

# Проект: Кексобукинг

- Название сайта: **Кексобукинг**.
  - Кексобукинг — сервис размещения объявлений о сдаче в аренду недвижимости в центре Токио. Пользователям предоставляется возможность размещать объявления о своей недвижимости или просматривать уже размещённые объявления.
- 

## Описание функциональности

### Состояния страницы

**Неактивное состояние.** 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` ;
- Поля формы `.ad-form` заблокированы с помощью атрибута `disabled` , добавленного на них или на их родительские блоки `fieldset` .

Единственное доступное действие в неактивном состоянии — перетаскивание **метки** `.map__pin--main` , являющейся контролом указания адреса объявления. Первое перемещение метки переводит страницу в активное состояние.

**Активное состояние.** 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.

## 1. Заполнение информации

### 1.1. Заполнение информации и отправка данных:

- заголовок;
- адрес;
- вид недвижимости;
- количество комнат;
- количество спальных мест;
- время заезда и выезда из квартиры.
- дополнительные параметры:
  - парковка;
  - Wi-Fi;
  - кондиционер;
  - кухня;
  - стиральная машина;
  - лифт;
- фотографии
- свободное текстовое описание

**1.2.** Заполнение всей информации производится на одной странице без промежуточных переходов. Порядок заполнения информации не важен.

1.3. После заполнения всех данных, при нажатии на кнопку «Опубликовать», все данные из формы, включая изображения, с помощью AJAX-запроса отправляются на сервер <https://js.dump.academy/keksobooking> методом `POST` с типом `multipart/form-data`.

1.4. Страница реагирует на неправильно введённые значения в форму. Если данные, введённые в форму, не соответствуют ограничениям, указанным в разделе, 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.

1.5. При успешной отправке формы, страница, не перезагружаясь, переходит в изначальное неактивное состояние: все заполненные поля стираются, метки похожих объявлений и карточка активного объявления удаляются, метка адреса возвращается в исходное положение, значение поля адреса корректируется соответственно положению метки. Показывается сообщение об успешной отправке формы: у блока `.success` нужно убрать класс `hidden`. Нажатие на кнопку `Esc` или клик по произвольной области страницы скрывает блок `.success`, добавляя ему класс `hidden`.

1.6. Если при отправке данных произошла ошибка запроса, показывается соответствующее сообщение. Стиль и содержимое сообщения произвольные. Должна быть предусмотрена возможность закрыть сообщение или же оно само должно исчезать через некоторое время.

1.7. Нажатие на кнопку `.ad-form__reset` сбрасывает страницу в исходное неактивное состояние без перезагрузки: все заполненные поля стираются, метки похожих объявлений и карточка активного объявления удаляются, метка адреса возвращается в исходное положение, значение поля адреса корректируется соответственно положению метки.

## 2. Ограничения, накладываемые на поля ввода

2.1. Заголовок объявления:

- Обязательное текстовое поле;
- Минимальная длина — 30 символов;
- Максимальная длина — 100 символов;

2.2. Цена за ночь:

- Обязательное поле;
- Числовое поле;
- Максимальное значение — 1 000 000;

2.3. 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;

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. Даже если текущее значение не попадает под новые ограничения не стоит без ведома пользователя изменять значение поля.

## 2.4. 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.

2.5. 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.

2.6. 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»;

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. Выбор адреса на карте:

3.1. Приблизительный адрес квартиры указывается перетаскиванием специальной метки по карте Токио. При перемещении изменяется значение соответствующего поля ввода. Содержимое поле адреса не может быть пустым: сразу после загрузки страницы и после сброса формы, значение должно соответствовать координатам метки.

3.2. Формат значения поля адреса: `{{x}}, {{y}}`, где `{{x}}` и `{{y}}` это координаты, на которые метка указывает своим острым концом (середина нижнего края блока с меткой).

3.3. Обратите внимание на то, что координаты по X и Y, соответствующие адресу, должны высчитываться не от левого верхнего угла блока с меткой, а от места, куда указывает метка своим острым концом.

3.4. Чтобы метку невозможно было поставить выше горизонта или ниже панели фильтров, значение координаты Y должно быть ограничено интервалом от 130 до 630. Значение координаты X должно быть ограничено размерами блока, в котором перетаскивается метка.

### Например

Если метка `.map_pin--main` имеет координаты 300×200, то в поле адрес должно быть записано значение `300, 200`.

3.5. При сбросе формы, страница возвращается в исходное неактивное состояние и метка перемещается на изначальные координаты. Соответствующее значение поля ввода адреса так же должно быть обновлено.

## 4. Сравнение с похожими объявлениями

4.1. Полный список похожих объявлений загружается после перехода страницы в активное состояние с сервера <https://js.dump.academy/keksobooking/data>. 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.

4.2. При нажатии на метку похожего объявления, показывается карточка, содержащая подробную информацию об объявлении. 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. В случае если данных для заполнения не хватает, соответствующий блок в карточке скрывается. Например, если в объявлении не указано никаких удобств, нужно скрыть блок `.popup_features`. Сразу после перехода в активный режим, карточка не отображается, она показывается только после нажатия на одну из меток. При этом активной метке добавляется класс `.map_pin--active`. Нажатие на метку `.map_pin--main` не приводит к показу карточки.

4.3. В каждый момент времени может быть открыта только одна карточка, то есть нажатие метку другого похожего объявления должно закрывать текущую карточку, если она открыта и показывать карточку, соответствующую другому объявлению.

4.4. Открытую карточку с подробной информацией можно закрыть или нажатием на иконку крестика в правом верхнем углу объявления или нажатием на клавишу Esc на клавиатуре.

4.5. Объекты, расположенные неподалёку, можно фильтровать. Фильтрация производится по тем же параметрам, которые указываются для объявления:

- тип жилья;
- цена за ночь;
- число комнат;
- число гостей;
- дополнительные удобства;

Фильтрация производится при изменении значений соответствующих полей формы `.map_filters`.

4.6. 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.

4.7. Блок, с помощью которого производится фильтрация похожих объявлений на момент открытия страницы заблокирован и становится доступным только после окончания загрузки всех похожих объявлений.

4.8. Отрисовка соответствующих, выставленным фильтрам элементов, должна происходить не чаще чем раз в полсекунды (*устранение дребезга*).

4.9. При изменении фильтров, карточка, показывающая подробную информацию о похожем объявлении должна быть скрыта.

## 5. Доступность и активные элементы:

5.1. 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.

## 6. Необязательная функциональность

6.1. В форме подачи объявления показывается аватарка пользователя и фотографии объявления при изменении значений соответствующих полей.