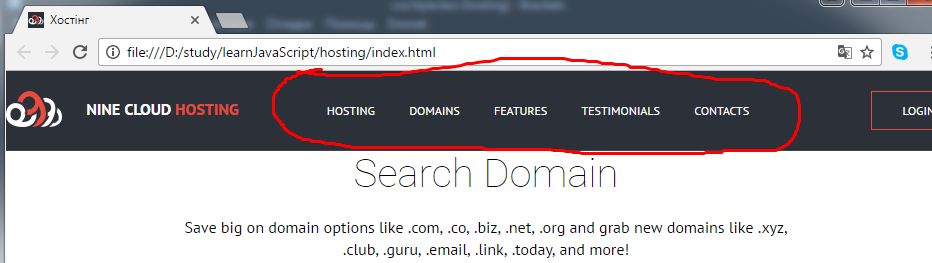
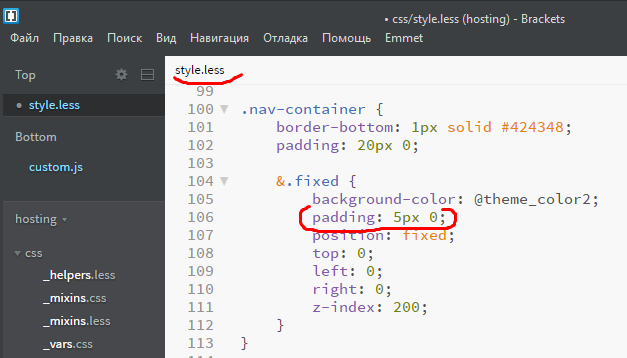
**Тема 5. Скролінг від елементів меню**

Ми маємо меню.

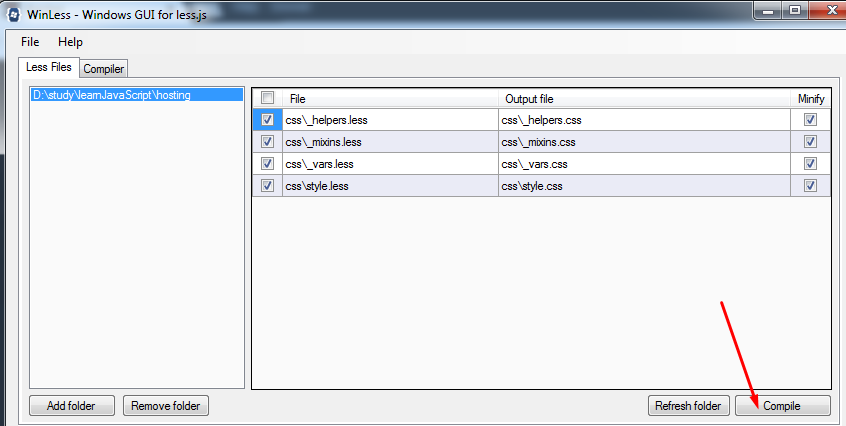


Зробимо меншими відступи (padding) зверху та знизу.

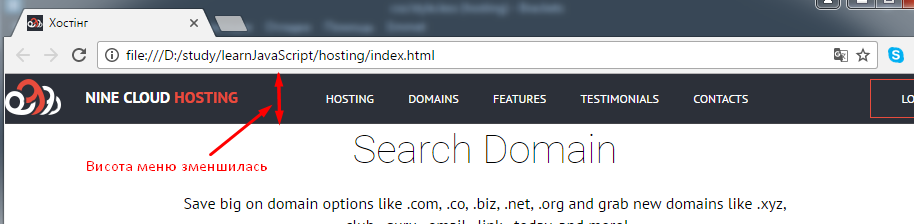
Для цього перейдемо в CSS і зробимо зміни.



Через WinLess виконуємо компіляцію less-файла в css-файл.

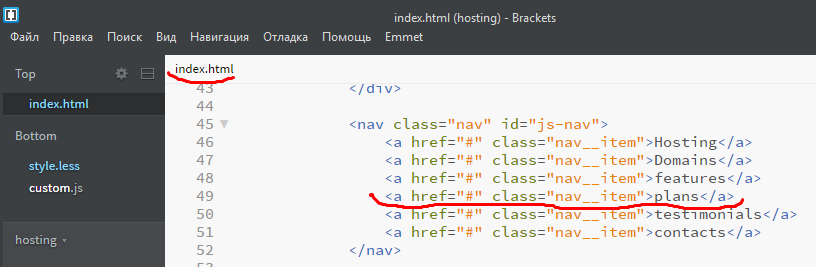


І перевіряємо через браузер.

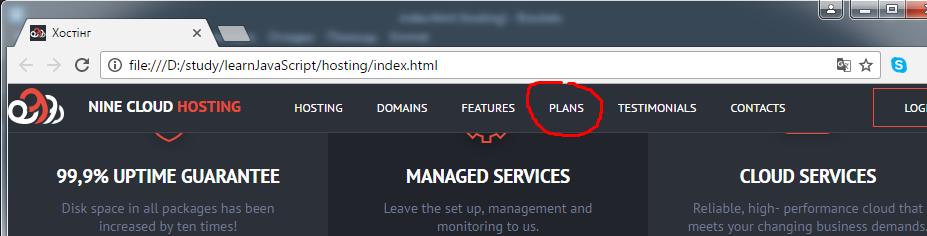


Далі зробимо наступне: при кліку на елемент меню, щоб відбувався скролінг до потрібного блоку.

До речі, в меню немає елементу Plans (де розташовані тарифні плани). Додамо цей елемент в меню.



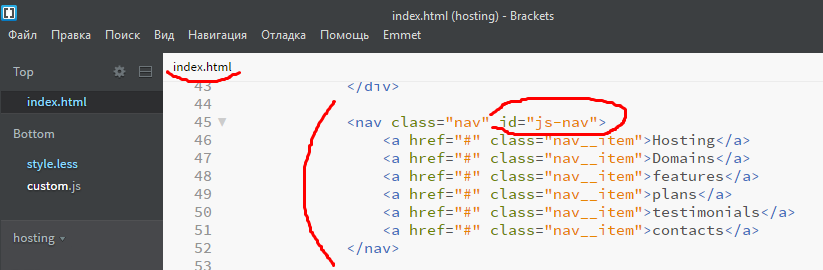
Переглядаємо в браузері



Дійсно новий пункт меню PLANS додався на сайті.

**Пропишемо для кожного елементу меню подію click.**

Для меню ми вже створили id=”js-nav”



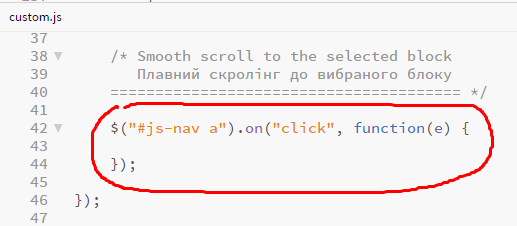
Залишилось лише отримати посилання на кожний елемент меню.

У файлі custom.js створюємо ще один блок.

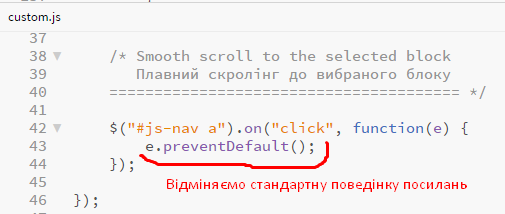


Отже прописуємо заготовку для події click по одному з посилань у списку меню

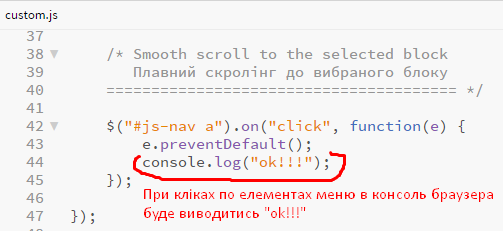
 - вибираються всі посилання, які знаходяться в об’єкті з id=”js-nav” (в даному випадку вибираються всі посилання в меню).

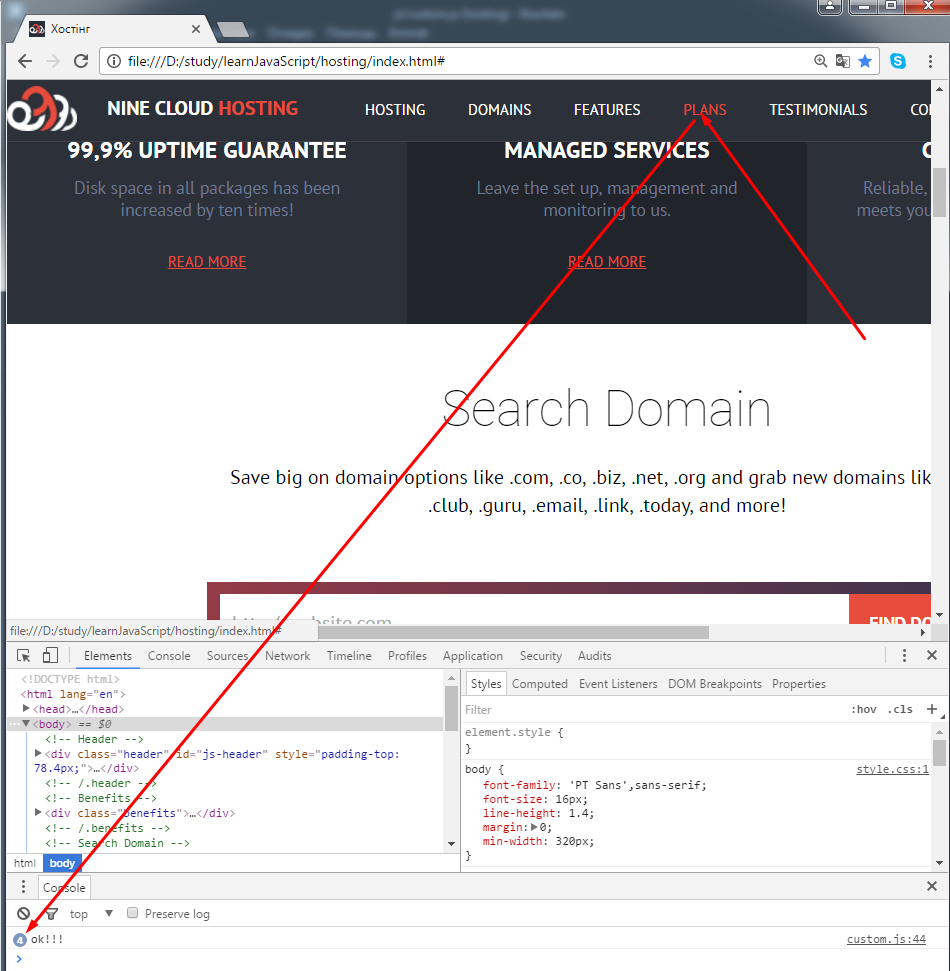


Далі відміняємо стандартну поведінку посилань.



Перевіримо, що подія спрацьовує. Для цього виведемо якесь повідомлення в консоль браузера.

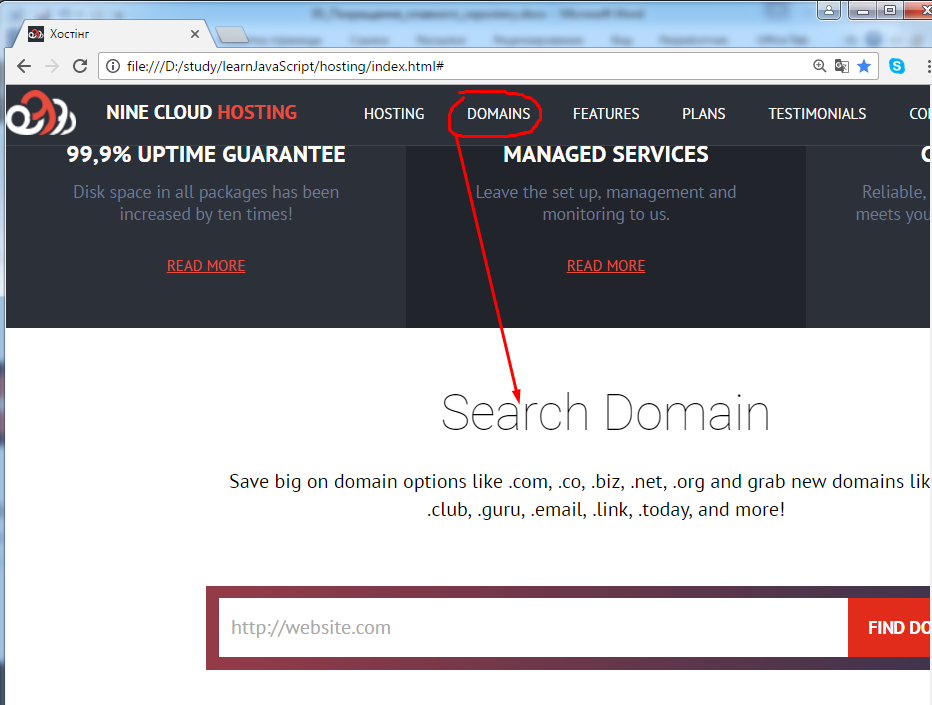




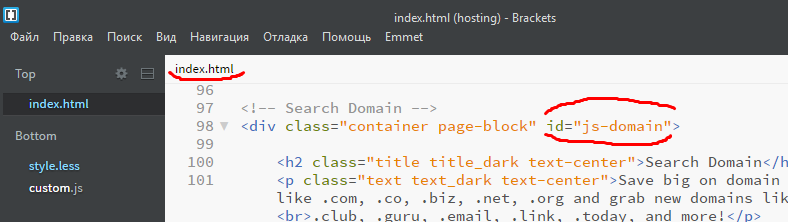
Отже, ми маємо заготовку для події click на кожне посилання на елемент меню.

Для того, щоб реалізувати перехід від елемента меню до конкретного блоку, треба кожний блок помітити через id.

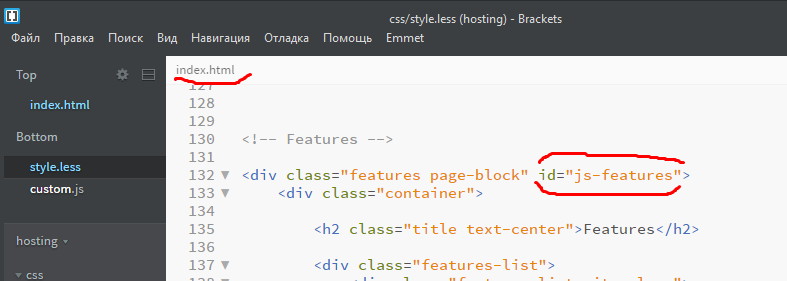
Елемент меню DOMAINS має вести на наступний блок тексту

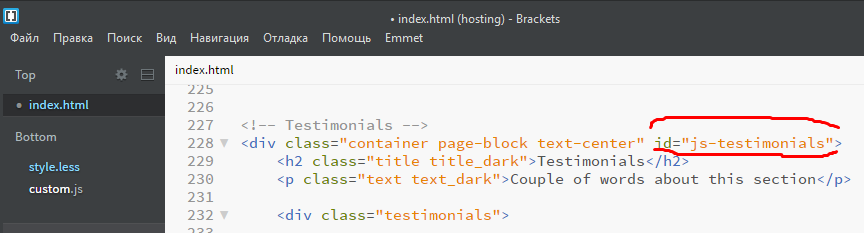


Тому присвоюємо наступний id (для блоку Search Domain)

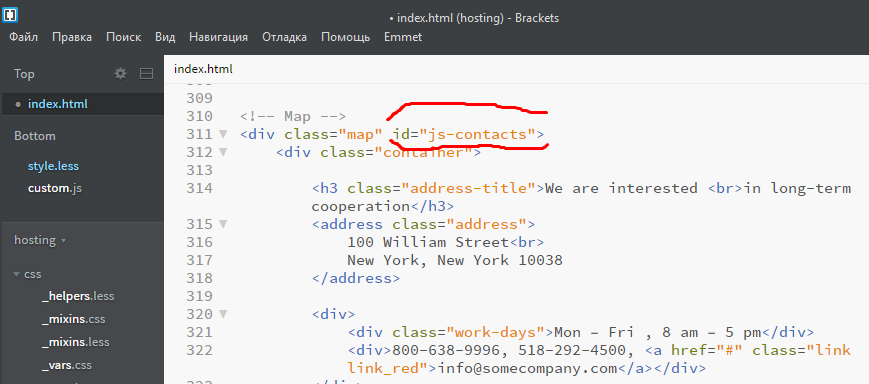


Аналогічно створюємо id для блоку Features





Контакти – прикріпимо до блоку «map»



Для того, щоб скролінг відбувався правильно до потрібного блоку, вказуємо необхідне посилання. Використовуємо для цього атрибут href

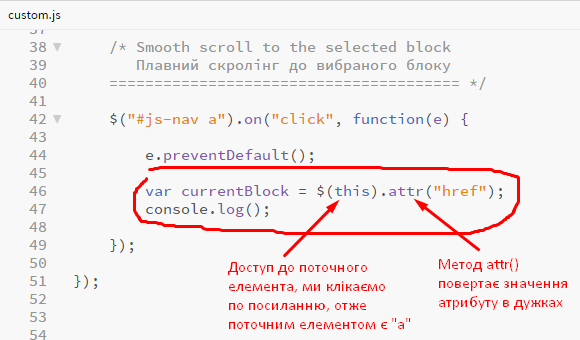


Тепер при кліку на кожне із посилань, нам треба одержати значення атрибуту href і проскролити до елемента з вибраним id.

Виведення на консоль нам більше непотрібне, ми його вилучаємо.

І створюємо змінну currentBlock, в якій збережемо значення атрибуту href.

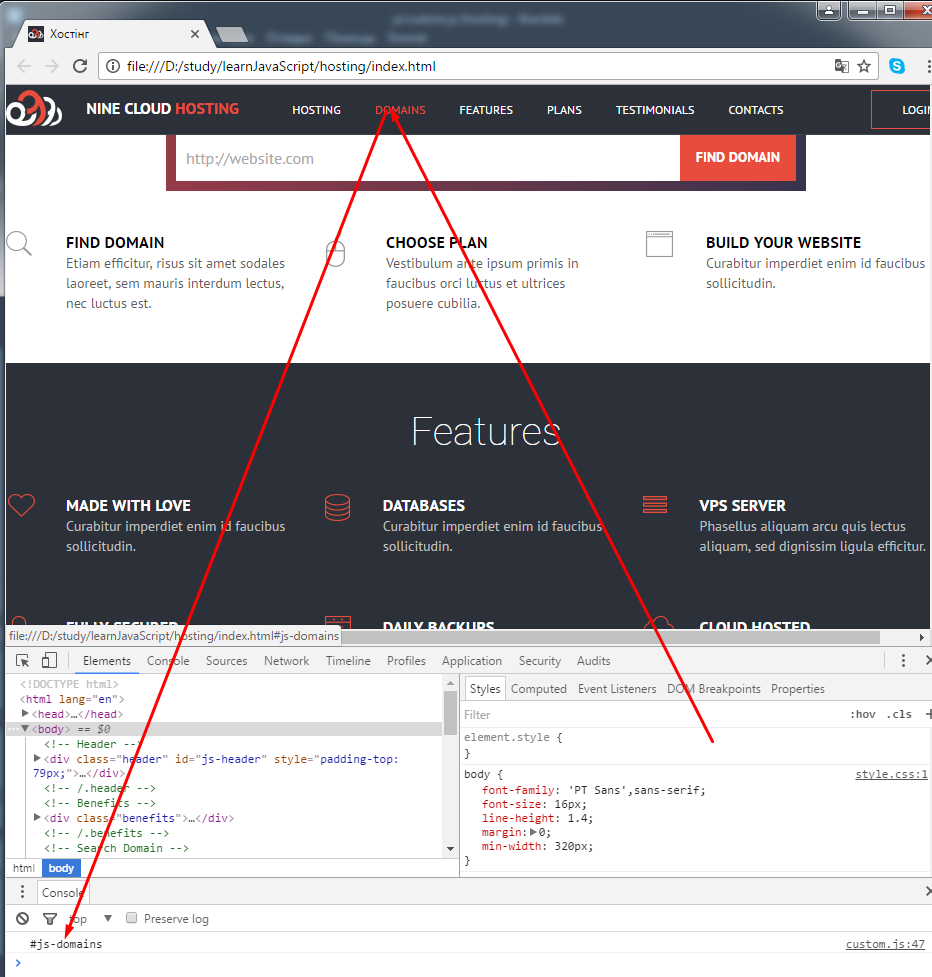
Ми клікаємо по посиланню і одержуємо доступ нього через $(this).



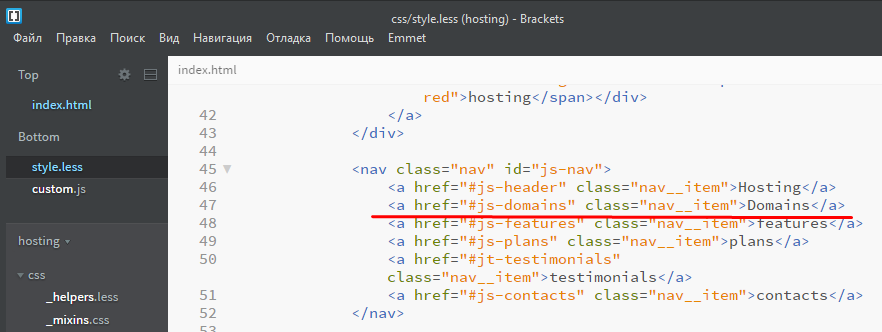
Просто console.log() – нічого не виводить в консоль. В дужках треба вказати, значення якої саме змінної ми хочемо вивести в консоль.



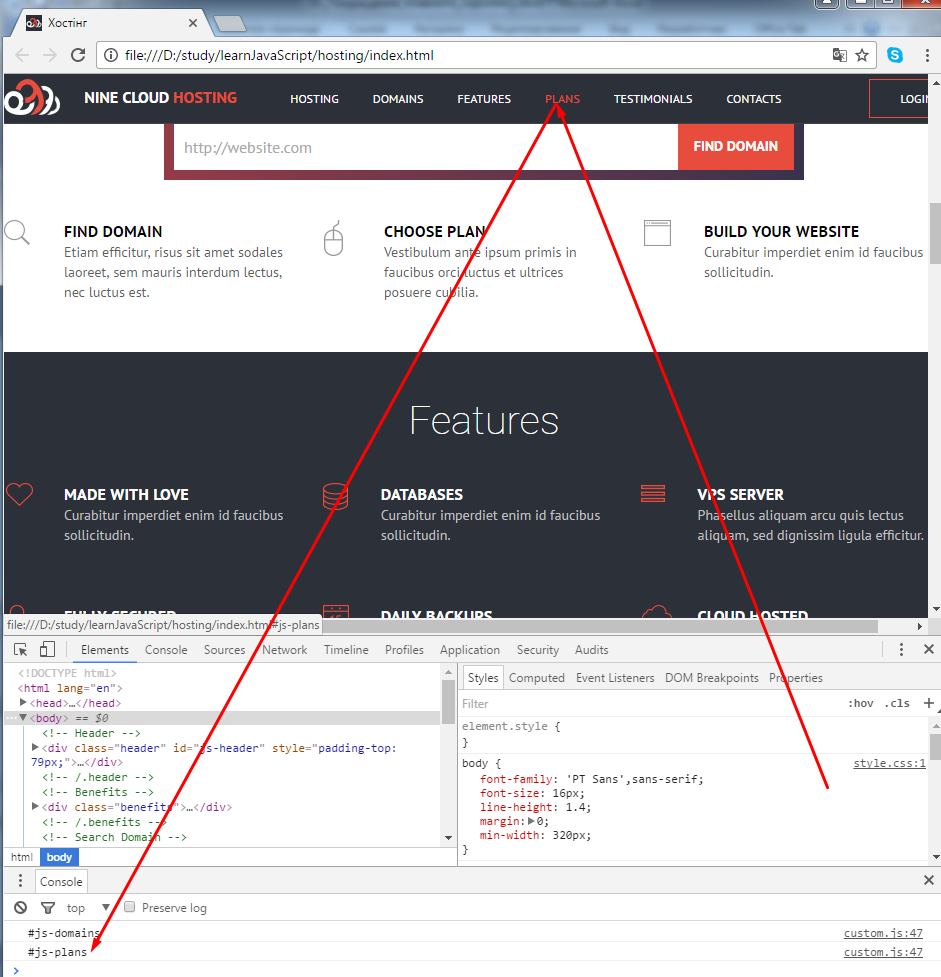
Зберігаємо, переглядаємо в браузері.



При кліку по елементу меню DOMAINS в консолі виводиться значення атрибуту href - #js-domains. Дійсно, якщо переглянути



то для елемента меню DOMAINS значення атрибуту href=”#js-domains”. Аналогічно, виконуючи кліки по інших елементах меню, одержимо в консолі значення атрибуту href.



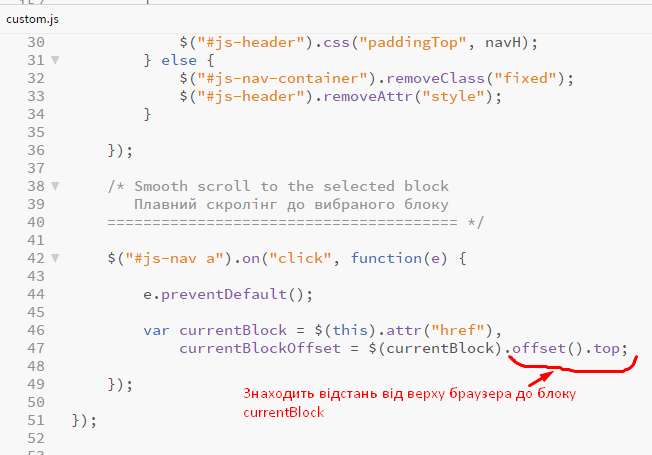
Отже, по якому елементу меню клікаємо, таке значення атрибуту href і отримуємо.

Далі через значення href ми вираховуємо висоту, на яку нам треба скролити.

Виведення в консоль вже непотрібне, вилучаємо.

І робимо вибірку елемента.

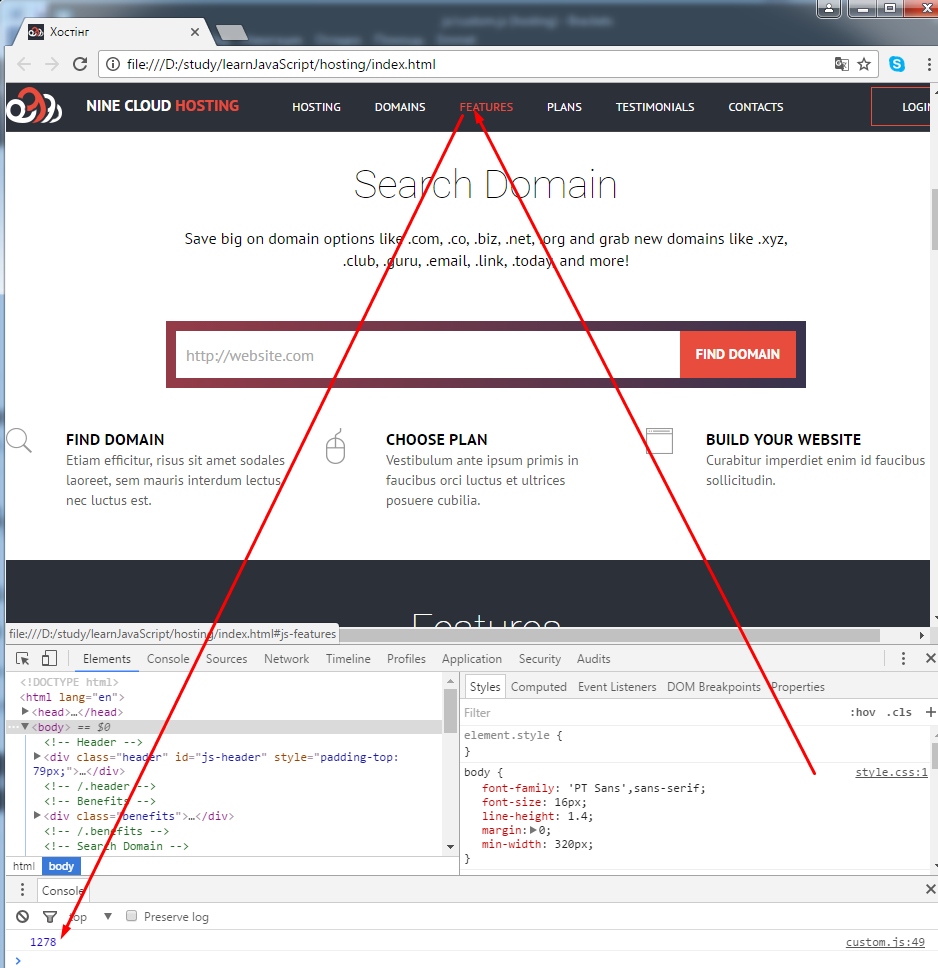
Нам потрібна відстань від верху браузера до вибраного блоку.



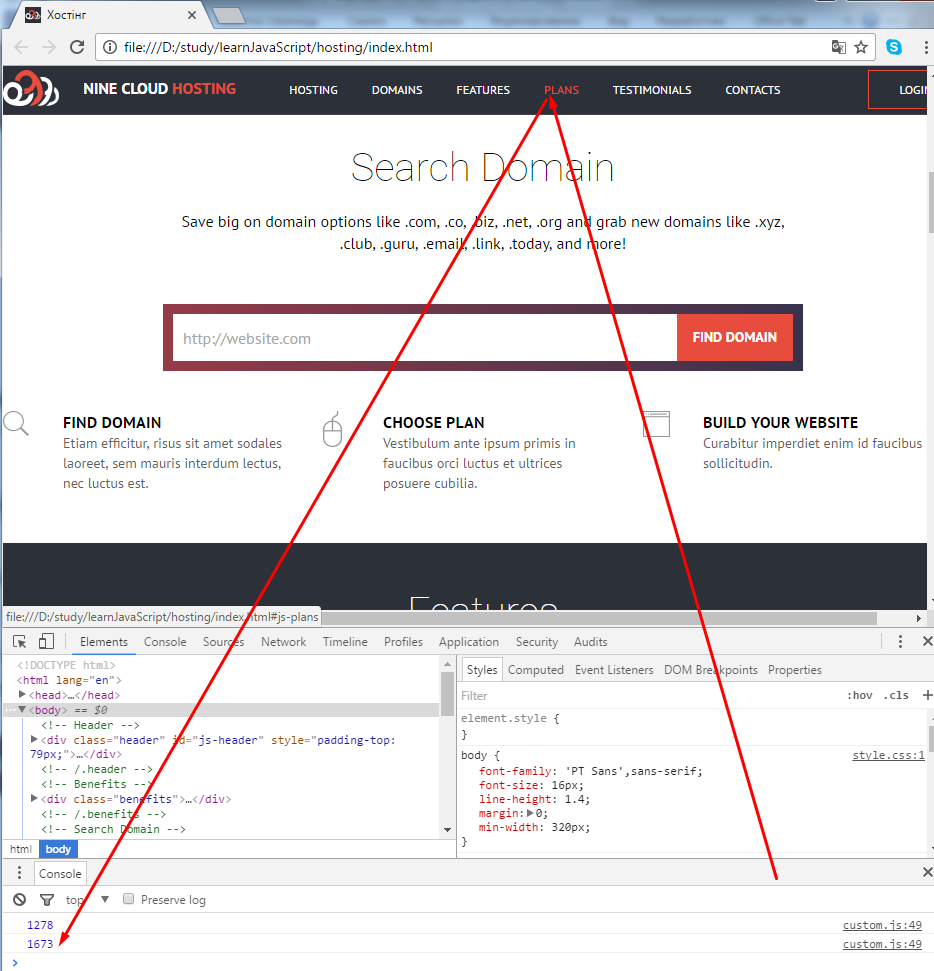
Для перевірки виведемо одержане значення на консоль.



Переглянемо в браузері: при кліку по кожному елементу меню, в консолі ми повинні отримувати пікселі відстані до відповідного блоку.

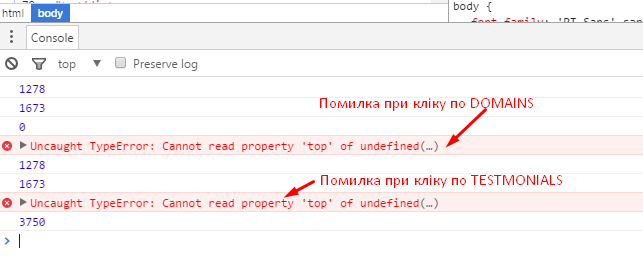


Як бачимо, при кліку по елементу меню FEATURES консоль нам говорить, що відстань до даного блоку становить 1278 пікселів. Можна переглянути інші відстані до інших блоків (клікаючи по елементах меню).

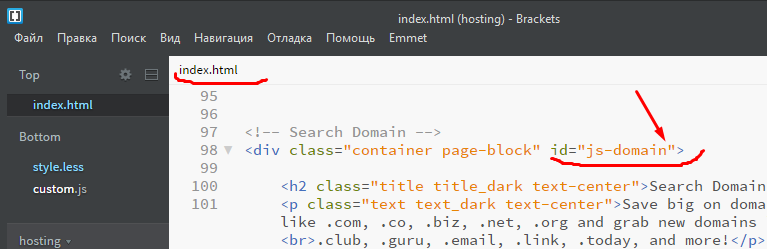


Клікаючи по елементу меню Plans, одержуємо відстань 1673 пікселі (ми вказали **.top** – тому відлік починається від верху браузера.

Можна поклікати по всіх елементах меню, щоб перевірити, чи не видає помилок якийсь із елементів.

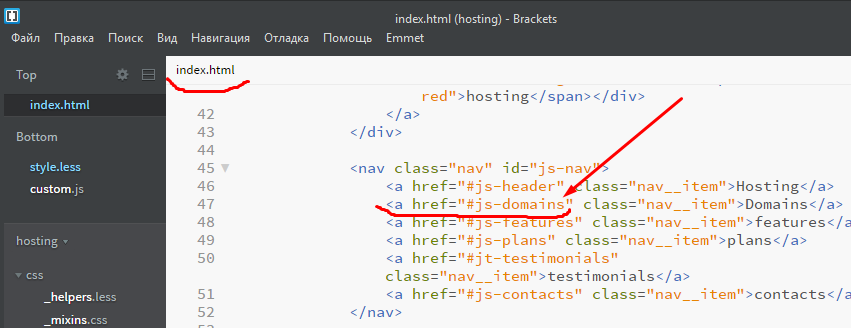


Перевіримо

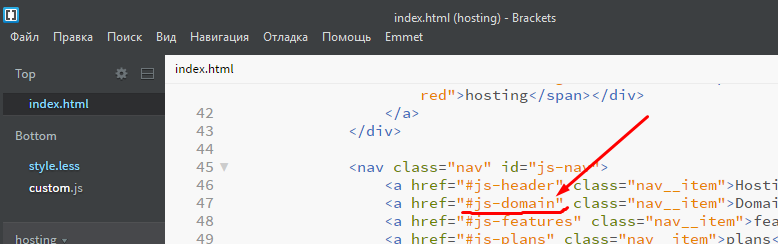


Бачимо, що у файлі index.html для блоку ми вказали id рівним **js-domain** (без літери **s** в кінці).

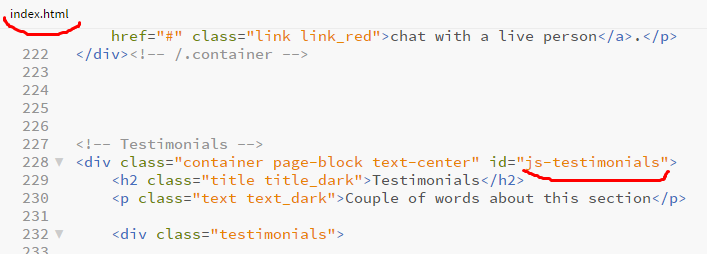
А в меню прописали

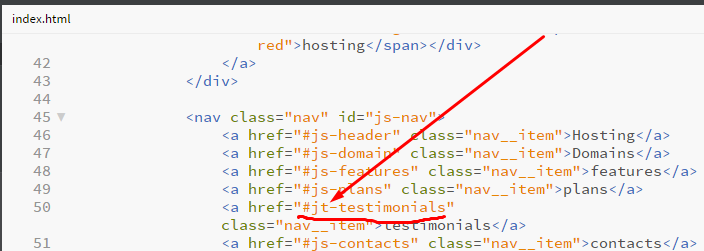


перехід до блоку з id=”js-domains” (з літерою **s** в кінці). Тому такий елемент не знаходиться і видається помилка, виправляємо – вилучаючи літеру s в кінці фрази.

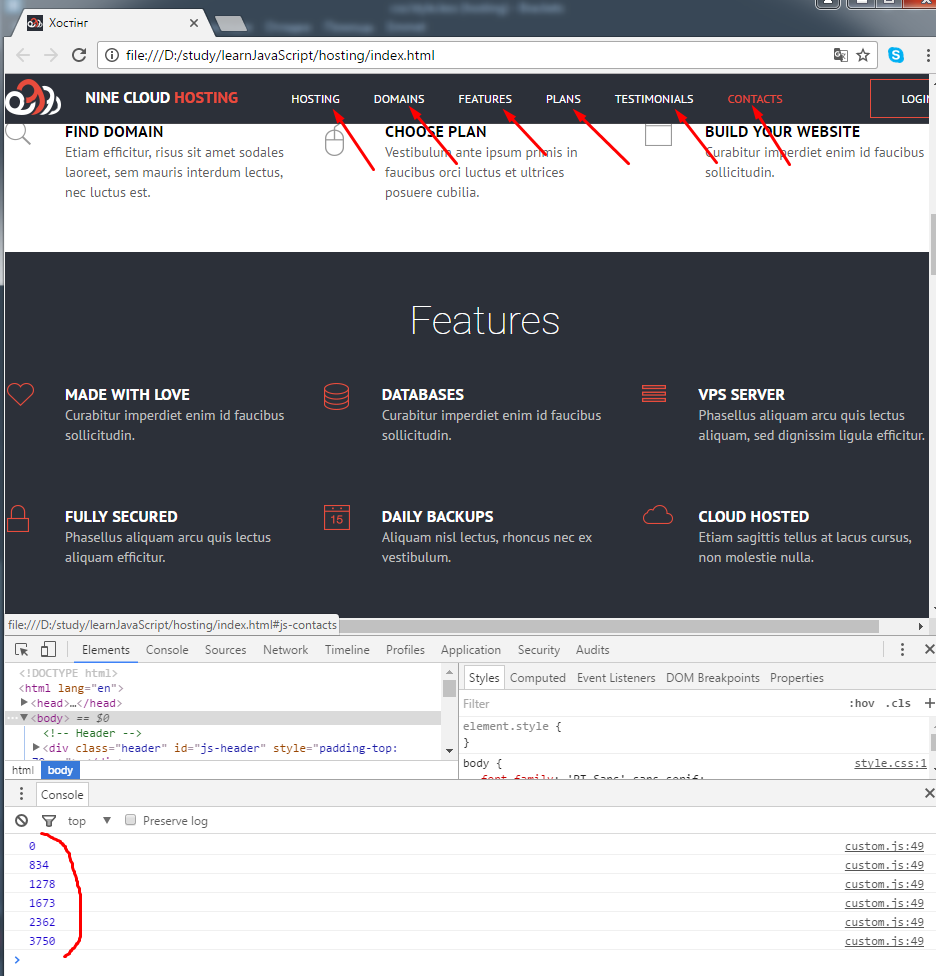


Аналогічно переглянемо, чому виникла помилка з TESTMONIALS.





Тут також невідповідність (помилково замість js написано jt). Також виправляємо.

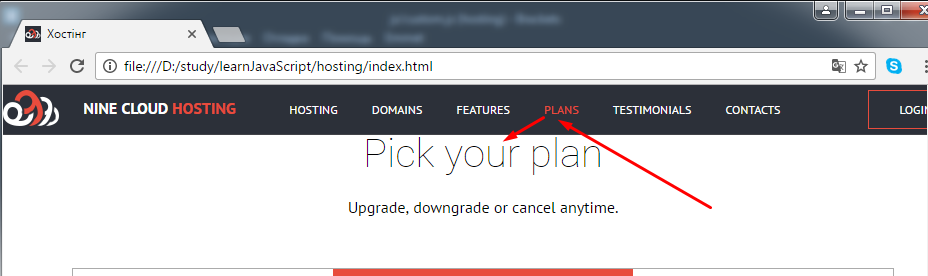


Після виправлень клікаємо по всіх шести елементах меню – помилок немає, для кожного елемента меню виводиться відстань в пікселях до відповідного блоку.



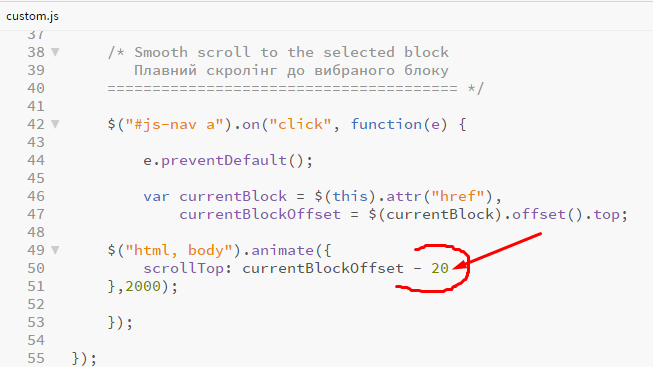
Вказуємо, що потрібно скролити до значення змінної currentBlockOffset.

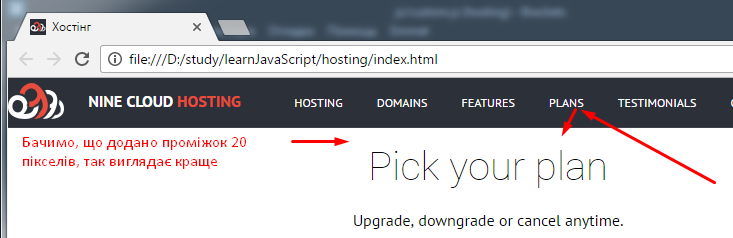
Перевіряємо в браузері, клікаючи по відповідних елементах меню.



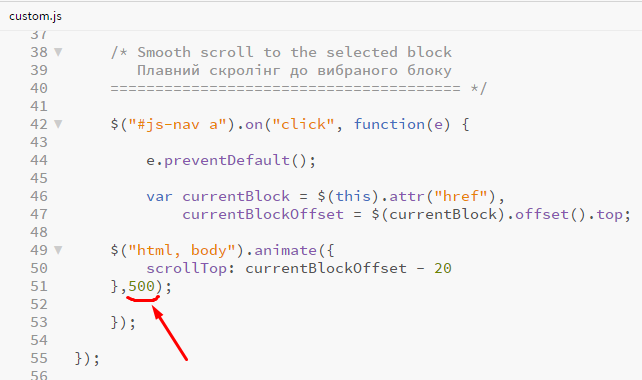
Все працює. Не дуже швидкий скролінг, адже ми вказали час для скролінгу 2000 мілісекунд (2 секунди). Можна вказати інший час, це все – на власний розсуд.

Що не зовсім добре – бачимо, що блок «Pick you plan» аж сильно близько наблизився до меню. Треба трохи зменшити відстань скролінгу, наприклад, на 20 пікселів.

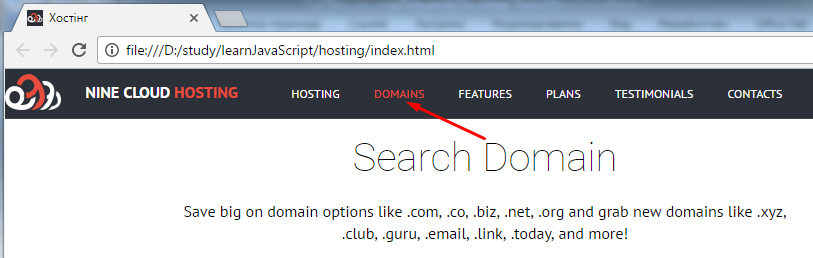




Поставимо час скролінгу не 2 секунди, а пів секунди (значення 500).

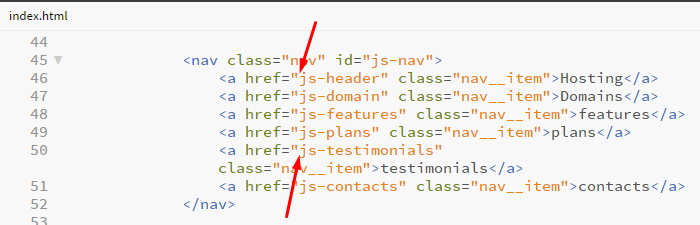


Тепер скролінг виконується швидше.



Отже, все працює.

Які ще можуть бути нюанси. Якби значення href були без решітки, наприклад,

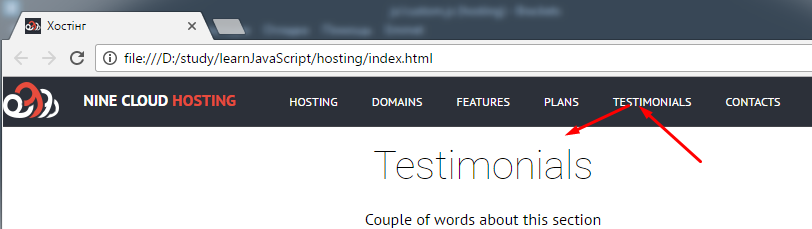


(ми кругом перед js решітку вилучили),

тоді прийшлось би дописувати решітку наступним чином

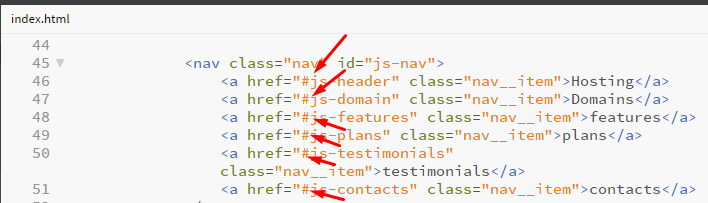


Перевіряємо такий варіант.



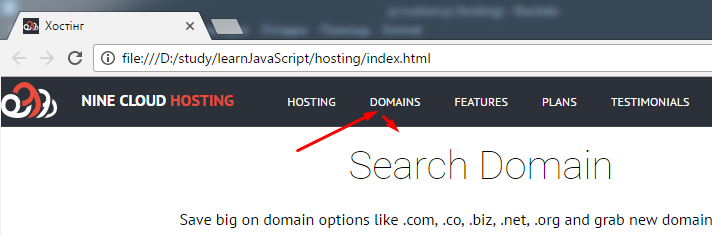
Так працює все добре також.

Але повертаємо назад все, як було раніше (щоб нічого не склеювати).





І ще раз перевіряємо, що все працює.



Виправили все добре, все робить.