

Приглашаем Вас пройти тестовое задание в компании Inssmart (МК Групп)

Результаты необходимо отправить на e-mail e.zhuravleva@inssmart.ru с указанием Ваших контактных данных.

Вопросы по заданиям также можно направлять на почту. Все отправленные задания я просмотрю, но ответить всем не смогу из-за большого количества кандидатов.

Ориентировочные сроки выполнения: 2 дня

Задание 1

Ссылка на страницу, которую необходимо протестировать:

<https://widgets.inssmart.ru/contract/mortgage/?appId=79f46bfb-d4ba-50aa-8269-7289dd16700c&secret=6ca520c0-328b-5b65-ab5c-b41332a3e667>

Задача:

1. Составьте перечень проверок (чек лист) для первой страницы, который по Вашему мнению будет достаточным для выпуска продукта в продакшен.
2. Сгруппируйте проверки в форме иерархического списка. В этом вам поможет <https://checkvist.com/> (должен получиться полноценный чек лист, с понятной группировкой. Документ должен соответствовать вашему представлению о том, что можно использовать в реальной работе, не забывая про ожидаемый результат)
3. Укажите приоритеты проверок.
4. Откройте доступ к списку (кнопка Share)
5. Ссылку на список отправьте на почту

Условия:

Проверку необходимо сделать только для первого экрана, экран с предложениями и последующие тестировать не нужно.

Критерии оценки по приоритетам:

1. Структура и приоритезация проверок.
2. Полнота проверок
3. Правильно расставленные приоритеты

Задание 2

Коллекция в swagger, с которой предстоит работать: <https://petstore.swagger.io/>

Вам необходимо на основе этой коллекции, создать коллекцию в Postman

Нужно будет работать с запросами:

1. Создать нового питомца (**метод POST /pet**)

Прописать скрипт, который берет из тела ответа id созданного питомца и записывает в окружение.

Подсказка: скрипт прописывается во вкладке «tests», для записи переменной в окружение можно использовать `rm.environment`
Если Вы не можете решить проблему с записью `id` в окружения целиком - можете задать `id` изначально при создании питомца

2. Запросить питомца по `id` (**метод GET /pet/{petId}**) используя в запросе ранее записанную переменную из окружения
Прописать скрипт, который будет брать тело ответа, менять параметр «status» на «sold» (продан) и перенаправлять его в запрос на изменение питомца (**метод PUT /pet**)

Подсказка, скрипт прописывается во вкладке «tests», для перенаправления запроса можно использовать `rm.sendRequest`

3. Коллекцию необходимо экспортировать (Export/ Export Collection) и отправить на почту (ссылками на постман не принимаю!).

В итоге должно быть два запроса, при отправке которых должен создаваться новый питомец и он же продаваться

Задание 3

В системе есть три пользователя:

1 Иванов Иван Иванович, 1990 г.р. У него есть двое детей - дочь Маша 2010 г.р. и сын Коля 2020 г.р.

2 Смирнова Виктория, 1995 г.р. У неё есть две дочери - Катя 2015 г.р. и Настя 2020 г.р.

3 Петров Петр Петрович, 2000 г.р. У него нет детей

Какой ответ в формате json вы ожидаете получить на запрос всех пользователей?

Задание 4

Таблицы, с которыми предстоит работать:: заказы (Orders) и клиенты (Customers), найти их можно тут:

https://www.w3schools.com/sql/trysql.asp?filename=trysql_asc

Необходимо вывести `id` клиента и имя клиента, сделавших больше 5-ти заказов, а также количество заказов и дату последнего заказа. Список должен идти по убыванию от клиентов с большим количеством заказов к клиентам с меньшим количеством заказов

PS

В заданиях представлено то, с чем нужно уметь работать уже на начальных этапах.

Если Вы чего-то не знаете, то это прекрасная возможность разобраться в этом самостоятельно. В любом случае подобные навыки нужны тестировщику

Очень надеюсь, что у Вас все получится. Удачи =)