**Адаптивність**

<https://html5book.ru/css3-mediazaprosy/>

Типи адаптивності:

1. Фіксована

**Фіксована адаптивність:**

При зменшенні розмірів екрану, частина контенту виходить за рамки екрану і ми її не бачимо

Для виправлення даного непорозуміння використовуємо мета-тег:

1. <meta **name = ‘viewport’content=”width: contentContainerWidth\_without pixels**”>
2. В body I container: задаємо **min-width: розмір контейнера**

*!якщо є бажання зробити відступи по бокам:*

1. Min-width: **для container задаємо більшою із урахуванням padding**
2. Значені min-width **– задаємо для meta teg і body**

Поганий варіант для адаптації і його варто використовуваит лише для дуже старих сайтів, де не хочеться робити адаптивність

**Відзивчива верстка:**

Контент весь реагує на зміни ширини екрану і підлаштовуєть для коректного відображення

**Для задання відзивчивої верстки:**

1. <meta **name = ‘viewport’content=”width: device-width**”>
2. Контейнеру задаємо розмір з допомогою **max-width**
3. **Розміри** елементів сторінки прописуватит **в відсотках**
4. Якщо присутня колоночна система, то задавати велечини в **flex 0 1 value%**

**Адаптивна верстка:**

Адаптивність з використанням медіа файлів для прелаштування контенту під розміри

**Відивчиво-адаптивна верстка:**

Поєднання двох способів верстки, підлаштування контенту під розміри і зміна вигляду сторінки завдяки медіа запитам

**Адаптивна верстка**

**Робота з media запитами не впливаючими на відображення при зміні розмірів екрана:**

@media all {} **– для будь-яких розмірів**

@media print {} – **для печатного варіанту**

@media screan {} – **для кольорових моніторів**

@media speech {} – **для різних синтизаторів мови**

**Робота з media запитами на орієнтацію екрану**

@media ( **orientation:** portait ) {} – стилі для портретного режиму

@media ( **orientation:** landscape ) {} – стилі для альбомного режиму (горизонтального)

**Робота з media запитами на розширення екрану**

@media ( (max/min)**resolution: 300dpi** ) {} – стилі для екранів з даним розширенням (к-сть точок на дюйм)

@media ( **monochrome** ) {} – стилі для чорно-білтх екранів

**Робота з класичними breakpoint ( ключовими точками ):**

@media ((max/min)width/height: ) – ключова точки які діють в момент використання девайсом із таким розширенням

**Поєднання різних властивостей через** **and/or:**

**And – строга властивість**

**Or – не строга властивість**

@media screen and (max-width: 767px){} – на кольорових екранах і при даному діапазоні шириниі

**Верстка Mobile First:**

Верстка від мобільної версії до destop

**Загальні фішки адаптивності:**

1. **Використання мета тегу для хорошого відображення на мобільних пристроях**:

<meta name=”viewport” content=”width: device-width, initial-scale=1.0, maximum-scale= 1.0, user-scalable=0 ”> - але можу унеможливти на деяких девайсах навмисне масштабування сторінки

1. Використання **меню-бургер** для мобільних версій
2. Можемо приховати деякі елементи сайту для мобілок, якщо вони не дуже важливі(**захвувати неважливі блоки в спойлери**)
3. **Зменшення розмірів шрифта** при зменшенні екрану
4. **Збільшуємо розміри навігаційних елементів** при зменшенні екрану
5. **Картинкам задаємо max-width:** 100% - для максимального підлаштування під її батька при зміні його розміру, сама картинка адаптується і по висоті і по ширині
6. **Вкладати медіа запити в стильовий контейнер елемента** для якого ми описуємо адаптивність
7. Уменьшение количества колонок (столбцов) и постепенная отмена обтекания для мобильных устройств
8. Использование свойства max-width вместо width при задании ширины блока-контейнера
9. Уменьшение полей и отступов на мобильных устройствах (например, нижних отступов между заголовком и текстом, левого отступа для списков и т.п.).
10. Уменьшение размеров шрифтов для мобильных устройств.
11. Превращение линейных навигационных меню в раскрывающиеся
12. Скрытие второстепенного содержимого на мобильных устройствах с помощью display: none.
13. Подключение фоновых изображений уменьшенных размеров.

**Адаптація картинок (відзивчива верстка):**

**Контентних:**

Використання max-width: 100% для контентних картинок

Для колоночних картинок

1. Object: cover – застосовується для фотографій різного розміру в колонках
   1. Fill – розтягує картинку, щоб вона використовуючи власні ширину і висоту покрила всього батька
   2. Contain – змушує повністю поміститись картинку в середину батька
   3. None – фото виглядає, як background
   4. Cover – покриває батька масштабуючи себе і зберігаючи максимум корисної інфрмаці на картинці, намагаючись її відобразити в батькові
2. Object – position – аналогічно background-position

**Приклад адаптування колоночних картинок різного розміру для пропорційного зменшення:**

**.card\_\_image{**

**Overflow: hidden;**

**Position: relative;**

**Padding: 0 0 58% 0;**

**}**

**.card\_\_image img{**

**Width: 100%;**

**Height: 100%;**

**Object-fit: cover;**

**Position: absolute;**

**Top: 0;**

**Left: 0;**

**}**

**Для картинок різних розмірів в строці з текстом збоку:**

**.img\_\_items img{**

**Width: 100%;**

**Height: 100%;**

**Object-fit: cover;**

**}**

**Якщо ми хочемо, щоб картинка була розміром з контент строки:**

**.img\_\_items {**

**Flex: 0 0 33.333% -** для відзивчивої ширини

**Position: relative;**

**Min-height: -** якщо контент знадто малий длякоректного відображення картинки

**}**

**.img\_\_items img{**

**Width: 100%;**

**Height: 100%;**

**Object-fit: cover;**

**Position: absolute;**

**Tor: 0;**

**Left: 0;**

**}**

**Робота з картинками webp (tag picture and source):**

<Source srcset=’src’, type=’image/jpeg’, media=’(min-width: …px)’>

Source – можна продублювати, і вказати інший діапазон медіа і ту ж фотографію, але іншого розміру

**Адаптація шрифтів (відзивчива верстка):**

1. Знати ширину екрана (макета) **W**
2. Розмір шрифта в макеті **S**
3. Розмір шрифта на най меншому діапазоні мобільної версії(який найбільше підходить) **MS**
4. **Метод для адаптації шрифта**: font-size: calc(MS + (S-MS)\*(100vw / W));
5. Для адаптації шрифта після 320px:
   1. Створюємо медіа запит в районі 767px
   2. Вставляємо туди клас\тег з нашою формулою і міняємо формулу
   3. font-size: calc(MS + (S-MS + ((S-MS))\* 0.7)\*((100vw-320px) / W));

**Динамічна адаптивність:**

Спосіб динамічного переміщення DOM елемента по структурі без нагрузки:

Атрибут: **data-da = *“елемент від якого ми відштовхуємось при розміщенні даний елемента, позиція до чи після нього(1, 2\ first, last), range*”**

**Адаптивні відступи для Items-ів:**

1. Задаємо від’ємний margin по бокам – тим самим відступ робить по центру контейнера між items: 50% від потрібного значення
2. Задаємо бічні padding – 50% від необхідного центрального відступу