

# Usage of Category Service

---

The following five APIs are provided currently:

1. `getCategories(type=None)-> list` :  
If no type is specified, it will return all categories by default; otherwise, it will return all categories of the type.
2. `getCategoryById(id)-> Category` : Use id to get the category item.
3. `getAttributesByCategory(category)-> dict` :  
Use specified category to get category attributes of the category. It will return in the format of dict. The key of the dict is the name of the CategoryAttribute and the value is the CategoryAttribute itself.
4. `getAttributeValues(instance, attributeValueModel)-> dict` :  
Use specified entity, such as EvSKU, Product, etc., objects with their own category, and AttributeValue class of the object to get all AttributeValues of the object.

It will return in the format of dict. The key of the dict is the name of the CategoryAttribute and the value is the value of the AttributeValue itself.

For example, if I have a Product objects called p, it has two CategoryAttributes and the (name, value) are ('attr1', 1), ('attr2', 'attr 2 val') respectively. The return value of using `getAttributeValues(p, ProductAttributeValue)` will be a dict whose value is `{'attr1': 1, 'attr2': 'attr 2 val'}`.

5. `saveAttributeValues(instance, attributeValueModel, instanceValueDict)-> object` :  
A value dict which uses specified entity, AttributeValue class and value of the entity. It returns the entity.

For example, there is a Product objects called p, it has two CategoryAttributes which are 'attr1' and 'attr2' respectively. Now receive a request to set the following values to p: `{'attr1': 1, 'attr2': 'attr 2 val'}`, then build a dictionary `d = {'attr1': 1, 'attr2': 'attr 2 val', 'editor': request.user}` with dict as its value. **Please pay attention to set editor!**

Next, use `saveAttributeValues(p, ProductAttributeValue, d)` to complete the update. The return value is p.

If you have any doubts about usage, you can refer to `TestCategoryService.testCategoryService()` in `category/tests.py`. It is a test of CategoryService operated with the APIs which can be regarded as an example of using these APIs.

If you still have some questions, or feel that there are some problems when using above APIs, please contact with Jacky.