



web дизайн

Урок 8

Робота з плагінами

Зміст

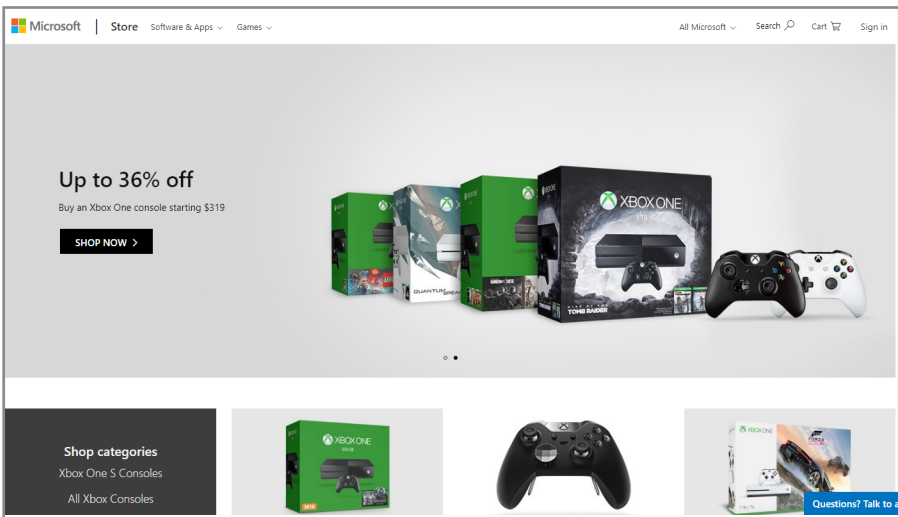
Робота з плагінами.....	3
Використання каруселі на сайті	3
Створення каруселі.....	6

Матеріали уроку прикріплені до даного PDF-файлу. Для доступу до них, файл необхідно відкрити в програмі Adobe Acrobat Reader.

Робота з плагінами

Використання каруселі на сайті

Карусель — це цікавий спосіб розміщення інформації на сайті. Карусель — один із найпопулярніших компонентів у дизайні. Вона реалізується за допомогою CSS, JavaScript. Її часто використовують, щоб привернути увагу відвідувачів. Приклад каруселі на сайті Microsoft (мал. 1).



Малюнок 1

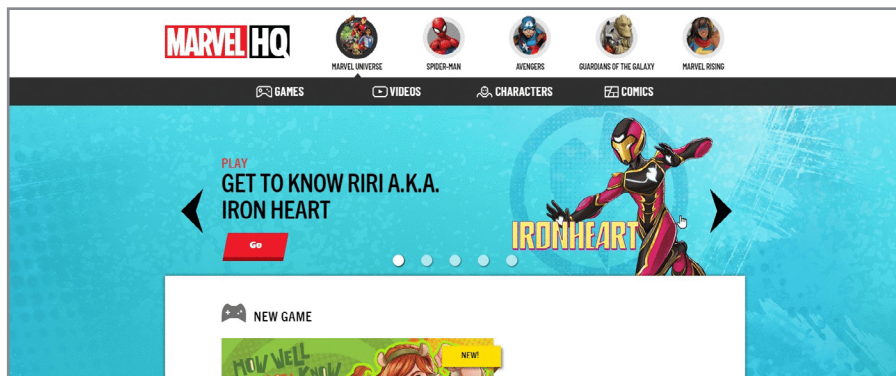
На ресурсі MarvelHQ карусель містить сторінки різних персонажів (мал. 2).

Є безліч типів каруселей, однак у всіх можна виділити загальні риси:

1. **Розташування.** Найчастіше карусель розташована у верхній частині екрана, під хедером і навігаційним

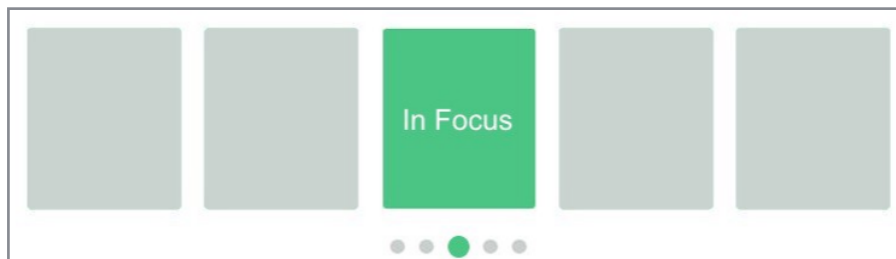
меню. А сам слайдер займає більшу частину простору на сторінці.

- Зміст.** Карусель найчастіше складається з зображення, тексту і, за необхідності, кнопки.



Малюнок 2

- Кнопки та індикатори.** Можна побудувати навігацію по каруселі. Можна надати користувачам можливість гортати слайди самостійно, додавши кнопки прокрутки (вправо/вліво). Також можна додати до каруселі індикатори стану, щоб завжди можна було перегорнути всі слайди до кінця (мал. 3).



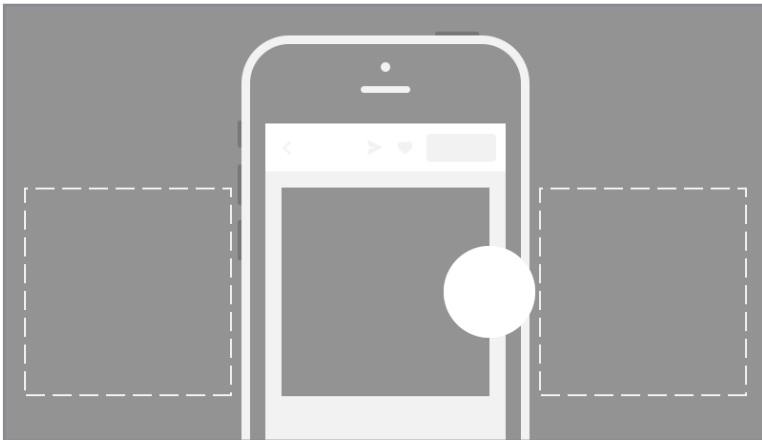
Малюнок 3

4. **Адаптивність.** Надзвичайно важливо створювати адаптивні каруселі, які б коректно відображалися на пристроях з різним дозволом екрана (мал. 4).



Малюнок 4

5. **Автопрокрутка.** Встановлюючи автопрокрутку, слайди не мають змінюватися занадто швидко або повільно (мал. 5).



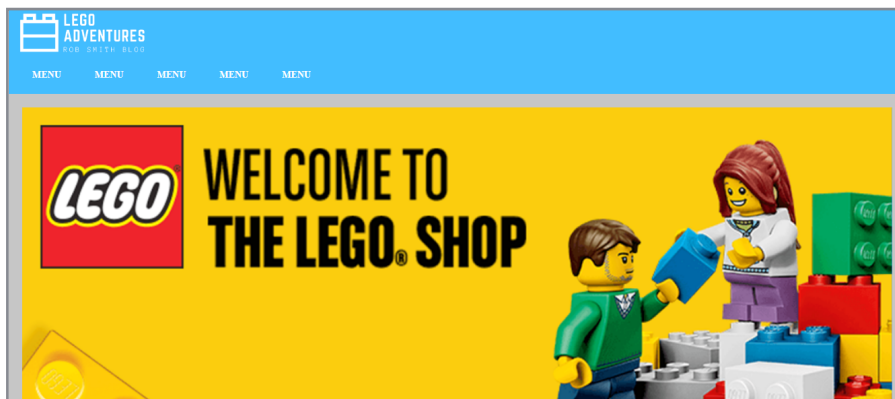
Малюнок 5

Карусель ефективна в тому випадку, якщо містить корисний контент. Тобто послідовно прокручуючи слайди, користувач отримує нову для себе інформацію. Однак карусель має і мінуси. Наприклад, користувачі часто її ігнорують. До того ж, коли користувач бачить тільки зображення, він не отримує цілісного уявлення про продукт.

Виділимо основні плюси використання каруселі. На головній сторінці слайдер займає центральне місце. Це перше, що побачить відвідувач. Також на одному слайді можна розмістити цілий блок інформації, яка привертає увагу користувачів.

Якість каруселі визначається її текстовим наповненням і зображеннями. Слайди каруселі треба створювати в загальній стилістиці сайту, дотримуючись загальної колірної схеми. Послідовність слайдів потрібно вибудовувати відповідно до їхньої важливості. Спочатку треба подавати важливу інформацію.

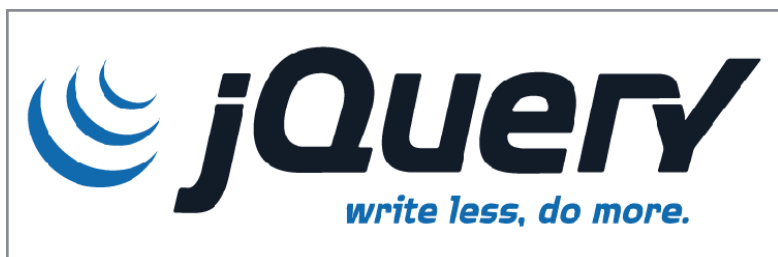
Створення каруселі



Малюнок 6

Помістимо карусель з банерами про LEGO на головній сторінці (мал. 6).

Для цього нам знадобиться JavaScript. Не лякайтеся, що не знаєте цю мову, адже зробити це досить просто. Для цього нам знадобиться бібліотека jQuery, яка значно спрощує роботу з JavaScript (мал. 7).



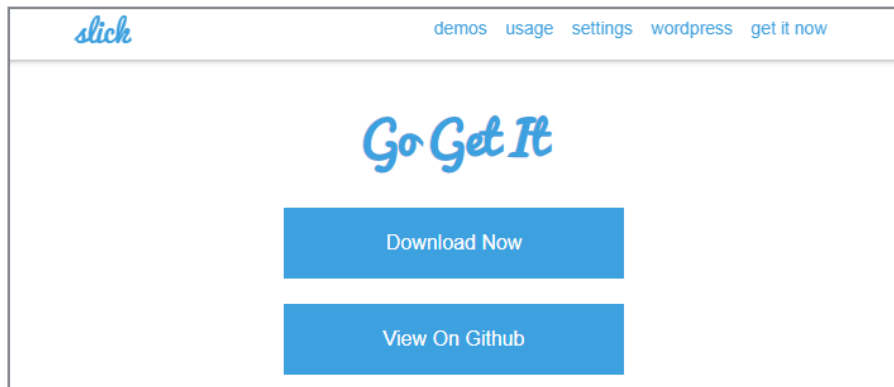
Малюнок 7

Підключимо до неї плагін Slick для створення каруселей (мал. 8).




Малюнок 8

Почнемо з того, що розмістимо файли Slick у папці з сайтом. У додаткових матеріалах ви знайдете архів. Також Slick можна завантажити на офіційному сайті (мал. 9): <https://kenwheeler.github.io/slick/#go-get-it>.



Малюнок 9

В архіві знайдіть папку *slick* і перетягніть її в папку з вашим сайтом. У папці з сайтом створіть ще одну папку — *js*. У ній будуть зберігатися ваші скрипти. Створіть у ній порожній текстовий файл *script.js* (мал. 10).

Name	Date modified	Type	Size
 script.js	5/7/2020 3:59 PM	JavaScript File	1 KB

Малюнок 10

Тепер відкрийте в редакторі сторінку, на якій хочете розмістити карусель. Наприклад, головну сторінку *index.html*. Спочатку підключіть до файлу jQuery. Для цього потрібно додати `<script>` з параметрами в кінець `<body>`.

Офіційна документація *slick* рекомендує підключати jQuery ось так:

```
<script type="text/javascript"
  src="https://code.jquery.com/
  jquery-1.11.0.min.js"></script>

<script type="text/javascript"
  src="https://code.jquery.com/
  jquery-migrate-1.2.1.min.js"></script>
```

Потім треба підключити скрипт зі Slick, а також створений нами файл *script.js*:

```
<script type="text/javascript"
  src="slick/slick.min.js"></script>

<script type="text/javascript"
  src="js/script.js"></script>
```

Так після закриваючого тегу `</footer>` і до закриваючого тегу `</body>` у нас має вийти такий запис (мал. 11):



```
</div>
</footer>

<script type="text/javascript" src="https://code.jquery.com/jquery-1.11.0.min.js"></script>
<script type="text/javascript" src="https://code.jquery.com/jquery-migrate-1.2.1.min.js"></script>
<script type="text/javascript" src="slick/slick.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>

</body>
</html>
```

Малюнок 11

Також потрібно підключити таблицю стилів Slick. Для цього просто додайте в `<head>` два рядки (мал. 12):

```
<link rel="stylesheet" type="text/css"
      href="slick/slick.css"/>
```

```
<link rel="stylesheet" type="text/css"
      href="slick/slick-theme.css"/>
```

```
<!DOCTYPE HTML>
<html>
  <head>
    <link rel="icon" href="img/favicon.ico" type="image/x-icon">
    <link rel="stylesheet" href="style.css">
    <link rel="stylesheet" type="text/css" href="slick/slick.css"/>
    <link rel="stylesheet" type="text/css" href="slick/slick-theme.css"/>
  </head>
  <title>Lego Adventures</title>
```

Малюнок 12

Тепер у місці, де ви хочете помістити карусель, створіть `<div>` з тегом `carousele`. У нього помістіть `<div>` з тегом `carousele-element`. Таких `carousele-element` ви можете створити стільки, скільки хочете. Наприклад, два.

```
<div class="carousele">
  <div class="carousele-element">
  </div>

  <div class="carousele-element">
  </div>
</div>
```

У самому `carousele-element` має бути контент, який буде відображатися в каруселі. Це може бути текст, відео, але найчастіше туди поміщають зображення. Так і зробимо:

```
<center>
  
</div>
</center>
```

Тепер додамо стилі для нашої каруселі в *style.css*.

```
.carousele{
  margin: 10px;
}
.carousele div img{
  width: 100%
}
```

Тепер потрібно налаштувати скрипт з нашою каруселлю *slick*. Відкрийте в редакторі *script.js*. Додайте туди наступний запис:

```
$(document).ready(function(){
  $(''.carousele').slick(
    {

  });
});
```

У *.slick({})*; ви можете додавати налаштування каруселі. Повний список налаштувань доступний на офіційному сайті *slick*: <https://kenwheeler.github.io/slick/#settings>

Ми ж використаємо тільки дві з них: відключимо стрілки для перемикання між банерами і включимо автоматичну зміну банерів.

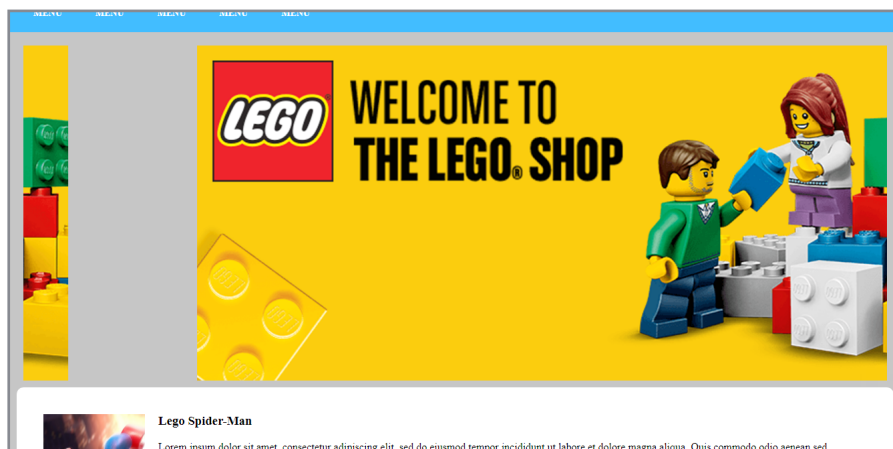
```
arrows: false,  
autoplay: true
```

Наш скрипт виглядатиме так (мал. 13):

```
$(document).ready(function() {  
    $('.carousele').slick({  
        {  
            arrows: false,  
            autoplay: true  
        }  
    });  
});
```

Малюнок 13

Тепер банери будуть перемикатися автоматично (мал. 14).



Малюнок 14

STEP IT Academy, www.itstep.org

Усі права на захищені авторським правом фото, аудіо та відеотвори, фрагменти яких використані в матеріалі, належать їхнім законним власникам. Фрагменти творів використовуються з ілюстративною метою в обсязі, виправданому поставленим завданням, в межах навчального процесу і в навчальних цілях. Відповідно до ст. 21 і 23 Закону України «Про авторське право й суміжні права». Обсяг і спосіб цитованих творів відповідає прийнятим нормам, не завдає шкоди нормальному використанню об'єктів авторського права і не обмежує законні інтереси автора та правовласників. Цитовані фрагменти творів на момент використання не можуть бути замінені альтернативними, що незахищені авторським правом аналогами, і як такі відповідають критеріям сумлінного і чесного використання.

Усі права захищені. Повне або часткове використання матеріалів заборонено. Узгодження використання творів або їхніх фрагментів проводиться з авторами і правовласниками. Узгоджене використання матеріалів можливе лише за умов згадування джерела.

Відповідальність за несанкціоноване копіювання і комерційне використання матеріалів визначається чинним законодавством України.