**Functionality**

**No todos**

When there are no todos, #main and #footer should be hidden.

**Новый todo**

Новые задачи вводятся в верхней части приложения. Элемент ввода должен быть сфокусирован при загрузке страницы, предпочтительно с использованием атрибута ввода автофокуса. Нажатие Enter создает задачу, добавляет ее в список задач и очищает ввод. Убедитесь, что .trim () ввод, а затем убедитесь, что он не пустой, прежде чем создавать новый todo.

**Отметить все как завершенное**

Этот флажок переключает все задачи в то же состояние, что и он сам. Обязательно снимите отмеченное состояние после нажатия кнопки «Очистить завершено». Флажок «Пометить все как завершенное» также должен быть обновлен, когда отдельные элементы задачи отмечены / сняты. Например. Когда все задачи проверены, они также должны быть проверены.

**Item**

У элемента есть три возможных взаимодействия:

1. Нажав на флажок, можно пометить задачу как завершенную, обновив ее завершенное значение и переключив класс, завершенный, на его родительский <li>
2. Двойной щелчок на <label> активирует режим редактирования, переключая класс .editing на его <li>
3. При наведении курсора на задачу отображается кнопка удаления (.destroy)

**Editing**

Когда режим редактирования активирован, он скрывает другие элементы управления и переносит ввод, содержащий заголовок задачи, который должен быть сфокусирован (.focus ()). Редактирование должно быть сохранено как для размытия, так и для ввода, а класс редактирования должен быть удален. Убедитесь, что .trim () ввод, а затем убедитесь, что он не пустой. Если он пуст, то вместо этого следует уничтожить задачу. Если во время редактирования нажимается escape, состояние редактирования должно быть оставлено, а любые изменения отменены.

**Counter**

Отображает количество активных задач в множественной форме. Убедитесь, что номер обернут тегом <strong>. Также убедитесь, что вы правильно указали слово «0», «1», «2». Пример: осталось 2 предмета

**Clear completed button**

Удаляет завершенные задачи при нажатии. Должен быть скрыт, когда нет завершенных задач

**Persistence Упорство**

Ваше приложение должно динамически сохранять задачи в localStorage. Если среда имеет возможности для сохранения данных (например, Backbone.sync), используйте это. В противном случае используйте vanilla localStorage. Если возможно, используйте ключи id, title, заполненные для каждого элемента. Убедитесь, что используете этот формат для имени localStorage: todos- [framework]. Режим редактирования не должен сохраняться.

**Routing**

Маршрутизация требуется для всех реализаций. Если поддерживается фреймворком, используйте его встроенные возможности. В противном случае используйте библиотеку маршрутизации Flatiron Director, расположенную в папке / assets. Следующие маршруты должны быть реализованы: # / (все - по умолчанию), # / active и # / complete (#! / Также разрешен). Когда маршрут изменяется, список задач должен фильтроваться на уровне модели, а выбранный класс в ссылках фильтра должен переключаться. Когда элемент обновляется в отфильтрованном состоянии, он должен обновляться соответствующим образом. Например. если фильтр активен и элемент отмечен, он должен быть скрыт. Убедитесь, что активный фильтр сохраняется при перезагрузке.