Showing SubCategory and SubSubCategory simultaneously

**models.py**

class SubCategory(models.Model):

sub\_cat\_image = models.ImageField(upload\_to='SubCategory',null=True, blank=True)

sub\_cat\_title = models.CharField(max\_length=255)

sub\_cat\_overview = models.CharField(max\_length=500)

sub\_cat\_parent = models.ForeignKey(Category, on\_delete=models.CASCADE)

sub\_cat\_slug = models.SlugField(unique=True)

class SubSubCategory(models.Model):

subsub\_cat\_title = models.CharField(max\_length=150)

subsub\_cat\_parent = models.ForeignKey(SubCategory, on\_delete = models.CASCADE)

subsub\_cat\_slug = models.SlugField(unique=True)

**views.py**

def tutorials(request, category, sub\_category, tutorial):

sub\_cats = SubCategory.objects.filter(sub\_cat\_parent\_\_cat\_slug = category)

return render(request, 'tutorials/tutorials.html', {'sub\_cats': sub\_cats, })

**tutorials.html**

{% for sub\_cat in sub\_cats %}

<button class="dropdown-btn"> {{ sub\_cat.sub\_cat\_title }}

<i class="fa fa-caret-down"></i>

</button>

<div class="dropdown-container">

{% for subsub\_cat in sub\_cat.subsubcategory\_set.all %}

{% url 'tutorials:tutorials'

subsub\_cat.subsub\_cat\_parent.sub\_cat\_parent.cat\_slug

subsub\_cat.subsub\_cat\_parent.sub\_cat\_slug

subsub\_cat.subsub\_cat\_slug as tutorial\_link %}

<a href="{{ tutorial\_link }}"

class="dropd-links

{% if request.path == tutorial\_link %}active-link{% endif %}"

>

{{ subsub\_cat.subsub\_cat\_title }}</a>

{% endfor %}

</div>

{% endfor %}