**Описание функциональности**

[**1.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-1)**Состояния страницы**

* [1.1. Неактивное состояние.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-1-1) При первом открытии, страница находится в неактивном состоянии: блок с картой находится в неактивном состоянии, форма подачи заявления заблокирована.
  + Блок с картой .map содержит класс map--faded;
  + Форма заполнения информации об объявлении .ad-form содержит класс ad-form--disabled;
  + Все <input> и <select> формы .ad-form заблокированы с помощью атрибута disabled, добавленного на них или на их родительские блоки fieldset;
  + Форма с фильтрами .map\_\_filters заблокирована так же, как и форма .ad-form;
  + Единственное доступное действие в неактивном состоянии — перемещение **метки .map\_\_pin--main**, являющейся контролом указания адреса объявления. Первое взаимодействие с меткой (mousedown) переводит страницу в активное состояние.
* [1.2. Активное состояние.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-1-2) В активном состоянии страница позволяет вносить изменения в форму и отправлять её на сервер, просматривать похожие объявления на карте, фильтровать их и уточнять подробную информацию о них, показывая для каждого из объявлений карточку.

[**2.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2)**Заполнение информации**

* [2.1.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-1) Заполнение информации и отправка данных:
  + фотография пользователя;
  + заголовок;
  + адрес;
  + тип жилья;
  + цена за ночь;
  + количество комнат;
  + количество спальных мест;
  + время заезда и выезда из квартиры;
  + дополнительные параметры:
    - парковка;
    - WiFi;
    - кондиционер;
    - кухня;
    - стиральная машина;
    - лифт.
  + свободное текстовое описание;
  + фотография жилья.
* [2.2.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-2) Заполнение всей информации производится на одной странице без промежуточных переходов. Порядок заполнения информации не важен.
* [2.3.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-3) После заполнения всех данных, при нажатии на кнопку «Опубликовать», все данные из формы, включая изображения, с помощью AJAX-запроса отправляются на сервер <https://javascript.pages.academy/keksobooking> методом POST с типом multipart/form-data.
* [2.4.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-4) Страница реагирует на неправильно введённые значения в форму. Если данные, введённые в форму, не соответствуют ограничениям, указанным в [разделе](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3), описывающем поля ввода, форму невозможно отправить на сервер. При попытке отправить форму с неправильными данными, отправки не происходит, а неверно заполненные поля подсвечиваются красной рамкой. Способ добавления рамки и её стиль произвольные.
* [2.5.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-5) При успешной отправке формы страница, не перезагружаясь, переходит в изначальное [неактивное состояние](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-1-1), а также:
  + все заполненные поля возвращаются в изначальное состояние, в том числе фильтры;
  + метки похожих объявлений и карточка активного объявления удаляются;
  + метка адреса возвращается в исходное положение;
  + значение поля адреса корректируется соответственно положению метки.
* [2.6.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-6) Если отправка данных прошла успешно, показывается соответствующее сообщение. Разметку сообщения, которая находится блоке #success внутри шаблона template, нужно разместить в <main>. Сообщение должно исчезать по нажатию на клавишу Esc и по клику на произвольную область экрана за пределами блока с сообщением.
* [2.7.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-7) Если при отправке данных произошла ошибка запроса, показывается соответствующее сообщение. Разметку сообщения, которая находится в блоке #error в шаблоне template, нужно разместить в <main>. Сообщение должно исчезать после нажатия на кнопку .error\_\_button, по нажатию на клавишу Esc и по клику на произвольную область экрана за пределами блока с сообщением.
* [2.8.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-2-8) Нажатие на кнопку .ad-form\_\_reset сбрасывает страницу в исходное [неактивное состояние](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-1-1) без перезагрузки, а также:
  + все заполненные поля возвращаются в изначальное состояние, в том числе фильтры;
  + метки похожих объявлений и карточка активного объявления удаляются;
  + метка адреса возвращается в исходное положение;
  + значение поля адреса корректируется соответственно положению метки;

[**3.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3)**Ограничения, накладываемые на поля ввода**

* [3.1.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-1) Заголовок объявления:
  + Обязательное текстовое поле;
  + Минимальная длина — 30 символов;
  + Максимальная длина — 100 символов.
* [3.2.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-2) Цена за ночь:
  + Обязательное поле;
  + Числовое поле;
  + Максимальное значение — 1 000 000.
* [3.3.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-3) Поле «Тип жилья» влияет на минимальное значение поля «Цена за ночь»:
  + «Бунгало» — минимальная цена за ночь 0;
  + «Квартира» — минимальная цена за ночь 1 000;
  + «Дом» — минимальная цена 5 000;
  + «Дворец» — минимальная цена 10 000.

**Обратите внимание:** вместе с минимальным значением цены нужно изменять и плейсхолдер.

**Обратите внимание:** ограничение минимальной цены заключается именно в изменении минимального значения, которое можно ввести в поле с ценой, изменять само значение поля не нужно, это приведёт к плохому UX (опыту взаимодействия). Даже если текущее значение не попадает под новые ограничения не стоит без ведома пользователя изменять значение поля.

* [3.4.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-4) Адрес: ручное редактирование поля запрещено. Значение автоматически выставляется при перемещении метки .map\_\_pin--main по карте. Подробности заполнения поля адреса, описаны вместе с поведением метки.
* [3.5.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-5) Поля «Время заезда» и «Время выезда» синхронизированы: при изменении значения одного поля, во втором выделяется соответствующее ему. Например, если время заезда указано «после 14», то время выезда будет равно «до 14» и наоборот.
* [3.6.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-6) Поле «Количество комнат» синхронизировано с полем «Количество мест» таким образом, что при выборе количества комнат вводятся ограничения на допустимые варианты выбора количества гостей:
  + 1 комната — «для 1 гостя»;
  + 2 комнаты — «для 2 гостей» или «для 1 гостя»;
  + 3 комнаты — «для 3 гостей», «для 2 гостей» или «для 1 гостя»;
  + 100 комнат — «не для гостей».

**Обратите внимание:** допускаются разные способы ограничения допустимых значений поля «Количество мест»: удаление из разметки соответствующих элементов option, добавление элементам option состояния disabled или другие способы ограничения, например, с помощью метода setCustomValidity.

* [3.7.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-3-7) Значением полей «Ваша фотография» и «Фотография жилья» может быть только изображение.

[**4.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-4)**Выбор адреса на карте:**

* [4.1.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-4-1) Приблизительный адрес квартиры указывается перемещением специальной метки по карте Токио. Содержимое поля адреса должно соответствовать координатам метки:
  + в неактивном состоянии страницы метка круглая, поэтому в поле адреса подставляются координаты центра метки;
  + при переходе страницы в активное состояние в поле адреса подставляются координаты острого конца метки;
  + при перемещении (mousemove) метки в поле адреса подставляются координаты острого конца метки.
* [4.2.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-4-2) Поле адреса должно быть заполнено всегда, в том числе сразу после открытия страницы (в неактивном состоянии).
* [4.3.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-4-3) Формат значения поля адреса: {{x}}, {{y}}, где {{x}} и {{y}} это координаты, на которые метка указывает своим острым концом. Например, если метка .map\_\_pin--main имеет CSS-координаты top: 200px; left: 300px, то в поле адрес должно быть записано значение 300 + расстояние до острого конца по горизонтали, 200 + расстояние до острого конца по вертикали. Координаты не должны быть дробными.
* [4.4.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-4-4) Для удобства пользователей значение Y-координаты адреса должно быть ограничено интервалом от 130 до 630. Значение X-координаты адреса должно быть ограничено размерами блока, в котором перемещается метка.
* [4.5.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-4-5) При ограничении перемещения метки по горизонтали её острый конец должен указывать на крайнюю точку блока. При выходе за границы блока часть метки скрывается. Скрытие реализовано стилями блока.

**Обратите внимание:** пункт про разницу CSS-координат и координат адреса справедлив для меток всех объявлений — и главной метки, и неглавных меток. Координаты адреса X и Y, которые вы вставите в разметку, это не координаты левого верхнего угла блока метки, а координаты, на которые указывает метка своим острым концом. Чтобы найти эту координату нужно учесть размеры элемента с меткой.

[**5.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5)**Сравнение с похожими объявлениями**

* [5.1.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-1) Полный список похожих объявлений загружается после перехода страницы в активное состояние с сервера <https://javascript.pages.academy/keksobooking/data>. **Данные с сервера могут быть получены не в полном объёме.**
* [5.2.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-2) Каждое из объявлений показывается на карте в виде специальной метки: блока, имеющего класс map\_\_pin. Шаблонный элемент для метки .map\_\_pin находится в шаблоне template. Разметка каждой из меток должна создаваться по аналогии с шаблонным элементом. **Если в объекте с описанием объявления отсутствует поле offer, то метка объявления не должна отображаться на карте.**
* [5.3.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-3) При нажатии на метку похожего объявления, показывается карточка, содержащая подробную информацию об объявлении (сразу после перехода в [активный режим](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-1-2), карточка не отображается). При этом активной метке добавляется класс .map\_\_pin--active (с других меток он, соответственно, должен удаляться). Разметка карточки должна создаваться на основе шаблонного элемента .map\_\_card, расположенного в элементе template. Данные в карточку вставляются по аналогии с данными, вставленными в шаблонную карточку в качестве примера. **Если данных для заполнения не хватает, соответствующий блок в карточке скрывается. Например, если в объявлении не указано никаких удобств, нужно скрыть блок .popup\_\_features. При отсутствии полей не должно возникать ошибок.**
* [5.4.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-4) Нажатие на метку .map\_\_pin--main не приводит к показу карточки.
* [5.5.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-5) В каждый момент времени может быть открыта только одна карточка, то есть нажатие на метку другого похожего объявления должно закрывать текущую карточку, если она открыта и показывать карточку, соответствующую другому объявлению.
* [5.6.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-6) Открытую карточку с подробной информацией можно закрыть или нажатием на иконку крестика в правом верхнем углу объявления или нажатием на клавишу Esc на клавиатуре.
* [5.7.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-7) Объекты, расположенные неподалёку, можно фильтровать. Фильтрация производится при изменении значений соответствующих полей формы .map\_\_filters по тем же параметрам, которые указываются для объявления:
  + тип жилья;
  + цена за ночь;
  + число комнат;
  + число гостей;
  + дополнительные удобства.
* [5.8.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-8) Как до изменения фильтров, так и при изменении фильтра, на карте должны показываться все подходящие варианты, но не более пяти меток, независимо от выбранного фильтра.
* [5.9.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-9) Форма, с помощью которой производится фильтрация похожих объявлений на момент открытия страницы заблокирована и становится доступной только после окончания загрузки всех похожих объявлений.
* [5.10.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-10) Отрисовка соответствующих, выставленных фильтрам элементов, должна происходить не чаще чем раз в полсекунды (устранение дребезга).
* [5.11.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-5-11)  При изменении фильтров, карточка, показывающая подробную информацию о похожем объявлении должна быть скрыта.

[**6.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-6)**Доступность и активные элементы:**

* [6.1.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-6-1) Взаимодействие со всеми активными элементами на странице должно быть доступно не только с помощью курсора и кликов на них, но и с помощью клавиатуры: все активные элементы должны фокусироваться и реагировать на нажатие клавиши Enter так же, как и на клик.

[**7.**](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-7)**Необязательная функциональность**

* [7.1.](https://up.htmlacademy.ru/javascript/20/project/keksobooking#keksobooking-7-1) В форме подачи объявления показывается аватарка пользователя и фотография помещения при изменении значений соответствующих полей.