# Тестовое задание Frontend

## 1 Этап (Верстка)

Сверстать часть страницы списка товаров.  
Браузеры ИЕ9+, остальные последние 2 версии.  
Мобильные android 4+, IPhone 5+.  
Сетка товаров должна подстраиваться под разрешение окна браузера.  
Верхний список товаров в макете это hover состояние.

У товаров есть следующие атрибуты title, description, cost, discountCost, new и картинка. Если значение атрибута new true то на товаре должен быть соответствующий бейдж. Бейдж sale должен быть если присутствует discoutCost, (т.е. не равен null) В файле data.php есть тестовые данные, но картинка для всех одинаковая, картинки надо поменять на картинки из макета.

## 2 Этап (JS)

Сделать дозагрузку контента по нажатию на кнопку load more.

Изначально на странице 4 товара, по нажатию на кнопку должны подгружаться еще по 4-е.

Если товаров больше нет, кнопка load more должна исчезнуть.

Новые товары не должны появляться резко, надо сделать анимацию.

У пользователя должно создаваться впечатление, что товары подгружаются моментально. Для этого необходимо реализовать предзагрузку, страница возвращается с сервера с первыми 4-я товарами, сразу после загрузки страницы делаем AJAX запрос на следующие 4-е товара, но показываем их только когда пользователь нажал на кнопку load more, и сразу загружаем следующие 4-е и т.д.

Также надо учесть ситуацию когда пользователь нажал на кнопку load more раньше чем пришел ответ на AJAX запрос, (т.е. для следующей последовательности событий: 1 - загрузка страницы, 2 - отправка AJAX запроса, 3 - нажатие кнопки load more 4 - получение ответа на AJAX запрос), в таком случае надо показать прелоадер до тех пор пока не придет ответ от сервера (внешне поведение в этом случае выглядят так как бы оно выглядело без предзагрузки). В скрипте который возвращает список товаров (**list.php**) предусмотрена задержка 2 секунды (строка sleep(2) в list.php), для удобства тестирования этого кейса ее можно увеличить.



**test.psd** - макет

**index.php** - шаблон, в нем уже как-то выводятся товары, разметку можно менять как угодно, это просто пример

**list.php** - скрипт на который можно посылать запросы для получения товаров.

Принимает два get-параметра page и per\_page - номер страницы и количество записей на странице соответственно.

Возвращает JSON с данными по товарам (entities), и общим количеством товаров (total)

**data.php** - тестовые данные по товарам

Использовать можно все, что как вам кажется больше подходит для данного конкретного случая.

От вас хотелось бы получить архив с готовой версткой, исходными файлами и количеством затраченного времени на данный шаблон.

ссылка на файлы

<https://drive.google.com/open?id=0B1pWV3o2RHmvck5adDdkdHA0T3M>