

Lab 3 – View and search products - I

Aim

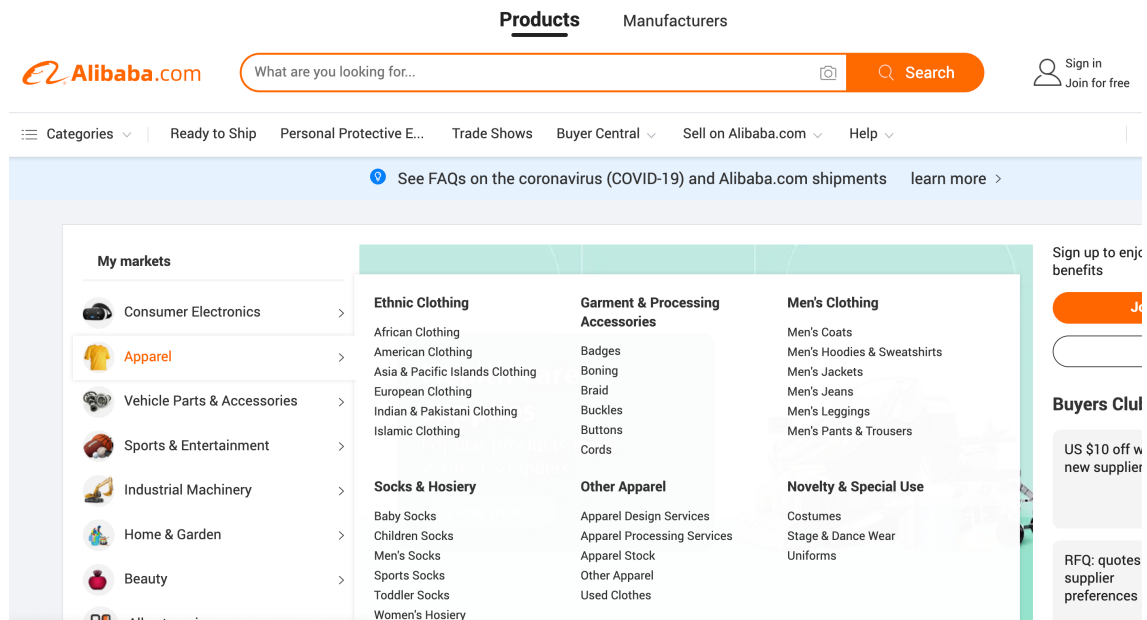
The aim of this lab is to learn how to setup the product related database and view the categories through web.

Tips:

1. If you are not sure why you are doing something, ask a TA. This is what they are here for.
2. The M-Dev-Store online videos are good references while our labs have different focus. If you want to be an expert, you are recommended take both labs and on-line videos.
3. The forums @ LMO are available for questions and discussions.
4. These labs are expected take more than the 2 allocated hours. You should complete them in your own time before the next lab. Practice makes perfect!

Side-bar with bootstrap:

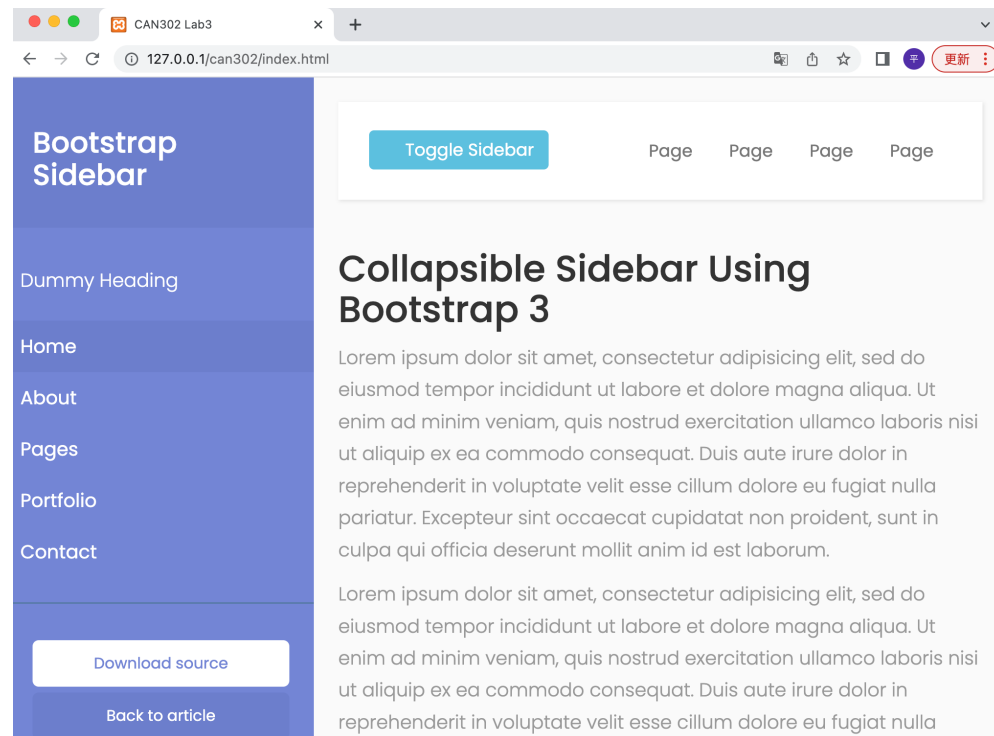
1. Search is the essential function for any E-commerce site. Customers always start their shopping from search. The products information needs to be stored in database to support the search. A typical homepage of E-commerce web has been shown as following:



There are two ways for search, one is the general one on the top navigation part – general search, it is more like the search engine (i.e., Google or Baidu). Another one is the left part search by categories.

Bootstrap has the pre-defined “navigation” element. It can bring user directly to many functions. While it does not have the “left-bar” style. We need to customize the styles to “left-bar” one as you seen in the above snapshot.

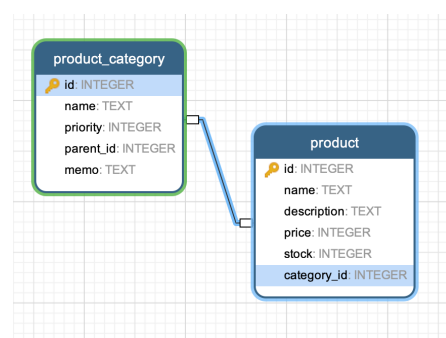
There are many on-line samples for the sidebar. Due to the native property of HTML, they are all open-source and can be directly used. The following example from <https://bootstrapious.com/p/bootstrap-sidebar> will be used in our lab:



You many download the source code and figure out how it works. This lab will use its CSS file directly for left-bar to show categories.

Organize the products in Database:

- To show the products on webpage, their information should be stored in the database accordingly. To support the search by categories, at least a table for category and a table for product are needed. It is a 1 to N relation and a foreign key should be used to keep the constraint. A sample design is as:



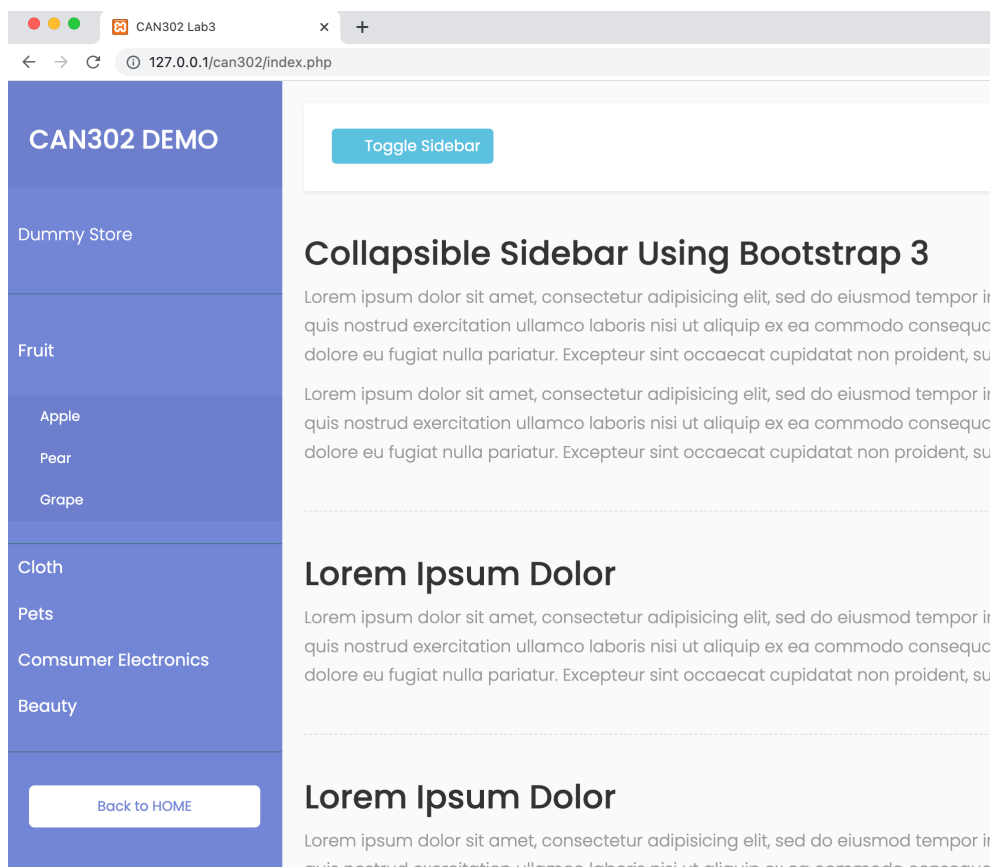
The PHP code to generate the database and tables are as:

```
sqlite.php
1 <?php
2 $db = new SQLite3('mysqlitedb.db');
3 $db->exec("CREATE TABLE product_category(id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT NOT NULL, priority INTEGER NOT NULL, parent_id INTEGER NOT NULL, memo TEXT)");
4 $db->exec("CREATE TABLE product(id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT, description TEXT, price INTEGER, stock INTEGER, category_id INTEGER, FOREIGN KEY(
5 category_id) REFERENCES product_category(id)) ON DELETE RESTRICT ON UPDATE RESTRICT");
```

If you do not like the PHP way, you can also use other database tools to generate it.

Show the categories by PHP:

3. To show the categories on webpage, we need to replace the html code in part1 with the dynamic data from database from part2.



Please notice the categories on left-bar are fetched from database. Obviously, there are different levels for the categories. How to arrange them properly on the webpage? “Recursion” is a good choice. Please try to make your code based on the knowledge you have got from previous labs.

The demo code will be released afterwards. But try to make your own which **IS** time consuming but **DO** help on the understanding of the algorithm in-deep.