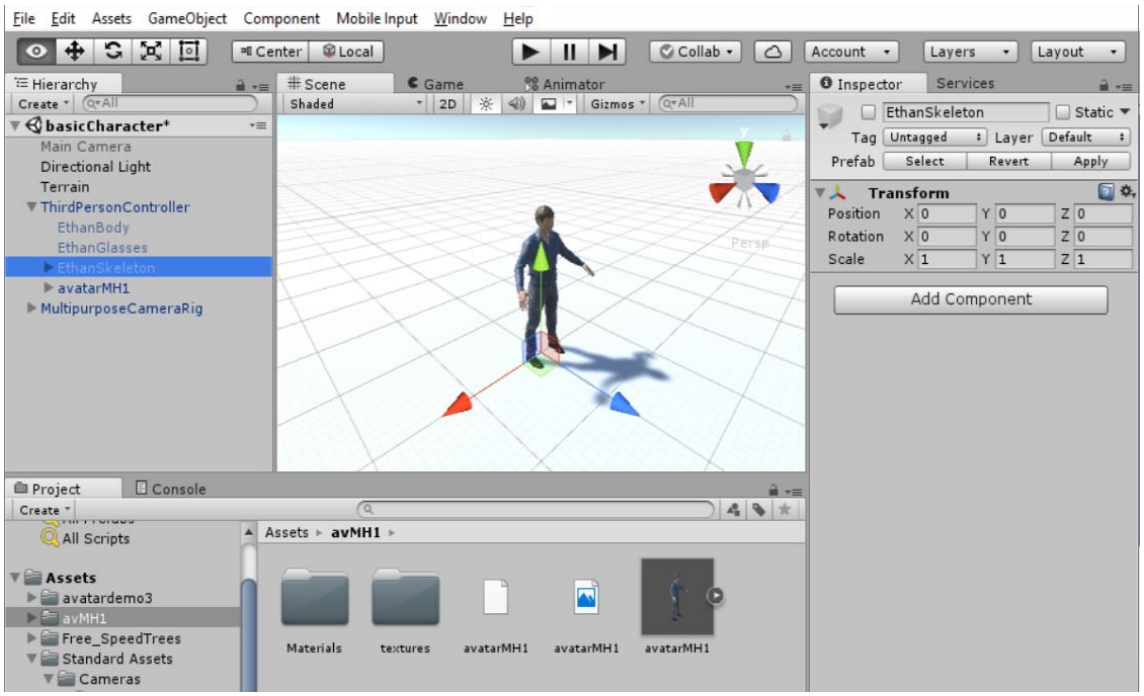


Figure 9 – Disabling ethan

3. Finally, in ThirdPersonController's Animator we need to replace "EthanAvatar" with "avatarMH1Avatar" to let it control the animations of the new avatar (Figure 10).

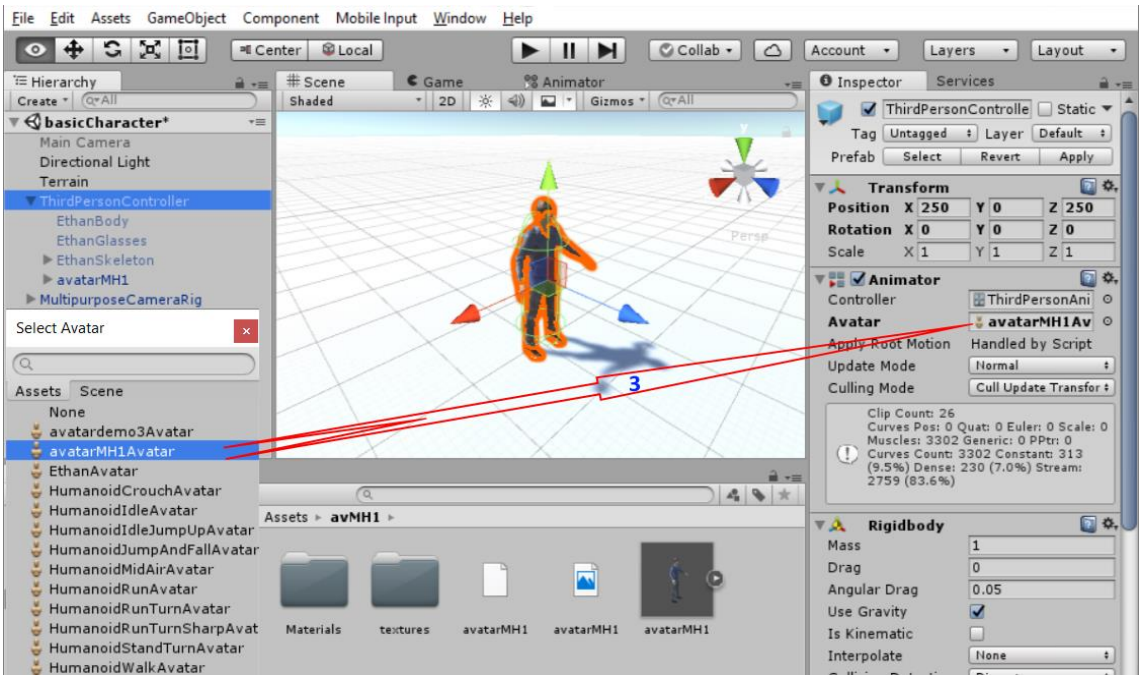


Figure 10 - Activating the animations on the new avatar

Now, click play and enjoy moving around with the newly imported avatarMH1 (Figure 1).

Online demo: http://odisseia.babelx3d.net/unity3d/webgl_makehuman_avdemo3/index.html

4. Enhancing the third person view experience

Mouse Rotating View: a simple enhancing way (Figure 11) is done by adding a component (1) to *Pivot* (2) inside *MultiPurposeCameraRig* like the *simple mouse rotator script* (3) available in "Standard Assets" which allows a controlled looking around the user view.

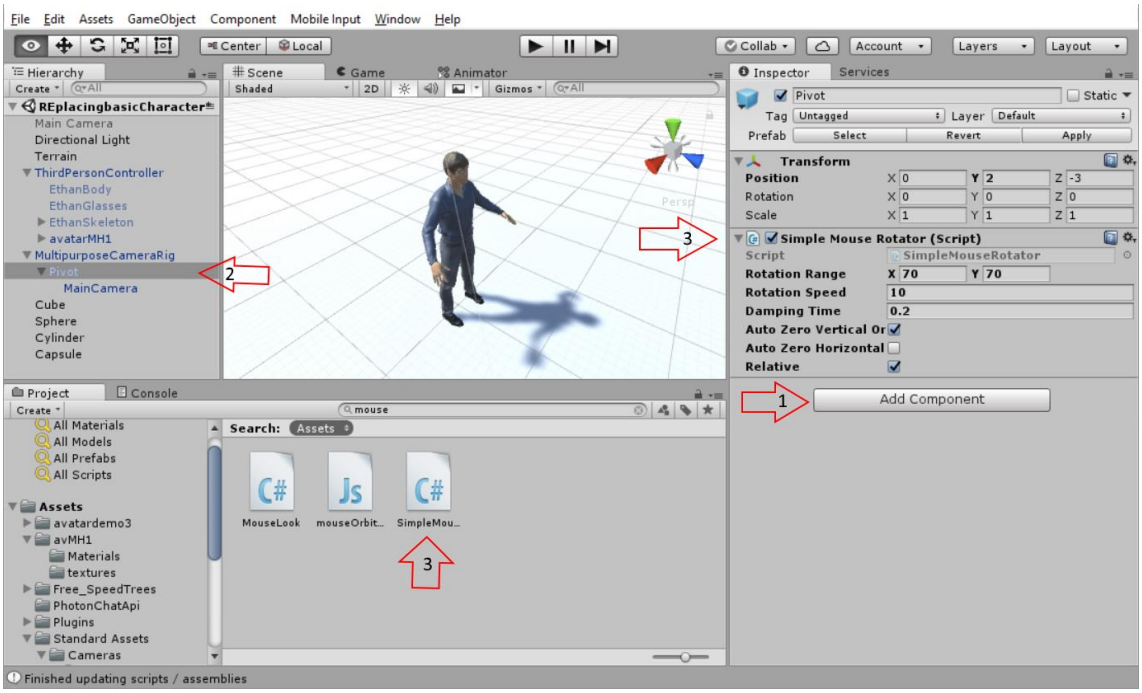


Figure 11- Enhancing the experience with a Mouse Rotating View

Defy to create a "Mouse scroll wheel Zoom": zooming in or out. A cool enhancement (Figure 12) achieved with an easy single instruction script (1) MouseScrollWheelZoom.js that captures the input and is attached to (2) MainCamera. The relevant content inside the script, you are defied to build yourself, is:

```
transform.Translate(Vector3.forward * 2 * Input.GetAxis("Mouse ScrollWheel"));
```

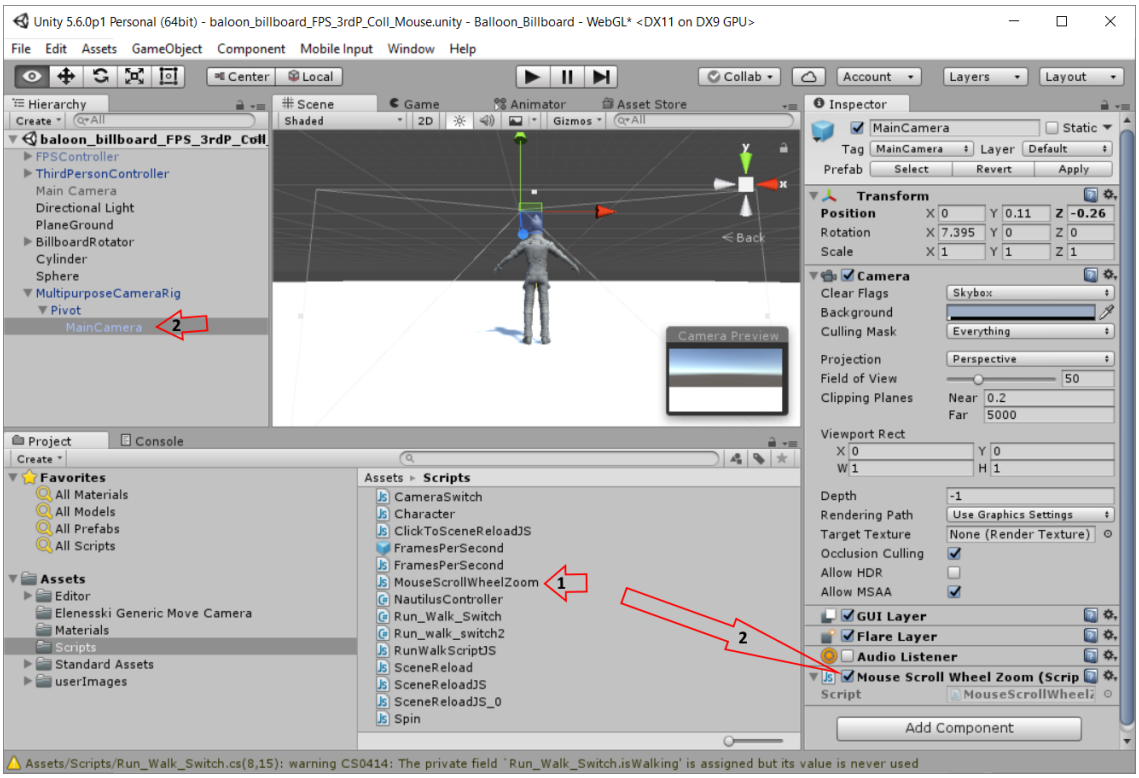


Figure 12 - Enhancing with a Mouse scroll wheel Zoom script.

On **Assets** menu select **Create**> **C# Script** and as name use "*MouseScrollWheelZoom.cs*". The script is automatically created in **Assets/Scripts** folder but we still need to add our instructions. **DoubleClick** on the script name and it will open in Unity's script editor (MS Visual Studio) seen below (Figure 13). Add the code line to `void Update()`, save and drag to **Pivot** node over MainCamera shown in (Figure 12). Save the scene, click play and test this zoom feature.

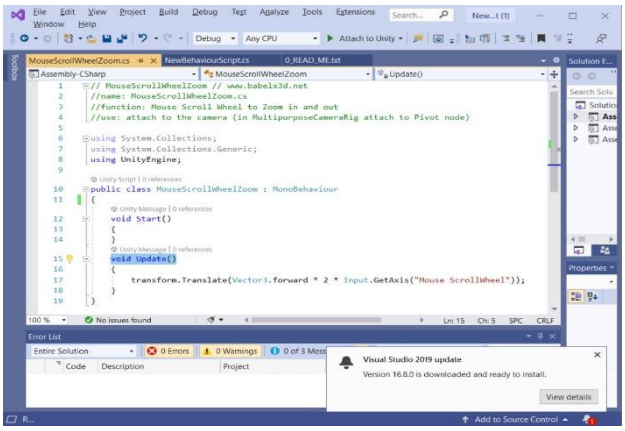


*Figure 13 – MouseScrollWheelZoom script (Hint: To speed up the zoom, use *4* instead of *2*)*

Note 1: in case you are not entirely satisfied with the avatar produced in MakeHuman, it can be imported to blender or 3DS Max and enhanced there with more sophisticated tools. After that can also be exported to Unity (read Note 2 below)

Note 2: as referred in the beginning of this tutorial, the procedures referred after point 1, are valid to import similarly configured characters made by any other software (Blender, 3DS Max, Daz Studio, Mixamo, …)

5. Defy

Using the tutorial info, create your own avatars and answer to this post on babelx3d forum's defy with images and links to your scenes and projects:
http://portal.babelx3d.net/content/tutorial-using-makehuman-characters-unity

6. References

Tutorials are simple texts aiming to motivate. To go beyond look for reference documents:

- Unity Documentation, is the fundamental reference text to learn from
  http://docs.unity3d.com/Manual/index.html
- Unity Learn –Tutorials, videos and training on Unity site.
  http://unity3d.com/learn

- An useful text about import, animate and control your character as a player in Unity:
  Turn your character into a player!
- A comprehensible introductory article about character animation in Unity:
  Unity: Mecanim Animation Basics

- Assets
  We can get Unity assets in many places and even use assets (images, objects, characters, ...) made for other 3D programs but the fundamental reference is the Unity *Asset Store* where we can find hundreds/thousands of free assets to use in our projects: https://www.assetstore.unity3d.com

-//-

Unity3D - Using MakeHuman Characters
By Vitor Cardoso
Pre-release, WIP version 1.0: 11.Jan.2017, for Unity3D ver. 5 and WebGL
Current release 2.1: 12.Nov.2020

Important: this tutorial is updated from time to time so, preferably, read it online or get the latest version from https://1drv.ms/f/s!AiVFncpESHaShZNdh_DaKhqFHCEtWQ

More tutorials are available at https://1drv.ms/f/s!AiVFncpESHaSrUR3FA2OO3Z138HJ