

## DOM III. Регистрация

Новый шаблон разметки:

[HTML разметка](#)

[CSS стили](#)

В этом задании вам предстоит создать динамическую страницу для регистрации. Регистрация будет разбита на шаги, которые будут последовательно сменять друг друга и изменять отображаемый контент на странице.

1. Но сначала сделаем лоадер. Лоадер – это такая штука, которая отображает состояние загрузки содержимого страницы. Крутящееся колесо, бегущая полоса и т.д. Их принято вставлять на страницы, которые относительно долго загружаются, чтобы пользователь понимал, что загрузка продолжается и всё идёт по плану.

Создайте лоадер с заполняющейся за 2 секунды шкалой, как показано далее:



Заполнение шкалы можно реализовать с помощью функции *setInterval* в JS, изменяя через свойство *style* CSS свойства элемента (например, ширину).

В шаблоне HTML страницы внутри *body* есть 2 блока: *section* и *main*. Лоадер нужно создать внутри блока *section*, а основной контент страницы будет находится в *main*.

Мы будем симулировать длительную загрузку, поэтому при открытии страницы должен отобразиться лоадер (основной контент страницы, т.е. *main* должен быть скрыт, например, с помощью *display: none*). И после его загрузки, через 2 секунды, лоадер (т.е. *section* в котором он находится) должен скрыться, а основная страница (*main*) стать видимой.

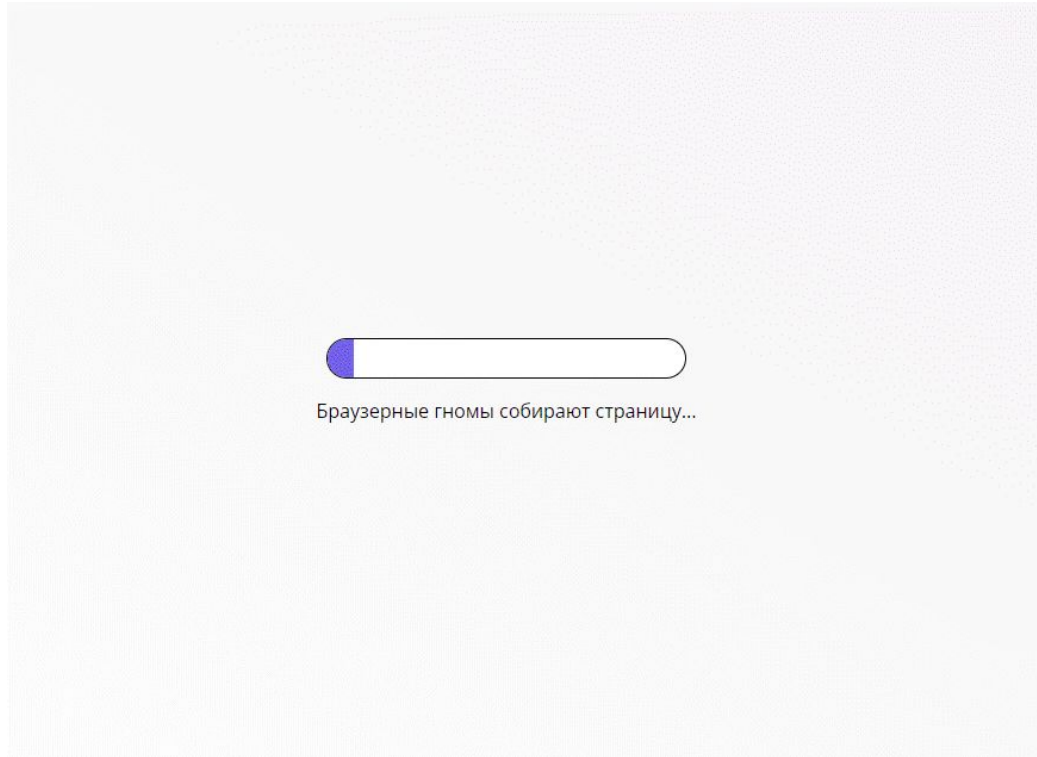
И ещё один момент: стили, используемые в части страницы, описывающей лоадер (*section*), и js код для его функционирования должны подключаться отдельными файлами! Причём JS файл лучше подключить

сразу после блока ладера. Ладер должен отобразиться сразу в начале загрузки страницы, и в реальных проектах, когда страница долго загружается по-настоящему, такое расположение и разделение подключаемых файлов может оказаться необходимым для его корректной работы.

Т.е. подключаться файлы должны как-то так:

```
10 <link rel="stylesheet" href="style.css">
11 <link rel="stylesheet" href="loader.css">
12 <title>Регистрация в Сервис</title>
13 </head>
14 <body>
15 <header></header>
16 <section id="loader">...
24 </section>
25 <script src="loader.js"></script>
26 <main id="page" class="invisible">
27 <div>
28 <section id="frame_1">
29 <p>Добро пожаловать в <span>Сервис</span>!</p>
30 <small>Для продолжения нужно зарегистрироваться</small>
31 <button type="button" name="reg">Регистрация</button>
32 </section>
33 </div>
34 </main>
35 <script src="code.js"></script>
36 </body>
```

Пример того, что должно получиться:



## 2. Теперь перейдём к описанию процедуры регистрации.

Регистрация будет разбита на 3 шага: ввод персональных данных, ввод и подтверждение телефонного номера, подключение средств оплаты. Все шаги будут выполняться на странице в рамках одного и того же блока, через динамическую смену содержимого.

Первый шаг: персональные данные.

В коде HTML странице внутри блока `main` находится блок *section* с `id="frame_1"`. Этот *section* описывает текущее содержимое в прямоугольнике по центру страницы. Для того, чтобы создавать новое содержимое вам нужно рядом с этим блоком *section* создать ещё один блок *section*, и писать весь код для этого шага в нём. Для следующих шагов также будет добавляться новый *section*. Суть в том, что в один момент времени должен будет (в соответствии с логикой, которую вам нужно прописать) отображаться только один блок, только одна секция. А остальные должны находиться в скрытом состоянии.

Т.е. Когда пользователь на первоначальной странице нажмёт кнопку “Регистрация”, видимость секций должна быть изменена таким образом, чтобы старая секция перестала отображаться, а на её месте появилась новая.

Фрейм для *section* с персональными данными должен быть похожим на такое:

The image shows a web form titled "Шаг 1. Персональные данные" (Step 1. Personal data). It contains several input fields arranged in two columns. The left column has fields for "Имя:" (Name), "Электронная почта:" (Email), "Страна:" (Country), "Пароль:" (Password), and "Повторите пароль:" (Repeat password). The right column has fields for "Фамилия:" (Surname) and "Адрес:" (Address). At the bottom right of the form are two buttons: "Отмена" (Cancel) and "Далее" (Next). The form is enclosed in a light blue border.

По нажатию кнопки “Отмена” содержимое страницы должно изменяться на первоначальное.

Кнопка “Далее” изначально должна быть заблокированной для нажатия. Чтобы на неё можно было нажать, пользователем должны быть заполнены все текстовые поля на этой странице (пароли в полях должны совпадать, разумеется). Если значения везде присутствуют, кнопка должна стать активной, и по её нажатию должен произойти переход к следующему шагу.

\*Дополнительно: Для поля “Страна” сделайте не простое текстовое поле, а выпадающий список с выбором всех существующих стран. Список стран, чтобы можно было скопировать и вставить, найдите в Интернете. Заполнить выпадающий список значениями нужно через JS.

\*Дополнительно: на этом и последующих шагах, где это актуально, напишите валидацию значений при помощи регулярных выражений (только буквы и основные символы в именах, корректный формат адреса электронной почты с @, и т.д.).

3. Второй шаг: телефонный номер.

