Demonstrate ability to use and customize categories

Describe category properties and features.

Category features:

- can be anchored/non-anchored. When anchored, all products from subcategories "bubble up" and show up in the category listing even when *not assigned* explicitly.
- include in menu switch
- display type - products only, cms block only, both
- you can assign direct category products and set their individual positions. Directly assigned products always show before "bubbled up" for anchor.

How do you create and manage categories?

- as always, prefer using repository over directly saving model
- Magento\Catalog\Model\CategoryRepository
- Magento\Catalog\Model\CategoryManagement - getTree, move, getCount
- category.move(newParent, afterId) – old approach

Category model - few changes since M1

- getTreeModel, getTreeModelInstance

- move - event `catalog_category_move_before` , `catalog_category_move_after` , event `category_move` , indexer.reindexList, clean cache by tag

- getProductCollection

- getStoreIds

- getPathIds, getLevel

- getParentId, getParentIds - without self ID

- getParentCategory

- getParentDesignCategory - Category

- isInRootCategoryList

- getAllChildren - = by default joined ids ("2,3,4…") + parent id, recursive, only active

- getChildren - joined ids ("2,3,4…"), only immediate, only active

- hasChildren - recursive, only active

- getAnchorsAbove - array of ids - parents that are anchor

- getProductCount - only directly assigned, all

- getCategories - tree|collection. recursive, only active, only include_in_menu, rewrites joined

- getParentCategories - Category[], only active

- getChildrenCategories - collection, only active, only immediate, include_in_menu*, rewrites joined

Describe the category hierarchy tree structure implementation (the internal structure inside the database).

- Every category has hierarchy fields: parent_id, level, path

- To avoid recursive parent-child traversing, flat category path is saved for every category. It includes all parents IDs. This allows to optimize children search with single query. E.g. to search children of category id 2 and knowing its path '1/2', we can get children:

```
select * from catalog_category_entity where path like "1/2/%" :
```

```
+-----------+-----------+--------------+-------+
| entity_id | parent_id | path         | level |
+-----------+-----------+--------------+-------+
|         3 |         2 | 1/2/3        |     2 |
|         4 |         3 | 1/2/3/4      |     3 |
|         5 |         3 | 1/2/3/5      |     3 |
|         6 |         3 | 1/2/3/6      |     3 |
|         7 |         2 | 1/2/7        |     2 |
|         8 |         7 | 1/2/7/8      |     3 |
....
```

What is the meaning of parent_id 0?

Magento\Catalog\Model\Category constants:

- ROOT_CATEGORY_ID = 0 – looks like some reserved functionality, has only one child

- TREE_ROOT_ID = 1 – all store group root categories are assigned to this node

- These categories are not visible in admin panel in category management.

- When creating root category from admin, parent_id is always TREE_ROOT_ID = 1.

- Magento\Store\Model\Store::getRootCategoryId:

  - TREE_ROOT_ID = 1 when store group is not selected
  - store group root category ID
- store group root category allows only selecting from categoryes with parnet_id = TREE_ROOT_ID = 1

How are paths constructed?

Slash-separates parents and own ID.

Which attribute values are required to display a new category in the store?

- is_active = 1
- include_in_menu = 1, but even if not displayed in menu, can be opened by direct link
- url_key can be created automatically based on name

What kind of strategies can you suggest for organizing products into categories?