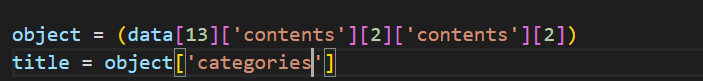
Text

Description automatically generated

* The structure is the primary cat here reference but they have the subcategory in the dict key category which gives me a list object which has the descending cats. Grabbing title of each one will give me the objects. I can create a sefer object for each thing here by just grabbing all the things in one cat and link them to the cats with a foreign key.

In this data[13]['contents'][2]['contents'][1])

* The first index is the primary cat. The second is the various subcats. An the third the actual objects.
* Elegant way to do it is to write the program to grab the secondary cats and create them If not already there.
* Need a nested set of loops where I loop through ever object of the subcategory before incrementing the subcategories. I move the last 0 through its total contents before resetting the two. Before resetting the first one if I want to make the full script.
* Can just index the final object for title, or categories like so
* 
* The stuff is in scripts and the remaining work is to nest the loops and run it
* The id field when printed returns text and I was trying to get a pk
* The api returns don’t totally match the layout on the website. There is no [0][3][0] which means that the targum cat on the website is stored somewhere else. The q is where and how to have checked this out earlier.
* Not true there are 7 ys for the z[0] but they aren’t getting picked up. The reason is that all the targumim are in one object. This is cuz targumim is really itself which has the objects one level down. But that means that I really need a program that can move down until it hits the level of the objects. There is a real skill which is how to work to get all of a certain type of object within json. There is another interesting approach here which involves moving everything to a GUI and manually manipulating but that mainly matter for the scale of the data.
* There is a bigger question which is how I want to organize this. Cuz these are my deepest seated categories. Hit the same issue with rishon on Mishnah. Can also split into separate programs cuz itll all work just with the later starting pt and the right cats
* There are issues in the tshvot part and have to think about how to test this also. There are issues in the excel with the total number and I dotn remember how I got it anyways
* Part of the issue is that its not spitting back an error its just not finding any objects
* Tachlis can just run manually at that level but not 100 percent easy cuz have to iterate other vars. Cant just add more nesting cuz its not failing silently
* GOALS find our where the pk is in the objects that don’t have a pk field. Im almost sure that im picking up the pk just getting it spit back as a name cuz of the function that spits back names.
* How to capture the things at a deeper level
* Beginning to sprawl and am going to need to break some of it into functions.
* BIG LESSON HERE IS THAT TRY EXCEPTS ARE A VERY BAD WAY TO DO FLOW CONTROL CUZ THEY WILL FAIL SILENTLY BY KICKING BACK TO A PREV LOOP. AND TO SAVE EVERY SIGNIFICANT PIECE OF WORK
* Some of the object are more levels down and so they aren’t getting picked up by the loop
* hfaText

  Description automatically generated
* these are the targum objects. Running the program with this setup iterates through all the targumim but I really want that as a tertiary object here. As well I need to iterate thru the higher cats for the objects. So what I need here are more loops to cover the new things. A check if there are objects at the end of the y loop meaning I loop through my ys until I hit nothing the problem with this is that there is no error so it wont pick up when to go down so what I really need to do is check at each point in the y loop of there is a downstream object and than nest two more loops of the same type.
* I can run the program manaually for every situation where I have this setup using the program I have and iterating the
* This is getting too complex to do manually cuz there are too many possible levels unless I just do one program for each level

Text

Description automatically generated

* There are multiple potential levels within each level. Might just leave this with a partial list of objects and wire some more of it together. Having this file in whats covered by git creates issues