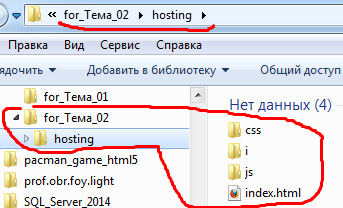
**Тема 2. Селектори**

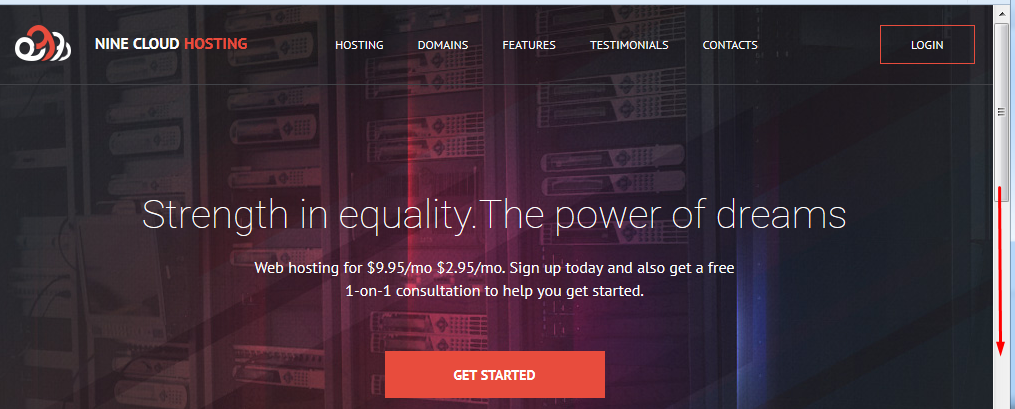
Для своєї роботи підключимо наперед заготовлений сайт.

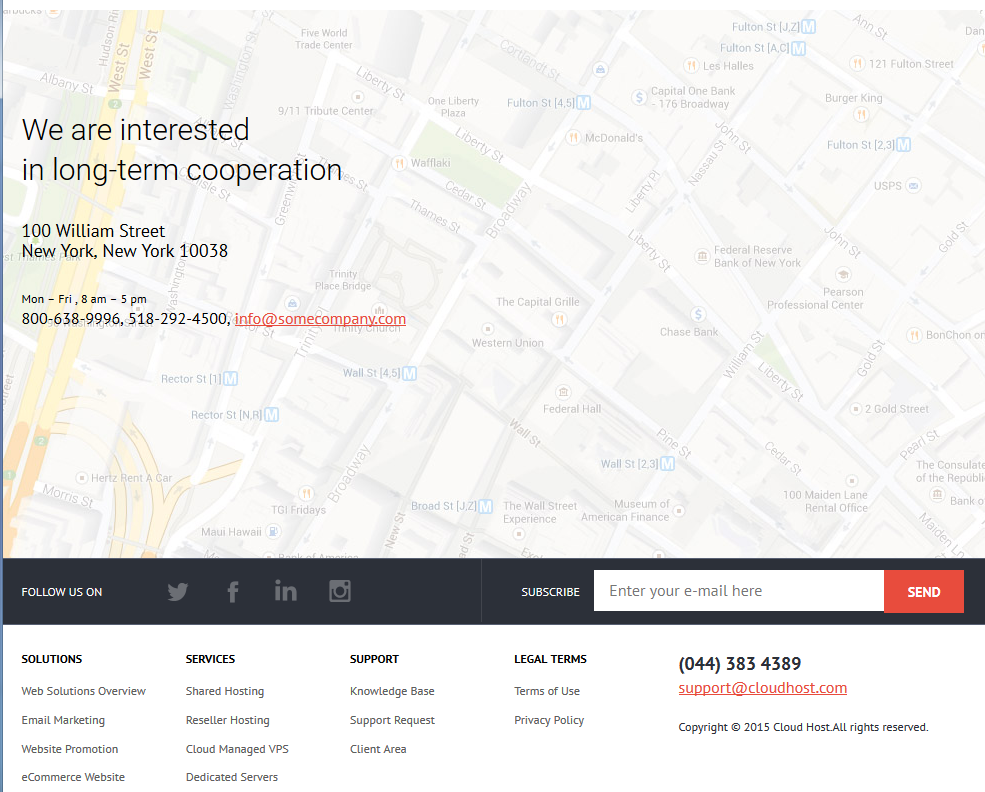
Буде краще, коли ми будемо тестувати JavaScript на реальному прикладі.

Отже, будемо використовувати наперед заготовлений сайт

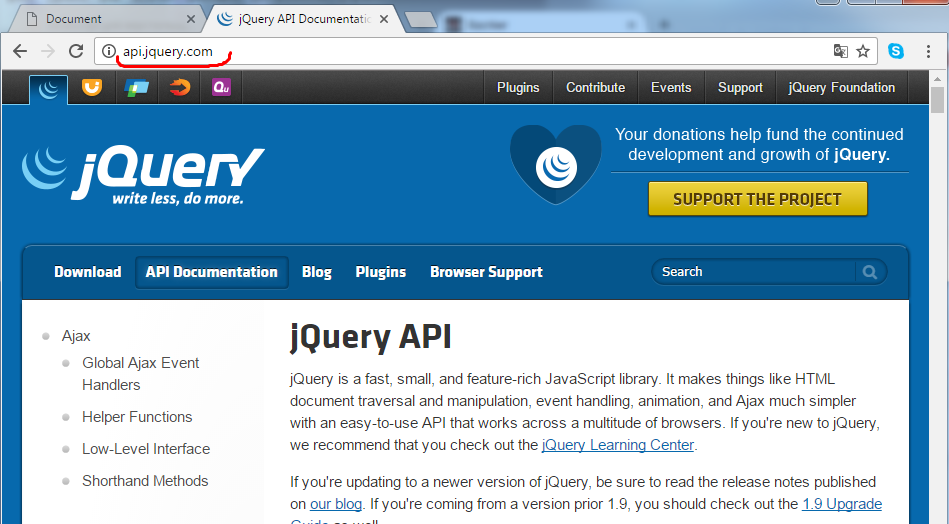


Запустимо index.html – подивимось на вигляд цього сайту





Відкриємо ще один сайт

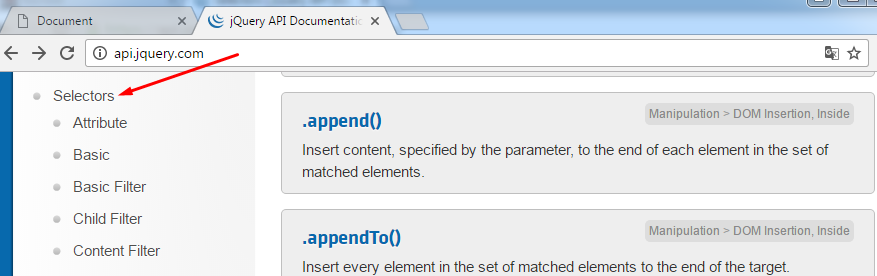


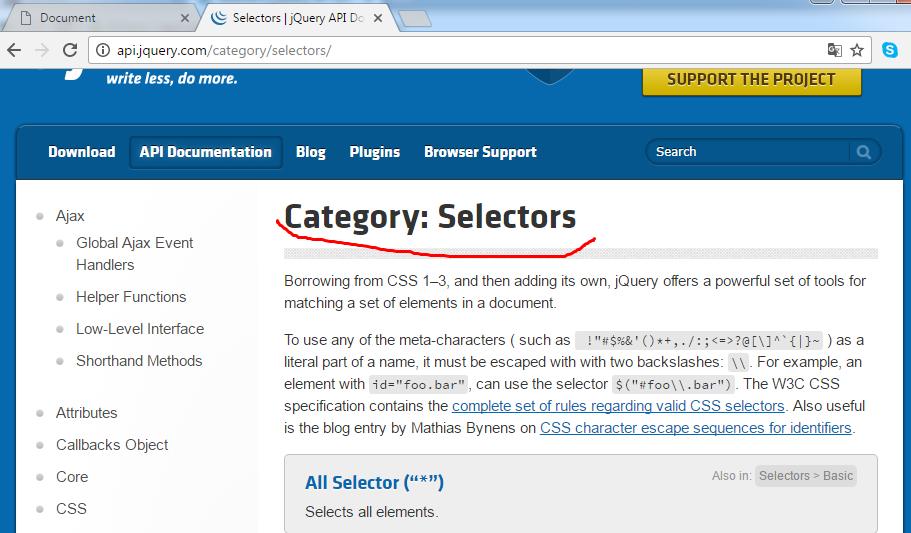
Цей сайт містить повну документацію по jQuery.

Тут можна знайти опис функцій, методів, подій, як робити вибірку і т.д.

Таким чином, цей сайт можна буде використовувати тоді, коли буде щось незрозуміло.

Відкриваємо розділ Selectors

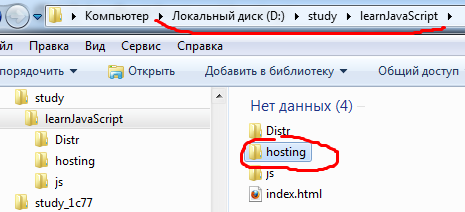


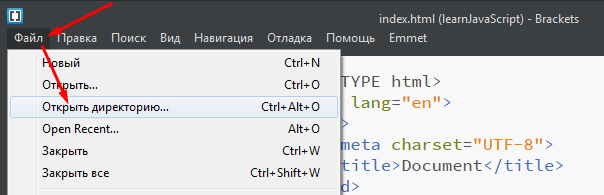


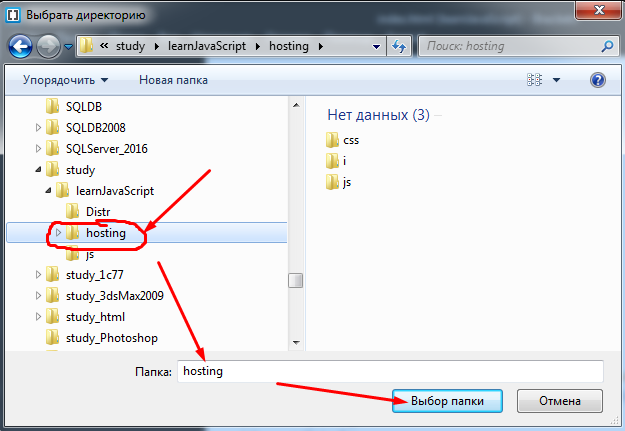
Тут ми можемо, наприклад, подивитись, як отримувати доступ до будь-яких елементів.

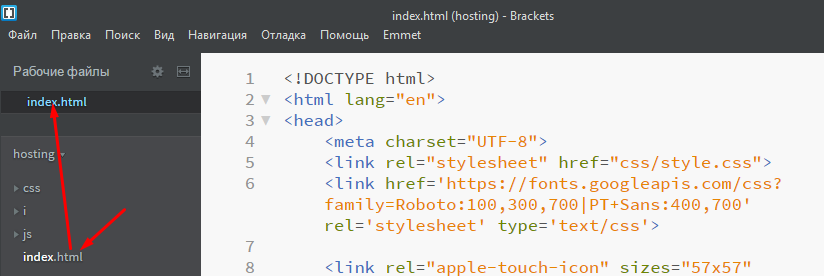
Доступ до елементів ми можемо отримати через атрибути, через клас, через id, через псевдокласи і т.д.

Переходимо до редагування сайту (через редактор brackets).

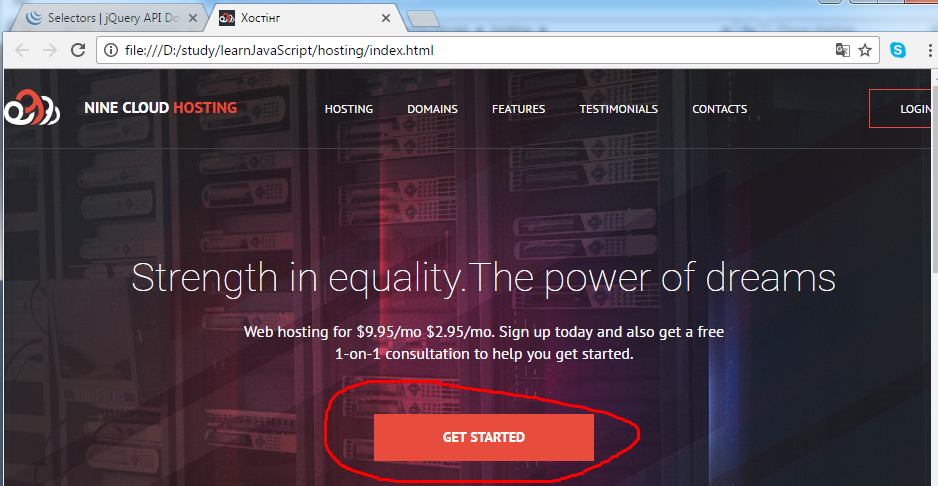




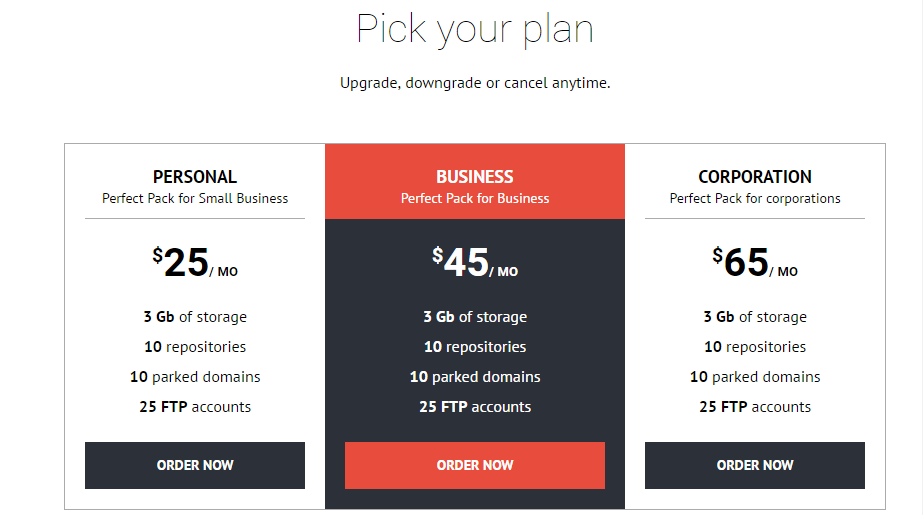




Відкриємо сайт в браузері

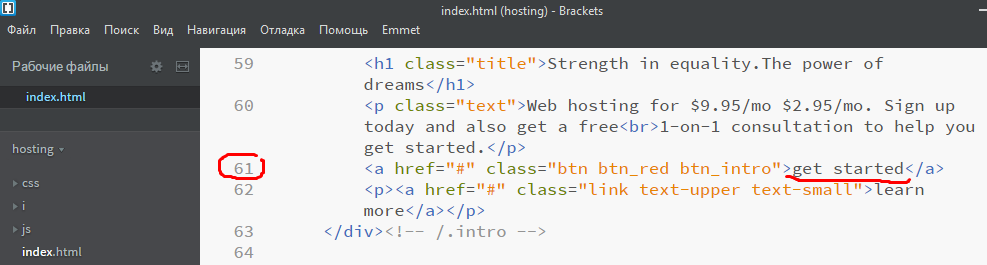


Нехай ми хочемо одержати доступ до кнопки GET STARTED. Ну а далі, коли ми будемо натискати на цю кнопку, то сторінка має скролитись донизу до блоку Pick you plan (Виберіть тарифний план).



Переходимо до html і шукаємо кнопку в розмітці.

Знаходимо це в рядку 61



Тут ми бачимо 3 класи, які відповідають за зовнішнє оформлення кнопки.



Ми можемо доступитись до елемента за допомогою будь-якого із цих класів.

Але так поступати не рекомендується.

Всі ці 3 класи відповідають за стилеве оформлення. І якщо ми js прив’яжемо, наприклад, до класу btn, а потім вирішимо, що даний клас тут непотрібний (вилучимо чи змінимо його назву) – змінимо CSS, то нам прийдеться редагувати ще й js.

Адже виходить, що ми прив’язали js-функціонал до класу, який відповідає за стилі.

Отже, не рекомендується прив’язувати js-функціонал до класів, які відповідають за стилі!!!

Краще писати додатковий клас з префіксом js.

Так буде простіше всім. Особливо тоді, коли ми працюємо не в одиночку, а в офісі, в команді.

Ми можемо переглядати CSS-розмітку. І коли ми побачимо три класи звичайних (без префіксу js), то ми відразу будемо робити висновок, що ці класи відповідають за зовнішнє оформлення елементу.

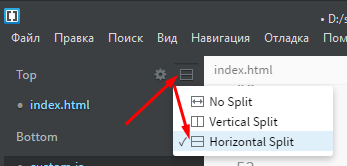
А якщо ми побачимо, наприклад,



то відразу зробимо висновок, що даний клас відповідає за функціонал в js. І через цей клас ми зможемо знайти потрібний нам елемент.

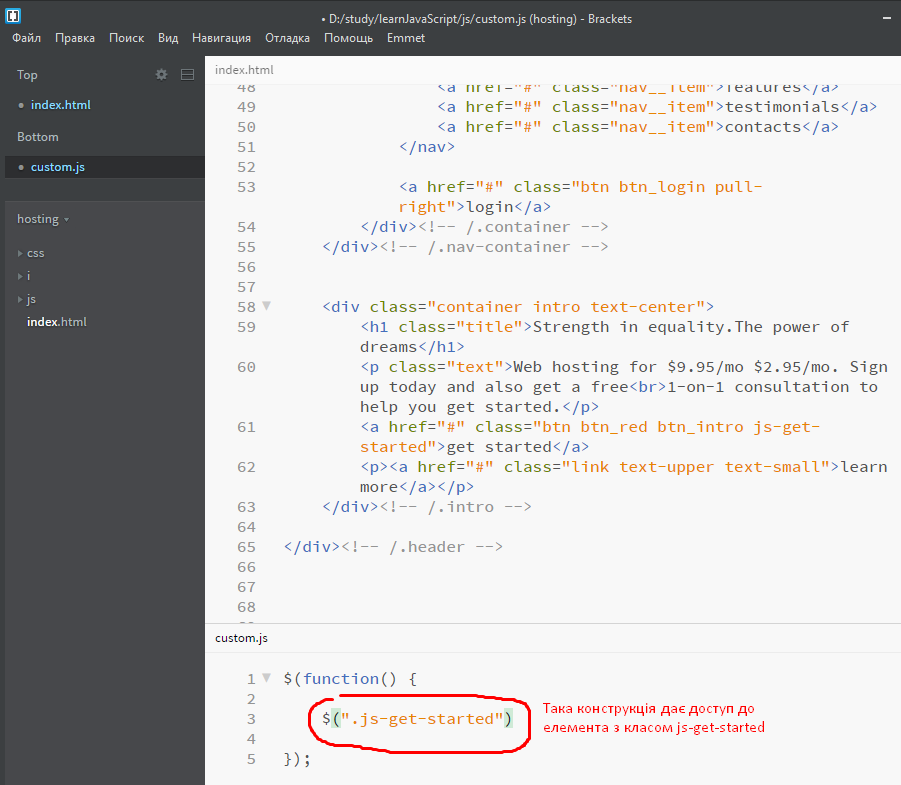
Отже, ми в рядку 61 написали перший клас js-get-started. Спробуємо зробити вибірку по цьому класу.

До речі brackets дає можливість одночасно відображати 2 файли.

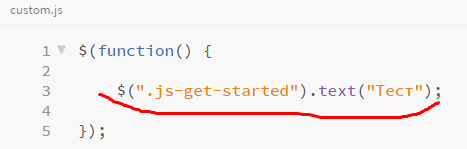


А далі в нижнє вікно перетаскуємо файл custom.js (обов’язково перевіряємо, щоб це був файл із папки hosting, не той, з яким ми проводили тренування).

І набираємо код.



І далі для цієї конструкції напишемо метод, який змінює текст на кнопці.

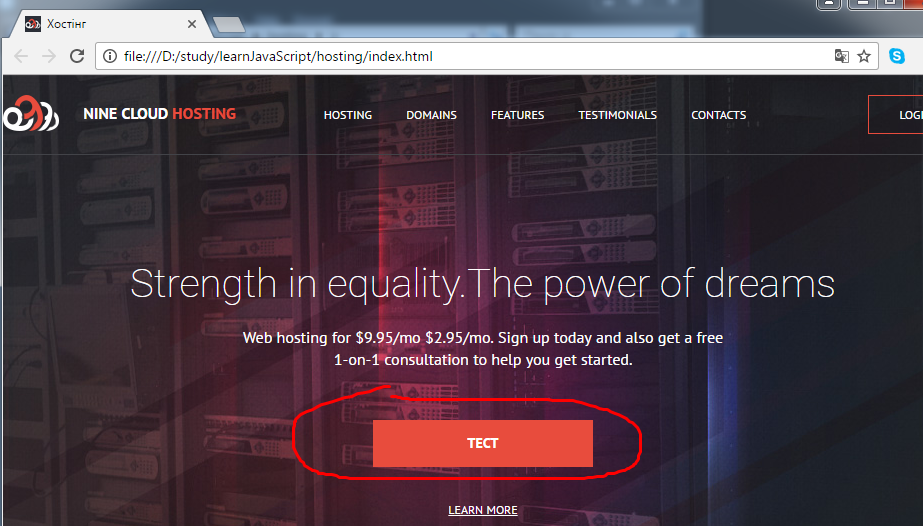


Перед тим, як переглянути контент в браузері, ми повинні перейти в кінець html-файлу і подивитись на підключення бібліотеки jQuery та файл custom.js.



Тут все нормально

Зберігаємо і обновляємо контент браузера.



Все працює. Отже, ми правильно елемент вибрали і провели з ним певну роботу.

Також ми можемо отримувати доступ до елементів через id.

Пам’ятаємо, що вибірка по id працює швидше, ніж вибірка по класу.

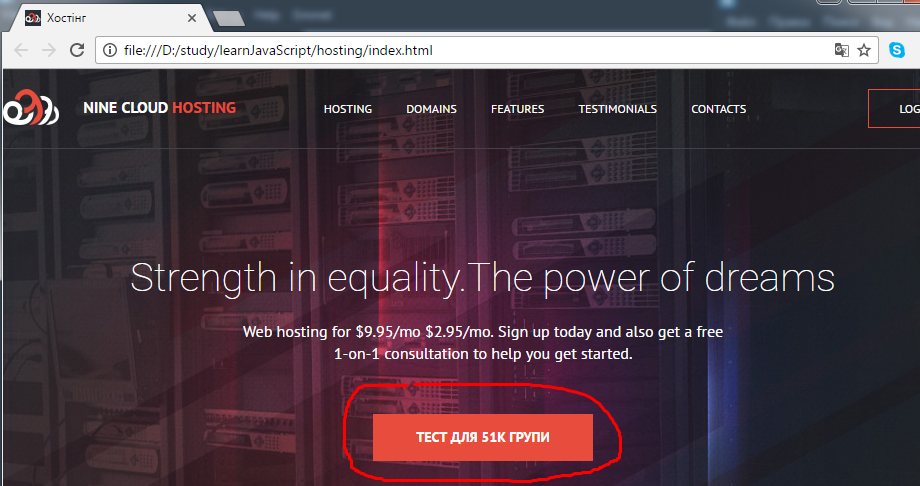
У нас кнопка з таким написом одна на сайті. Тому ми можемо змінити js-get-started з класу на id.



І перепишемо custom.js



Переглянемо в браузері

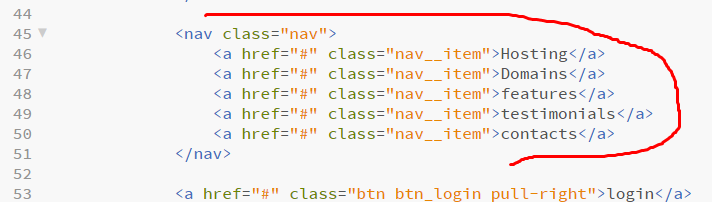


Бачимо, що все працює коректно. Вибірка пройшла успішна.

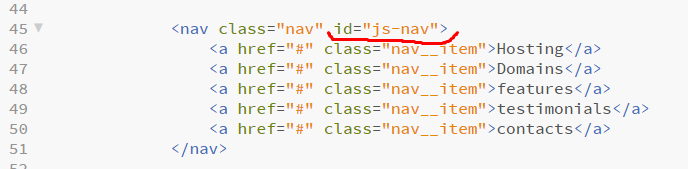
Найчастіше вибірки використовуються через id або через клас.

Але бувають також інші вибірки. Наприклад, треба вибрати дочірні елементи.

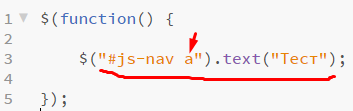
Розглянемо навігацію

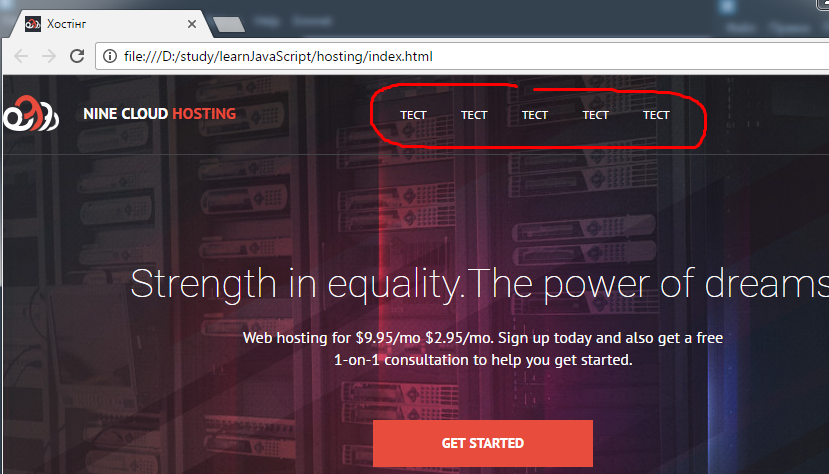


У нас на сайті навігація одна. Тому можемо присвоїти їй id для js.



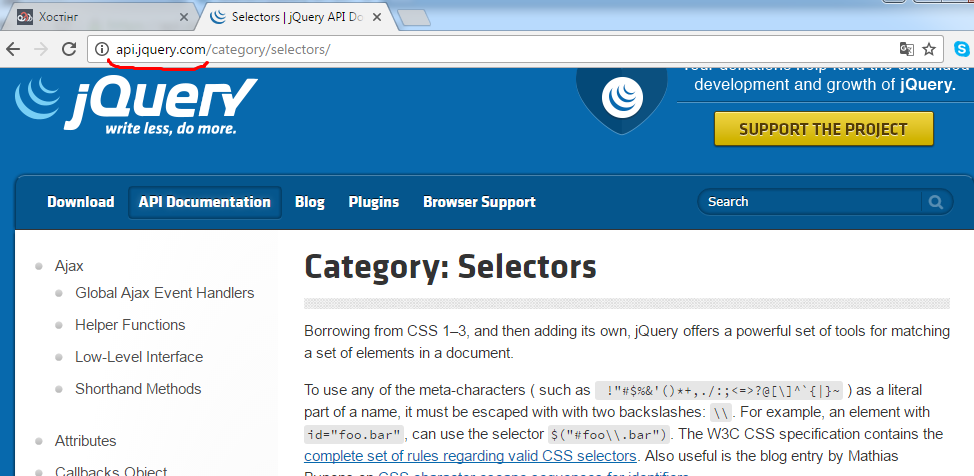
А далі спробуємо замінити всі посилання в js-nav на «Тест».



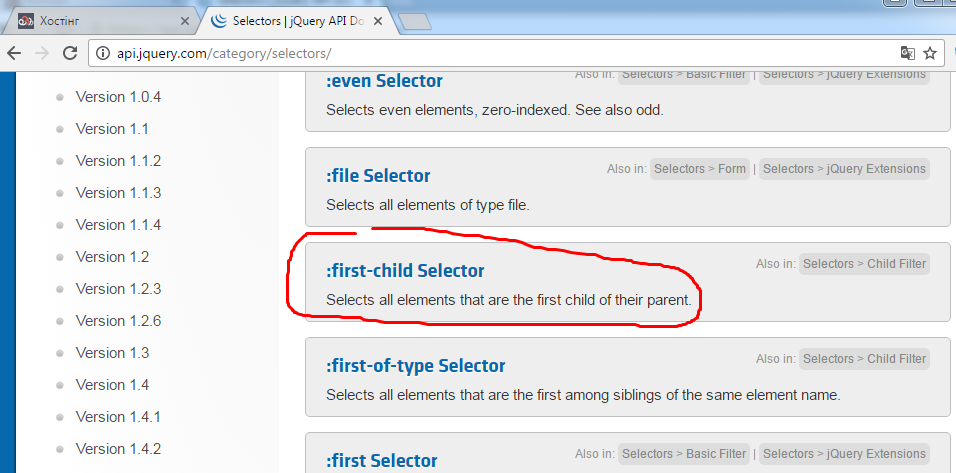


Дійсно, у нас всі елементи меню – це є посилання. І їх текст замінився тим, що ми прописали в js.

Ще раз звернемо увагу на корисність сайту



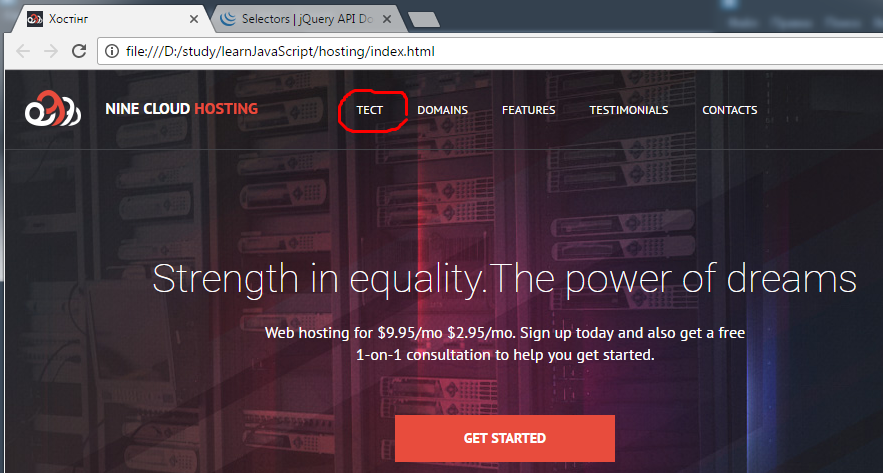
Наприклад,



Ми тут читаємо, що селектор :first-child дає доступ до першого елемента серед всіх вибраних. Спробуємо застосувати це.



Зберігаємо, переглядаємо в браузері



Дійсно, напис ТЕСТ відобразився лише для першого елемента навігації.