**Проектная работа «Оно тебе надо»**

[Макет проекта «Оно тебе надо»](https://www.figma.com/file/8KwhMpv8qnDocX4NVFQBpn/%D0%9E%D0%BD%D0%BE-%D1%82%D0%B5%D0%B1%D0%B5-%D0%BD%D0%B0%D0%B4%D0%BE?type=design&node-id=0%3A1&mode=design&t=SFxhzV6vT3fWUSjK-1) немного сложнее, чем готовый кейс, но главное — вы уже понимаете, как всё работает. С каждым макетом вы ближе к цели — верстать и принимать решения о вёрстке самостоятельно.

И всё-таки переживать перед работой над новым проектом — нормально. С этим сталкиваются даже опытные разработчики. Для начала сделайте первый небольшой шаг: откройте макет и оцените, какая интересная работа вас ждёт. После её выполнения вы почувствуете себя гораздо увереннее.

У вас обязательно получится!

**1. Проанализируйте макет и файлы заготовки**

На что стоит обратить внимание в макете:

1. Найдите повторяющиеся компоненты. Например, блок с адресом, оверлей поверх изображений — эти блоки, скорее всего, содержат одинаковый CSS-код. Запишите свои наблюдения, они пригодятся.
2. Посмотрите, какие в макете размеры и толщина шрифтов.
3. Отметьте, что на странице один шрифт.
4. Выпишите элементы, которые кажутся необычными и вызывают вопросы. Что за странные отступы между буквами в первой секции? А что за странные разделители между ссылками меню в шапке? Такие вопросы помогут отделить сложное от простого уже на старте.

Что интересного в заготовке:

* У вас уже есть все изображения. Мы экспортировали их, чтобы вы не тратили на это время.
* Внутри папки fonts/ описан файл шрифтов.
* В папке styles/ есть заготовка global.css и пустой файл для стилей. Обратите внимание на global.css. Это сброс части браузерных настроек, а также некоторые общие стили для всех элементов. Вам это поможет меньше переживать о странном поведении элементов в процессе вёрстки.
* В global.css мы назначили минимальную ширину для body. Макет рассчитан на экраны для ноутбуков. Стили для экранов поменьше вы будете изучать позже. Мы задали предел сжатия макета, чтобы ничего не наезжало друг на друга.

**Уроки, которые помогут**

Чтобы вам было проще освоиться, мы составили список уроков с теорией, которая поможет выполнить проект. В каждом разделе будут только новые уроки. Если урока нет в списке, ориентируйтесь на предыдущие разделы. Воспользуйтесь инструментом поиска, чтобы найти нужный урок по названию.

* «Работа с отступами, текстом и цветами» расскажет как посмотреть в макете шрифт, величину отступов между элементами, цвета и другие стили.
* «О браузерных стилях» напомнит, что такое браузерные стили и какие существуют способы их сброса.
* «Наследование» поможет избежать повторения стилей.

**2. Выполните общие настройки**

Подключите все CSS-файлы в <head> вашего HTML-файла. Сначала шрифты, потом глобальные стили, затем файл с вашими будущими стилями.

Создайте семантические контейнеры для наполнения будущей страницы. В макете есть шапка, блок основного контента с тремя секциями и подвал сайта.

Вот рекомендации по названию классов:

* header;
* для main нам класс не понадобился, но может быть вам понадобятся:
  + cover — для первой секции;
  + lots — для секции с карточками;
  + about — для секции с информацией об аукционе;
* footer.

**Уроки, которые помогут**

* «Структура HTML-документа» и «Разнообразие контейнеров HTML5» помогут создать базовую разметку файла index.html.
* «Относительные пути к файлам» расскажут, как правильно указывать путь к файлу.
* «Подключение шрифтов к странице», «Тонкости настройки шрифтов» и «Подключение внешних шрифтов» помогут настроить и подключить шрифты.
* [«](https://www.notion.so/a6786bcff8ac4357971e2fca2bc905c9?pvs=21)Связь CSS и HTML. CSS-файл[»](https://www.notion.so/a6786bcff8ac4357971e2fca2bc905c9?pvs=21) поможет подключить стили.
* «Страница рассказывает о себе» поможет заполнить метаинформацию страницы.

**3. Разберитесь в особенностях вёрстки шапки**

**Лейаут**

Обратите внимание, что расстояние от последнего пункта меню до логотипа не такое же, как от логотипа до блока с адресом. Логотип выровнен по центру страницы.

С этим нужно что-то придумать. Мы использовали гриды и фракции, но есть и другие варианты.

**Уроки, которые помогут**

* Лонгрид про типы раскладок напомнит о возможных значениях свойства display.
* «Что такое гриды», «Терминология гридов», «Столбцы и строки внутри грид-контейнера», «Единица измерения: фракция» и «Выравнивание сетки в грид-контейнере» напомнят, как пользоваться гридами.
* «Внешние и внутренние отступы» помогут вспомнить, какие отступы бывают и как их задавать.

**Меню**

Разделитель между пунктами меню нестандартный. Вам поможет свойство list-style со значением circle.

Расстояние между пунктами меню не так просто замерить. Может помочь Pixel Perfect Plugin и подгонка через инструменты разработчика.

**Уроки, которые помогут**

* «Семантика и выразительность HTML» поможет написать хорошую разметку шапки и стилизовать пункты меню.
* «Ссылка» поможет вспомнить синтаксис ссылок.
* «Изображения» помогут вспомнить синтаксис для подключения изображений, ещё может пригодиться урок про относительные пути к файлам, на него мы уже ссылались ранее.
* «Преобразование во флекс-контейнер» и «Направление внутри флекс-контейнера» помогут сверстать список меню, а «Настройка отступов с помощью gap» настроить отступы между его элементами.
* «Подход Pixel Perfect» и «Инструменты разработчика» расскажут про нужные инструменты и помогут вспомнить, как заниматься отладкой вёрстки.

**Блок с адресом**

Для адресов есть специальный тег — <address></address>. К этому тегу применяются браузерные стили. Их нужно будет сбросить и написать свои.

Для ссылок с номером телефона применяют конструкцию <a href="tel:+70000000">+7(000)000-00-00</a>, а для ссылок с адресом почты <a href="mailto:mail@example.com">mail@example.com</a>. Так браузер сразу осуществит звонок на номер или перейдёт в почтовый клиент.

Вам может помочь тег <br> — он одиночный, отвечает за перенос текста на следующую строку. Есть небольшое количество случаев, когда его уместно использовать. Один из них — чтобы визуально разделить семантически единый блок контента. Например, ссылки на телефон и почту, идущие одной строкой.

**Уроки, которые помогут**

Ориентируйтесь на упомянутые уроки о браузерных стилях, флексбоксах и семантике.

**Логотип**

Сделайте логотип ссылкой без атрибута href. Изображение необязательно подгонять по размеру, если сделать ссылку блочной и задать размеры. В файле globals.css мы сделали все изображения размером в ширину родителя.

**Уроки, которые помогут**

Ориентируйтесь на указанные уроки о ссылках и изображениях.

**Рекомендуемые имена классов**

* header — вся шапка;
* header\_\_menu — контейнер навигации;
* header\_\_links-list — список ссылок;
* header\_\_links-list-item — пункт списка ссылок;
* header\_\_links-list-item\_no-bullit — вам может потребоваться этот класс, чтобы сделать ссылку без кружочка перед ней;
* header\_\_link — ссылка;
* header\_\_link\_active — активная ссылка;
* header\_\_logo — контейнер логотипа;
* header\_\_logo-image — картинка с логотипом;
* address — контейнер с адресом;
* header\_\_address — потребуется, чтобы выравнивать блок с адресом. В футере используется другое выравнивание.

Мы обошлись без использования других классов, где-то применив селекторы вложенности. Вы можете пойти другим путём, это окей.

**4. Разберитесь в особенностях вёрстки входной секции**

На наш взгляд, это самая необычная секция на сайте. Здесь разные отступы между буквами и даже между словами. Скорее всего, хорошим подходом будет сверстать её сначала примерно, а потом уточнить все размеры через наложение изображения в Pixel Perfect плагине для браузера.

**Вёрстка секции**

У вас есть секция с фоном. Мы не использовали управление высотой блока, но пользовались внутренними отступами и собственными размерами элементов.

Поверх секции лежит оверлей. Это пустой полупрозрачный блок. Сверстайте его так, чтобы потом применить такой же класс к любому другому оверлею на странице.

Сознание может сопротивляться появлению пустого блока в разметке. Это хорошо. Уже скоро вы научитесь обходиться без таких пустых контейнеров, но всему своё время.

Вам придётся использовать позиционирование при вёрстке, поэтому следите за свойством z-index у всех элементов внутри секции. Они могут спрятаться под оверлеем.

**Уроки, которые помогут**

* «Фон элемента» и «Позиция, размер, повтор фона» помогут разместить фоновое изображение для блока cover.
* «Понятие потока и статическое позиционирование», «Относительное позиционирование», «Абсолютное позиционирование» и «Прозрачность» помогут сделать оверлей.
* «Свойство inset, упрощающее жизнь» и «z-index» помогут расположить оверлей.

**Заголовок**

Вся надпись «Оно вам надо» — семантически единый заголовок первого уровня. Для переноса строк и отступов между буквами используйте вложенный элемент <span> с дополнительными классами.

Вам потребуется несколько видов <span>-обёрток:

* для выравнивания строк;
* для управления расстоянием между буквами свойством letter-spacing;
* для обнуления расстояния между буквами у последней буквы.

Особенность макета в том, что расстояние между буквами необычное как во всём тексте целиком, так и у отдельных букв. Так что свойство letter-spacing придётся задать и всему заголовку, и отдельным обёрткам.

Ещё нужно разобраться с интерлиньяжем и отступами между строками. Ох уж этот дизайн!

**Уроки, которые помогут**

* Урок «Заголовки» поможет вспомнить, какие уровни заголовков существуют.
* «Типографика» поможет оформить текст и выровнять строки.

**Подзаголовок и кнопка**

Два элемента мы объединили в один контейнер. У подзаголовка нестандартное расстояние между словами. Управлять расстоянием между словами можно свойством word-spacing со значениями в пикселях. Ваша задача — подобрать эти значения.

Чтобы стилизовать кнопку, нужно сбросить её настройки по умолчанию — изменить фон на transparent (прозрачный), переопределить стиль границ и внутренние отступы. Это не все свойства, которые нужны кнопке. Остальные вы увидите и добавите сами.

**Уроки, которые помогут**

* «Выравнивание по основной оси» и «Выравнивание по дополнительной оси» помогут расположить элементы внутри блока.
* «Абзац» поможет разметить основной текст.
* «Мысли о стилях. CSS» напомнят о синтаксисе кнопок и их стилизации.

**Рекомендуемые имена классов**

* cover — вся секция;
* overlay — полупрозрачный блок поверх секции;
* cover\_\_title — заголовок;
* aligned-text, aligned-text\_center и aligned-text\_right — классы для выравнивания строк текста и превращения их в блочный элемент;
* letter-spacing-extended — класс для управления межбуквенным расстоянием в местах, где есть эти акценты;
* no-letter-spacing — класс для обнуления общего межбуквенного расстояния для последней буквы;
* cover\_\_description — контейнер для подзаголовка и кнопки;
* cover\_\_description-text — текст подзаголовка;
* bet-button — кнопка «Сделать ставку».

**5. Разберитесь в особенностях вёрстки секции с карточками**

В этой секции большая часть вопросов может возникнуть при вёрстке списка карточек. Мы разберём только его. С заголовком и ссылкой вы точно справитесь сами.

**Список карточек**

Для блока стоит использовать семантику списка с вложенными пунктами списка. В каждом пункте списка — ссылка, а внутри ссылки — элемент карточки (рекомендуем использовать тег <article>).

Списку не нужны буллиты или номера пунктов. Вы можете использовать его как флекс или грид-контейнер в зависимости от выбранного способа построения лейаута. Мы выбрали флексбокс, добавив к нему свойство переноса на новую строку. Нам показалось классным сделать перенос карточек на минимально доступной ширине.

Каждый пункт списка мы ограничили шириной карточки, а также задали ему минимальную высоту. Эта высота при дальнейшей вёрстке поможет расположить контент карточки согласно макету.

Неплохое решение — задать пункту списка flex-grow: 1. Так при переносе на новую строку карточка начнёт подстраиваться под ширину контейнера. Это классный эффект из мира адаптивности, в который вам ещё только предстоит погрузиться.

**Уроки, которые помогут**

* «Flex basis, flex-grow и flex-shrink» поможет сделать карточку, подстраивающуюся под ширину контейнера.
* «Перенос флекс-элементов» расскажет, как включить перенос карточек на новую строку, если не хватает места в текущей строке.

**Карточка**

Карточка лежит в ссылке, а ссылка должна стать блочной и без подчёркиваний текста. Ещё потребуется «прокидывать» высоту от родителя, то есть указать высоту в 100%.

Самой карточке стопроцентная высота тоже понадобится. Когда сверстаете, попробуйте убрать её — увидите, что она нужна. Внутри карточки лежит оверлей. Тот самый, который вы уже создали для входного экрана. Контент внутри карточки лучше распределять через флексбокс. И следите за z-index, ведь ваш оверлей спозиционирован абсолютно.

**Уроки, которые помогут**

Чтобы выполнить это задание, ориентируйтесь на предыдущие ссылки: флекбокс, выравнивание по осям флекс-контейнера, z-index, фон и относительное позиционирование.

**Рекомендуемые имена классов**

* lots — вся секция;
* lots\_\_heading — заголовок секции;
* lots\_\_card-list — список карточек;
* lots\_\_card-list-item — пункт списка карточки;
* card-link — ссылка, обрамляющая карточку;
* card — карточка;
* card\_type\_film, card\_type\_book и card\_type\_picture — классы, чтобы задать определённый фон каждой из карточек;
* overlay — он уже есть у вас в коде;
* card\_\_title — заголовок карточки;
* card\_\_text — текст карточки;
* lots\_\_look-more-link — ссылка «Посмотреть больше».

**6. Поработайте над секцией «Об аукционе»**

**Лейаут**

В секции две колонки. Найдите такое решение, чтобы левая колонка была такого же размера, как карточка выше, а правая занимала всё оставшееся пространство. Так задумано в дизайне.

**Уроки, которые помогут**

Чтобы выполнить задание, ориентируйтесь на предыдущие ссылки: гриды, столбцы и строки грид-контейнера и выравнивание внутри грид-контейнера.

**Логотип**

Здесь нужен родительский контейнер, который вы сделаете круглым и чёрным. По центру этого контейнера разместите белый логотип. Контейнер разместите по горизонтальному центру левой колонки.

**Уроки, которые помогут**

Урок «Скругления» напомнит, как сделать круглый фон для логотипа.

**Колонка с текстом**

Заголовок в правой колонке должен оказаться на одной линии с центром круга в левой колонке. Мы использовали для этого верхние и нижние поля.

Текст состоит из двух абзацев. Придумайте, как задать нижний отступ только первому абзацу. Второму абзацу нижний отступ не нужен.

**Уроки, которые помогут**

Ориентируйтесь на предыдущие ссылки про флексбокс и настройку отступов.

**Рекомендуемые имена классов**

* about — вся секция;
* about\_\_column — класс для колонок;
* about\_\_logo — контейнер с логотипом;
* about\_\_logo-image — изображение логотипа;
* about\_\_title — заголовок секции;
* about\_\_text — блок для текстов секции.

Другие классы мы не использовали, но вам они могут понадобиться. У задачи несколько решений. Это нормально в гибком мире лейаутов.

**7. Не забудьте про подвал сайта**

Самая простая секция сайта. Здесь, как нам кажется, меньше всего подводных камней.

* Вы уже верстали левую колонку. Переиспользуйте блок <address>. Учтите, что здесь другое выравнивание внутри контейнера, не такое, как в шапке.
* Центральная колонка — это список ссылок с текстом, выровненным по центру.
* Правая колонка — список со ссылками-изображениями.

Мы уверены, что эту часть сайта вы осилите без дополнительных подсказок.

**Уроки, которые помогут**

Чтобы выполнить задание, ориентируйтесь на предыдущие ссылки: семантика, гриды, столбцы и строки грид-контейнера, флексбокс и выравнивание по его осям.

**Рекомендуемые имена классов**

* footer — подвал;
* address — такой класс уже есть в вашей вёрстке, все вложенные элементы аналогичны;
* footer\_\_menu — блок навигации по списку ссылок. Мы рекомендуем окружать список тегом <nav> с этим классом;
* footer\_\_menu-list — список пунктов меню;
* footer\_\_menu-list-item — пункт списка меню;
* footer\_\_menu-link — ссылка меню;
* footer\_\_menu-link\_active — активная ссылка меню (без подчёркивания);
* footer\_\_social-list — список ссылок на социальные сети. Можно не добавлять тег <nav>, потому что это ссылки на внешние ресурсы;
* footer\_\_social-list-item — пункт списка ссылок на социальные сети;
* footer\_\_social-link — ссылка на социальные сети;
* footer\_\_social-icon — иконка социальной сети.

**8. Вот и всё!**

Первый проект — это всегда непросто, но захватывающе. Ещё недавно вы только начинали обучение, а теперь обладаете базовыми навыками, которых достаточно для работы над полноценным кейсом. Мы радуемся вместе с вами.

Помните: если у вас не получается выполнить какую-то часть проекта — не стесняйтесь задавать вопросы наставникам. Вам помогут.