Сверстать главную страницу новостного ресурса по макету, приложенному к домашнему заданию. Ниже — подробная информация.

**Дополнительные материалы и подготовка**

— Перед выполнением этого домашнего задания вам необходимо ознакомиться со следующими статьями и придерживаться изложенных в них принципов:

* <https://yoksel.github.io/easy-markup/>
* <https://yoksel.github.io/easy-css/>
* <https://yoksel.github.io/bad-practices/>
* <https://css-tricks.com/why-how-and-when-to-use-semantic-html-and-aria/>
* <https://isqua.ru/blog/2016/06/23/content-po-sentru-fon-po-shirinie/>
* <https://github.com/yoksel/common-words>

— Иногда будет возникать проблема с тем, что у какого-то элемента нет высоты. Связано это с тем, что внутри элемента находятся только float-элементы (блоки, имеющие float: left или right). С этим можно и нужно бороться — <https://webformyself.com/ochistka-plavayushhix-elementov-obzor-razlichnyx-metodov-clearfix/>. Либо есть [видео](https://yadi.sk/i/oHwod7LB8w2-WQ).

— Поскольку сайт необходимо сверстать в точном соответствии с макетом (то есть он должен быть “pixel perfect”), рекомендуем пользоваться [специальным расширением](https://chrome.google.com/webstore/detail/perfectpixel-by-welldonec/dkaagdgjmgdmbnecmcefdhjekcoceebi?hl=ru) для Google Chrome. Установите его, настройте и используйте. Так же есть [видео-инструкция](https://yadi.sk/i/1DA71WRqvtnekA). Важно понимать, что идеального совпадения может и не случиться, так как шрифты в браузере по другому рендерятся.

— Некоторые картинки нужно будет доставать “без текста”, или же “без фона”. На этот случай пользуйтесь [инструкцией](https://yadi.sk/i/hbcf8noLEzF_SA).

При выполнении этого домашнего задания методологию БЭМ применять не нужно. Мы будем изучать её позже.

**Общие указания**

**1.**Верстать необходимо с использованием float-ов(flex и grid в этом модуле использовать запрещено), позиционирование применять лишь в нужных случаях (о них будет дополнительно указано ниже в пошаговой инструкции).

**2.** Ширина контейнера-центровщика сайта в макете — 1000 пикселей. Мы оставим его таким, убрав отступы, так как адаптивность рассматривается не в этом модуле, и нам просто не пригодятся отступы. Контейнер-центровщик — это див, который имеет ширину и отцентрован с помощью css свойства margin со значением “0 auto”  - margin: 0 auto; , он служит только для того, чтобы контент сайта не разлетался по краям, а был в центре.

**3.** Задавать фиксированную высоту у элементов(height и max-height) - часто плохая практика, поскольку мы заранее не можем гарантировать, что в данный элемент не будет добавлен дополнительный текстовый или иной контент . Если требуется задать отступы от текста или содержимого элементов до их краёв, необходимо использовать “padding”. Если элемент слишком сильно сжимается по высоте, можно использовать свойство “min-height”.

**4.** Стили нужно писать только по селектору классаВ данный момент это является одной из лучших практик. Например использовать селектор по “id”, в отличие от использования  селектора по классу, является плохой практикой так как селектор по id обладает самой высокой специфичностью, его очень сложно переопределить и невозможно масштабировать (расширять) (добавлять два id к элементу). Например, вы не можете сделать id=”apple red”, но можете class=”apple red”.

**5.** Все иконки необходимо реализовать  через css свойство “background-image”, т.к. это декоративные неконтентные изображения, которые не должны индексироваться поисковыми системами. А вот изображения новостей, в том числе и изображения Крыма и Пекина, должны быть реализованы  используя  img элемент.

**6.** Примерная структура файлов и папок проекта:

    a) файл “index.html”

    b) папка для файлов стилей - “css” папка для картинок проекта - “img”

**7.**Некоторым элементам на странице нужен атрибут aria-label. Он нужен, когда внутри, например, кнопки, нет текстового контента. Чтобы слабовидящий пользователь сумел понять, на какую КОНКРЕТНО кнопку он нажимает и что она сделает - нужно прописать данный атрибут. Пример - aria-label=”Найти на сайте”.

**8.** Перед тем, как отправить код на проверку обязательно нужно проверить его в [валидаторе](https://validator.w3.org/), если есть ошибки нужно их исправить.

**9.** На этом и предыдущих видео для лёгкости восприятия было показано “лестничное” форматирование CSS-кода, при котором стили вложенных элементов располагались правее тех, в которые они вложены. Рекомендуется отказаться от этой практики, начиная с этого модуля, поскольку обычно такое форматирование не используется.

**Инструкция по вёрстке**

Весь сайт условно разделяем на четыре части - шапка (header), центр (три колонки), нижние новости и футер. По ходу инструкции будут приложены ссылки на примеры, однако эти примеры неполные, а лишь приблизительно показывают, как нужно сделать размещение элементов, опуская нюансы разметки.

**Шапка сайта (header) — разметка**

**1.** Элемент <header>, внутри него два главных <div>-элемента: верхний и нижний.

**2.** В верхнем <div> (в нём необходимо использовать контейнер-центровщик) должен быть главный заголовок сайта — <h1> (“Самые свежие новости...”) со ссылкой внутри. Справа  - время , которое нужно разметить с помощью элемента <time>. У этого элемента обязательно необходимо установить атрибут “datetime”. В документе еще много мест, где необходимо будет использовать <time> элемент.

Средняя часть хедера содержит в себе сразу несколько элементов: логотип, элемент выбора города, элемент погоды и форму поиска. Однако, если вы внимательно посмотрите на макет, то увидите, что там есть еще элемент, которые отображает дату и время . Несмотря на то, что он выглядит как отдельные дата и время - это один элемент,  одна временная метка, следовательно для его разметки нужно использовать один элемент <time>, а для добавления иконки часов и отступа, часть текстового контента внутри <time> нужно обернуть в элемент <span>.

**3.** Итак, логотип — это ссылка с картинкой внутри.  Текст логотипа — “Новостное агентство”, должен являться частью картинки(картинка должна состоять из глобуса и текста справа от глобуса), не нужно писать для него(текста из логотипа) отдельный элемент в разметке.

**4.**Далее идет Москва. Поскольку сам по себе селект достаточно сложно стилизуется, мы добавим вокруг него <label> (из 2-го модуля думаю понятно, как <label> работает). [Пример](https://codepen.io/anon/pen/RdPLRQ).

**5.**Далее элемент времени, его мы уже описали выше.

**6.**После элемента времени следует элемент погоды. Это также спан. Сделать символ градуса можно с помощью спец-символа &deg;

**7.**И последний элемент — форма поиска. Внутрь формы необходимо добавить поле ввода и кнопку отправки. Не забывайте, что у поиска есть специальный тип инпута(type=”search”).

**8.**Ну и нижняя часть хедера - навигация. Элемент <nav>, внутри него  будем использовать список(<ul>) ссылок(<a>) . Разделители пунктов меню( / ) в данном модуле делать не будем.

**Шапка сайта (header) - стили**

1. Самое важное — это базовая стилизация, расстановка элементов. В данном случае мы будем использовать float.
2. Позиционирование используем лишь в самых нужных местах, где без них никак. Например, кнопка поиска, или картинка в Крыму, чтобы сделать поверх нее текст. В других случаях оно нам не пригодится.
3. Между элементами используется margin, если нужно отодвинуть элементы “вглубь” — используем padding.
4. Итак, каждому из элементов нужно задать float: left, чтобы все они выстроились в строку. Так, у лого, москвы, времени, погоды и поиска будет флоат. Так же флоат понадобится и меню, которое ниже, иначе произойдет схлопывание блока и будут проблемы.
5. По сути, логотипу стили больше не нужны. Ссылка москвы и погода делаются [по одному принципу](https://codepen.io/MaxGraph/pen/BayVpzO)
6. Для поиска — более сложная стилизация, которая подразумевает использование position: absolute для кнопки поиска, чтобы разместить ее поверх [инпута](https://codepen.io/MaxGraph/pen/KbGwLE)
7. Ну и последнее по хедеру — это стилизация тени под ним. Ее лучше сделать через св-во box-shadow. Подберите верное значение.

**Центральная часть. Левая колонка. Разметка**

**1.** Следом за header начинается основное содержимое документа - основной контент — новости. Для разметки основного содержимого документа используется элемент <main>.

**2.** Взглянув на макет можно увидеть, что основное содержимое явно делится на части — “Новости дня”, центральная колонка новостей — без явного заголовка(позже мы обязательно добавим туда не явный заголовок), блок с рекламой и блок "Самые читаемые новости".  
Сразу бросается в глаза, что все эти блоки отцентрированы. Для реализации центрования блоков нужно обернуть их в один элемент — "КЦ" — контейнер-центровщик(fixed-container).

Блок с рекламой, по сути, не является основным контентом, но дизайнер так расположил этот блок, что нам не остается более адекватного выбора, кроме как поместить рекламу внутри <main>. Для разметки рекламного блока будем использовать элемент <aside>.  
“Новости дня”, центральная колонка новостей, и “cамые читаемые новости” — эти блоки задают основную структуру документа(разделяют документ), следовательно каждый из этих блоков — это секция документа(<section>)

**3.**Секция “Самые читаемые новости”: Первый элемент внутри секции —заголовок. Уровень заголовка определяется обычной последовательностью — у нас уже есть главный заголовок( <h1> ), значит “берем” заголовок на один уровень больше — 2 (элемент <h2>).  
Ниже заголовка мы видим перечисление новостей. Перечисление однотипных элементов можно смело назвать списком (новостей), значит используем элемент списка — <ul>.  
Далее нужно разметить каждую новость, то что внутри <li> (так как прямыми потомками списка, по спецификации могут быть только элементы <li>): новость — это самодостаточная часть документа, для ее разметки будем использовать тег <article> (обязательно прочтите в справочнике раздел по элементу article и section и поймите чем они отличаются и когда нужно использовать один, а когда другой!).  
Внутри новости (<article>) вырисовывается следующая структура:  
Заголовок (уровень заголовка должен быть на один ниже, чем у секции внутри которой расположена новость) — h3  
Ссылка на новостную рубрику — <a>  
Время публикации новости — <time>  
Количество комментариев, для этой сущности нет специального тега, поэтому будем использовать <span>  
Если присмотреться, то мы поймем — в нашей новости есть ссылка на рубрику, но нет ссылки на конкретную новость. Чтобы это исправить, обернем текст внутри заголовка в тег <a> (Или наоборот — заголовок обернем в ссылку). Можно было бы обернуть всю новость( <article> ) в ссылку, но внутри новости есть другая ссылка(на рубрику), а вкладывать одну ссылку в другую по спецификации запрещено.

Все остальные новости и новостные блоки на странице имеют практически одинаковую разметку, где-то внутри новости добавляется картинка, где-то параграф текста, но основная разметка остается неизменной!

**Центральная часть. Левая колонка. Стили**

1. Понятное дело, что колонки нужно расположить в строку. Мы уже делали это раньше с элементами хедера, поэтому просто используйте флоат: лефт.
2. Конкретно левой колонке еще пригодится ширина в 250 пикселей.
3. Основные стили колонки вполне легко написать самим, но чтобы правильно сделать дополнительную часть артикла — есть [пример](https://codepen.io/MaxGraph/pen/xxxLRdd).

**Центральная часть. Центральная колонка. Разметка**

1. В левой колонке мы уже сделали основную разметку. Центральная почти идентичная, поэтому это — на самостоятельную работу.

**Центральная часть. Центральная колонка. Стили**

1. Каждому элементу списка в центральной колонке нужно задать флоат лефт и ширину. Обратите внимание на большую новость. Ее ширина явно больше, чем у остальных, поэтому нужно обратиться к ней как li:first-child, либо просто для первого li добавить класс. и сделать ширину 100%. У остальных же новостей будет ширина 220 пикселей.
2. Между каждой новостью, опять же кроме первой, должен быть отступ в 20 пикселей. Однако, при таком решении возникнет проблема - новости просто не помещаются, т.к. два блока по 220 шириной + два их отступа по 20 = 480. А колонка 460.
3. В таком случае надо убрать у правых новостей отступ с помощью псевдокласса, ищущего [каждую вторую новость](https://codepen.io/MaxGraph/pen/EBzwXv). Поскольку у вас все пять новостей в списке, (2n) изменится на (2n+1).
4. Стилизация главной новости — самое интересное, что здесь есть, ибо нужно придумать, как расположить текст поверх картинки. Для этого нам снова понадобится [абсолют](https://jsfiddle.net/13c24e9y/).

**Центральная часть. Правая колонка. Разметка и стили**

1. Эта колонка довольно простая, ей задаете 250 пикселей ширины
2. Внутри нее — две ссылки с картинками внутри, нужно просто задать отступ между этими ссылками (ссылкам нужен инлайн-блок) и все.

**Нижняя часть. Разметка и стили.**

Поскольку основная разметка новостей, а также расположение элементов флоатами уже разобрано выше - данную колонку можно по сути собрать из сделанного выше. Вам необходимо реализовать её самостоятельно.

**Футер**

Футер также необходимо реализовать самостоятельно. Приводим ниже примеры и описание особо сложных мест:

[Для создания списка в левой части](https://codepen.io/MaxGraph/pen/OJPeeBw). Используем свойство columns для разделения на два столбца.

Правая часть состоит из двух списков — список соц. ссылок (сами иконки должны быть фоновые, как и везде) и списка из ссылок мейла и телефона. Их можно сразу сделать кликабельными, добавив специальные значения tel и

mailto для href.