Main Program:

1. Functions
   1. ???
2. Main
   1. Asks what someone want’s to generate, first indicated by a number from 1 to 4, then by name (possibly use or so we don’t lose functionality)

Objects:

1. Body
   1. Properties
      1. Name
         1. String, the name of the body, defaults to “Body”
      2. Mass
         1. The mass
         2. If python likes large numbers, go for it, otherwise, figure out a way to use sci notation, or assume it’s x10^6 Kg
      3. Radius
         1. The radius of the body
         2. In Km
      4. Orbiting, single body
      5. Orbited\_By list, can be empty
   2. Functions
      1. Functions to get/set all properties, properties are going to be private
      2. Unimplemented calc\_values functions, so each sub type can implement it
      3. Init with all properties specified
      4. Equals ?
      5. [Property]\_ToStr
         1. Prints out the value of a property with sci units at the end
      6. Is Orbited?
         1. Boolean indicating if the Orbited\_By list is empty
2. Star
3. Star System
4. Planet
5. Moon

Versions:

1. Version 1
   1. Random Star/Planet/Moon generation
   2. Console interface
2. Version 2
   1. Add Random star system generation
3. Version 2.5
   1. Make a gui
4. Version 3
   1. Add ability to specify details of generation
      1. Specific luminocity star
      2. Specify rocky or gas
      3. Moon mas, or radius ?
      4. Number of stars in a star system
   2. Set up the ability to have bodies orbit other bodies, and be orbited by bodies
5. Version 4
   1. Add the ability to force generate a star with a habitable planet
   2. ? generate a star with the maximum number of habitable planets, and force them to be habitable ?
   3. Implement version 3 and 4 features for gui
6. Version 5
   1. Add the ability to create custom stuff, by specifying all details
   2. Be able to generate or calculate stuff people don’t want to during this process
   3. Add this to gui