**Requirements :**

1. Python 2.7 (doesn’t work on python 3.x)(Better to install Anaconda and use ipython)
2. makehuman
3. Three.js
4. Django
5. Numpy
6. Pandas
7. Python- qt4

**Definitions:**

**clmh:** (Command line make human) it is used to execute commands which are required for starting makehuman and for importing required packages. And clmh.humanoid is a replacement for G.app.selectedHuman (defaut case).

**manage.py:** manage.py is the main django file, for modifying the django website.

**humanoid:** A makehuman object which contains the mesh and the attributes(height, weight, age, gender, muscle…).

**How to Setup in Ubuntu:**

## sudo apt install python2.7

## sudo apt install python-pip (used to install other packges)

## sudo pip install --upgrade pip //to upgrade the latest pip \*

## pip install django (to make makehuman a website)

## pip install numpy (required by makehuman for mesh transformation calculations)

## pip install pandas (used pandas dataframe in the height conversion)

## sudo apt install python-qt4

## open dir in terminal and python manage.py runserver //to run the Django Basic server

**Testing**

got to directory in terminal

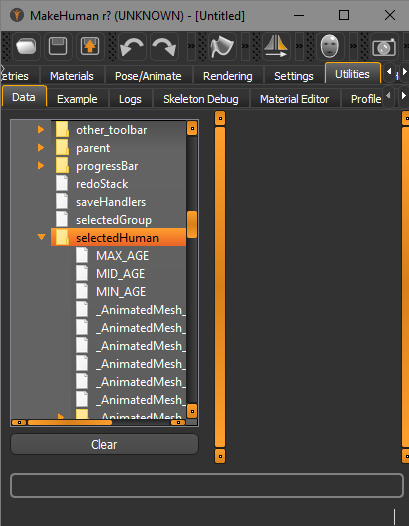
python enter

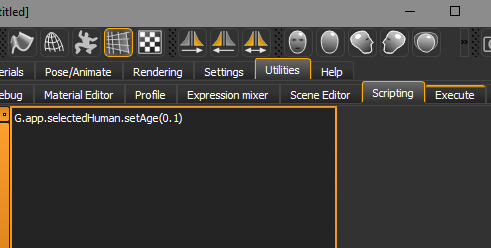
import clmh enter

clhm.fpa //to check absolute path where object will b stored

clmh.humanoid.setAge(0.7)

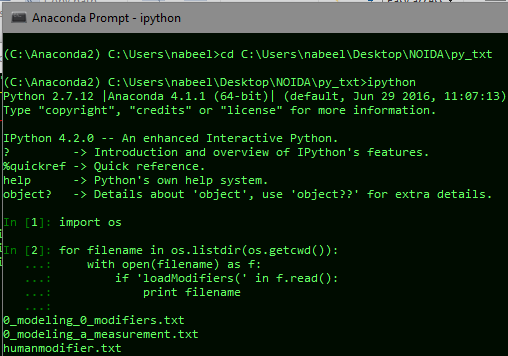
**Guidelines**:

* Avoid involvement of G-a global object used by makehuman (thus avoiding GUI and additional features), try to modify directly the human instance and use it.
* Unlink the ‘progress’ bar, as it is only a GUI feature.
* To avoid errors regarding G.app.selectedHuman. Import clmh replace G.app.selectedHuman with clmh.humanoid.
* Many functions and attributes can be accessed here

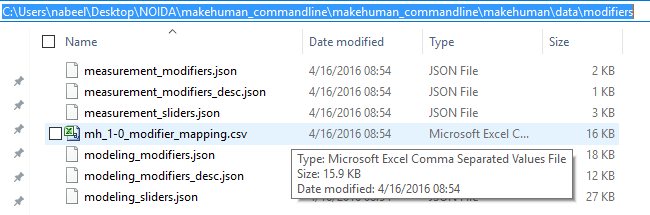
And can be tested and ran in execute tab.

Search technique :

For example searching for ‘loadModifiers(’ in the MakeHuman python code



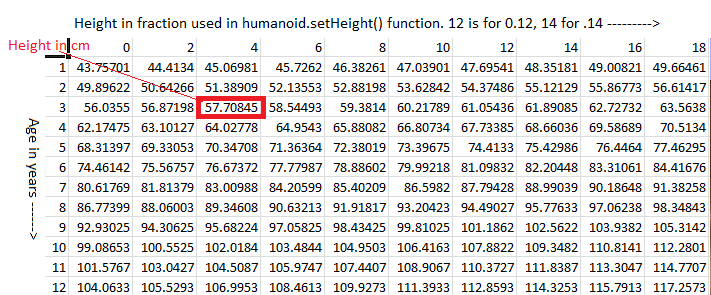
All the Modifiers



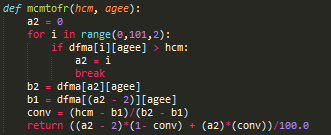
Note: Similiarly all the poses in data\pose

**Height in cm**

All the variables fed in makehuman are between 0 and 1. But after the mesh is built makehuman provides an option to see the height of the humanoid in cm. Using this ‘ma.p’ and ‘fem.p’ 90x51 tables are made.



This table is used to convert the height in cm to fraction b/w 0 and 1 by finding the weighted mean.



**Django Module (MusicApp)**

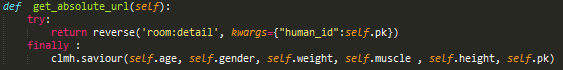
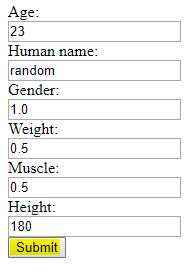
Open MusicApp directory in terminal

And run ‘python manage.py runserver 7000’ (7000 is the port number)

Open localhost:7000/room and click on Make Human to get the below form.

Made by modifying tutorial <https://www.youtube.com/watch?v=qgGIqRFvFFk&list=PL6gx4Cwl9DGBlmzzFcLgDhKTTfNLfX1IK>

When Submit button is pressed, get\_absolute\_url() from room/models.py is executed, which in turn executes clmh.saviour() and redirects to the detail url of each human.



savior() saves the humanoid obj file (named as pk.obj where pk is the primary key in the database) in the folder room/static/room/assets/models/ , which is then loaded in three js to render. 

**Pending**:

**Cloathing** : hard cloathing can be achieved by using makehuman, but soft cloathing not achieved

**Face Generation** : **Large Pose 3D Face Reconstruction from a Single Image via Direct Volumetric CNN Regression**

http://aaronsplace.co.uk/papers/jackson2017recon/

**Pose** : No Breakthrough

**Weight in Kgs :**

INPUTS:

height (in cm)

volume (in cm^3) {doesn't currently exist; for testing I used a Blender add-on}

gender (between 0 and 1; 0=female, 1=male)

muscle (between 0 and 1; 0=non-muscular, 1=muscular)

weight (between 0 and 1; 0=thin, 1=fat)

age (between 0 and 1; 0=0years, 0.5=25years, 1=90years)

frame (between 0 and 1) {doesn't currently exist, optional}

CONSTANTS:

density.bone.max = 1.75 g/cm^3

density.fat = 0.9 g/cm^3

density.muscle = 1.05 g/cm^3

frame.range = 0.25 {maximum increase and decrease of bone mass due to frame}

age.bone.min = 0.8333 {bone density at age 90 as ratio to peak bone density}

age.bone.max.pos = 0.5 {age at maximum bone density}

VARIABLES:

mass.bone.male = (0.07\*(height^2.33))

mass.bone.female = (0.03\*(height^2.48))

mass.bone.gender = (mass.bone.male\*gender)+(mass.bone.female\*(1-gender))

frame.bone.modifier = (((1+frame.range)-(1-frame.range))\*frame)+(1-frame.range)

age.bone.modifier = 1+(((age.bone.min-1)\*((age-age.bone.max.pos)^2)/((1-age.bone.max.pos)^2)))

mass.bone = mass.bone.gender\*frame.bone.modifier\*age.bone.modifier

density.bone = density.bone.max\*age.bone.modifier

volume.bone = mass.bone/density.bone

adiposity.male = (0.9323\*weight)+(0.1129\*muscle)-(1.0054\*weight\*(muscle^0.5))+0.03

adiposity.female = (1.3058\*(weight^1.2))+(0.2591\*muscle)-(1.471\*weight\*muscle)+0.15

adiposity = (adiposity.male\*gender)+(adiposity.female\*(1-gender))

volume.fat = (adiposity\*density.muscle\*(volume-volume.bone))/((adiposity\*density.muscle)+density.fat)

mass.fat = volume.fat\*density.fat

mass.muscle = (volume-(volume.bone+volume.fat))\*density.muscle

OUTPUTS:

mass.total = (mass.bone+mass.fat+mass.muscle)/1000 [in kilograms]

bodyfat = mass.fat/(mass.total\*1000)

bmi = mass.total/((height/100)^2)