



Задания.

Задание 1.

ТЗ от заказчика: "Реализовать форму, которая по введенным данным определяет, является ли человек совершеннолетним. Приложение должно быть с архитектурой клиент-сервер".

Предположим, что разработчик реализовал данную форму.

Начальный вид формы (в соответствии с макетом):

Введите ваш Возраст?

Введите цифру

лет

При нажатии на кнопку OK появляется одно из сообщений (для вывода используется функция JS alert):

Подтвердите действие

Вы совершенно-летний

Подтвердите действие

Не совершеннолетний

Ответьте на приведенные вопросы:

1) Какие могут возникнуть вопросы к требованиям?

В каких или каком формате, должно быть отображение возраста? Допустим: полное цифровое значение - 45, кириллица, дд/мм/гг, латиница.

Будет ли сервис **интернационален**? Так как в многих странах совершеннолетие может составлять не как в РФ 18 лет.

Пример : Дания (Гренландия — 12 лет, Фарерские острова — 14 лет, континентальная часть — 18 лет)

Иран (девочки — 9 лет, мальчики — 15 лет) - тут можно ознакомиться с более широким списком.

Получается если человек из другой страны , и в той стране возраст совершеннолетия 16, а не 18 , как сайт определит человека и его совершеннолетие? С какими правами доступа, или вообще получит ли человек доступ?

Функция JS alert оставляет пользователя без выбора, Есть кнопка "ОК", а кнопка "отмена" или закрытия окна отсутствует???

2)Какие виды и типы тестирования стоит проводить? Объясните, почему.

Основополагающие виды тестирования:

Функциональное тестирования - Проверим все кнопки и возможности продукта. Соответственно заявленные в ТТХ.

Стабильности - считаю что это основа, если продукт работает не стабильно , то кто и как им будет пользоваться?

Нагрузочное - В моем понимании мы должны знать что продукт должен работать быстро, должны понимать что системные требования чем ниже тем лучше.

Юзабилити - Продукт должен быть удобен, для этого мы тут и собрались

Кросс платформа - Продукт Будет более популярен, успешен, если он будет доступен на множестве ОС платформ.

Стресс тест - разработчик должен понимать как и заказчик. Вместе они должны оценить и примерно предположить какое будет количество людей юзать продукт, и конечно подготовится через стресс тест, что бы продукт не упал в первый же день релиза.

3)Что больше подойдет для организации тестовой документации по задаче - тест-кейсы или чек-листы? Почему?

Лично я думаю можно было бы обойтись **чек-листом**, так как мы работать будем с числовым полем.

Но после ввода даты, числа возраста, есть всплывающее окно, и для максимально понятной ситуации можно применить оба варианта **Чек-Лист** и **ТестКейс** .

Где: **чек-лист** дает нам понимание что нужно для числового поля возраст, а **тест кейс** нам говорит нажми кнопку "ок" , а после еще подтверди свое действие. Что бы даже самый маленький Джун понял).

(Если это долгосрочный проект надеемся на это, то в дальнейшем Джуну будет проще)

4)Какие техники тест-дизайна Вы бы использовали? Почему?

В данной ситуации мы будем использовать технику **Граничных значений**.

Потому что нам нужно точно определить является ли посетитель продукта совершеннолетним.

На примере РФ 18 лет. от 0 до 17 не совершенно летние от 18 до 100(условно совершеннолетние)

С этой техникой нам не нужно писать тест кейсы на все года и как продукт себя должен вести.

Мы укажем что от 0 до 17лет (один контент к примеру), а от 18 и до 100 (весь контент) условно, или то ограничение которое не противоречит законодательству стран где будет распространяться данный продукт. (конечно все это мы узнаем от заказчика и его

потребности).

5) Какие можно выделить негативные и позитивные входные данные?

Позитивные данные - это числовое значение в четком формате к примеру - "45", "12", "75". четко указывают возраст пользователя. Если отталкиваться от формы и что мы видим в ней.

Негативные данные - сорок пять, 13/04/1999, 13)04)1999, 03, 348, ффывл90;№\$, пустое поле и 17,5 отталкивая от формы, я считаю что это будут негативные данные.

6) Какие теоретически могут быть баги? Опишите 4-6 возможных бага.

Мир ошибок очень удивителен и разнообразен:

Возможна ошибка где совершенно летние будут как не совершеннолетние, а не совершеннолетние как совершеннолетние.

Форма может принимать любые цифровые значения

Форма может принимать любые символы

Форма может принимать любые буквенные значения

Форма может блокировать дальнейшую регистрацию (Критический баг)

Форма может содержать орфографические ошибки (тривиальный баг) - "Вы совершенно-летний", "Введите ваш Возраст?"

Это можно продолжать долго и может даже бесконечно, потому что все баги найти невозможно.

7) Есть ли баги любого типа на приведенных скриншотах? Если есть, опишите все.

Я не на что не претендую, но как говорится: критикуешь - предлагай!

Тривиальные ошибки но есть.

1) Баг - "Введите ваш Возраст?" - Пунктуационная ошибка.

хороший продукт должен и визуально и технически быть соответствующим слову "Хороший"

Укажите ваш возраст?

Введите ваш возраст.

2) Баг - "Вы совершенно-летний" орфографическая ошибка так как:

В данном случае написание напрямую связано с тем, что перед нами, - словосочетание или одно слово.

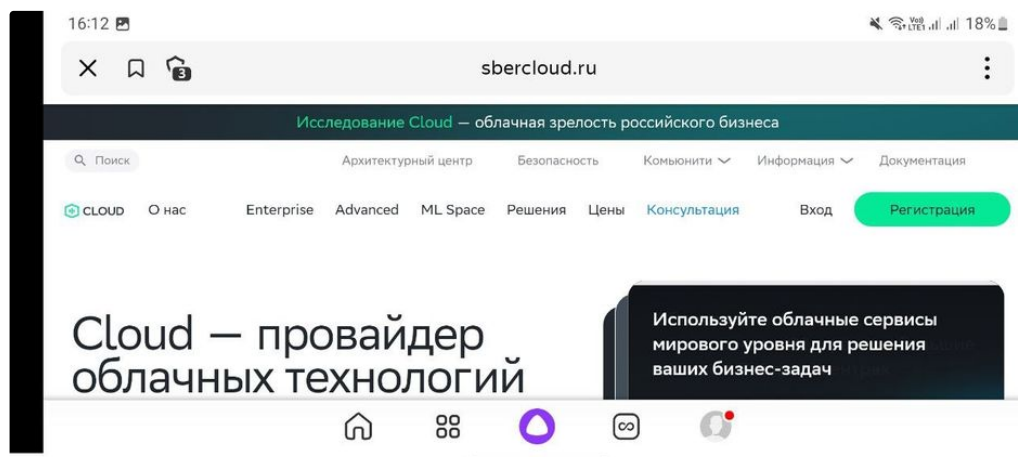
Совершенно (в какой мере, степени?) летние - **словосочетание** со связью примыкание, **пишем в два слова** (наречие+прилагательное), т.е. раздельно, напр., Наступили тёплые, совершенно летние дни.

Совершеннолетние (кто? или какие?) - **сложное слово**, в зависимости от контекста может быть и прилагательным, и субстантивированным сущ-м, пишем его **как одну лексему**, напр., Совершеннолетние (какие?) дети - люди уже взрослые/На лекцию приглашаются только совершеннолетние (кто?).

Задание 2.

Составить чек-лист проверок для [главной страницы сайта](#)

Чек лист может быть в два раза больше, но Хедер полностью дублирует всё наполнение главной страницы. Можете проверить и убедиться в этом сверив - функционал. Сам сайт не перегружен, ориентироваться легко, визуальных нарушений в отображении UI я не заметил. Каждая кнопка, фильтр, ссылка на сайте работает. Сайт работает очень быстро, Регистрация и вход защищены Паролем + Аутентификатор. Все устройства и версии браузера которые были под рукой работали быстро и четко.



16:08

VoLTE LTE1 19%

✕   sbercloud.ru



Исследование Cloud — облачная зрелость
российского бизнеса

☰  CLOUD



Регистрация

Cloud — провайдер облачных технологий и AI-решений


















Профессиональная поддержка для
реализации ваших бизнес-задач.



















Начать работу






Консультация

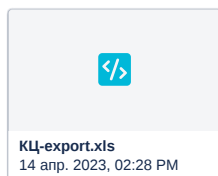
В два раза больше VDI
на три месяца



Хедер			17
НОМЕР	ПРОВЕРКА	ОЖИДАЕМЫЙ РЕЗУЛЬТАТ	
 0 sber 1	https://sbercloud.ru/ru	Сайт открывается и доступен	
 0 sber 2	F5, или повторный заход на сайт	При попытке повторно открыть сайт, он открывается и доступен	
 0 sber 3	Регистрация	Открывает меню регистрации (Регистрация проходит успешно) Так же повторно	
 0 sber 4	Вход	Открывает меню авторизации (Авторизация проходит успешно) так же повторно.	
 0 sber 5	Поиск	Выполняет поиск по ключевым словам (Принимает валидные запросы) введите к примеру IT	
 0 sber 6	Документация	Открывает страницу с описанием сервисов сайта	
 0 sber 7	Информация	Меню аккордион содержащие доп. информацию(каждый раздел исправно работает)	
 0 sber 8	Комьюнити	Меню аккордион содержащие доп. информацию (каждый раздел исправно работает)	
 0 sber 9	Безопасность	Открывает страницу с инструкциями о безопасности	
 0 sber 10	Архитектурный центр	Открывает страницу с меню о Архитектурном центре Cloud	
 0 sber 11	О нас	Открывает страницу с описанием деятельности сервиса	
 0 sber 12	Enterprise	Функциональное всплывающее меню (каждая ссылка в меню работает)	
 0 sber 13	Advanced	Функциональное всплывающее меню (каждая ссылка в меню работает)	
 0 sber 14	ML Space	Функциональное всплывающее меню (каждая ссылка в меню работает)	
 0 sber 15	Решения	Функциональное всплывающее меню каждая ссылка в меню работает)	
 0 sber 16	Цены	Открывает страницу с калькулятором сайта	
 0 sber 17	Консультация	Открывает меню с формой обратной связи	

Футер			18
НОМЕР	ПРОВЕРКА	ОЖИДАЕМЫЙ РЕЗУЛЬТАТ	
 sber 18	Техническая поддержка	Содержит – номера телефонов, меседжеры для связи, почтовые адреса.	
 sber 19	Решения	Содержит перечень услуг сервиса	
 sber 20	Документация	Содержит виды документации сервиса	
 sber 21	Платформы	Содержит виды платформ сервиса	
 sber 22	О компании	Открывает страницу с описанием деятельности сервиса	
 sber 23	Юридические документы	Открывает страницу с содержанием информации Юр. характера	
 sber 24	Партнерская программа	Открывает страницу Партнёрская программа Cloud	
 sber 25	Контакты	Открывает страницу с физическими контактами, адресами.	
 sber 26	Вакансии	Открывает страницу с требуемыми вакансиями	
 sber 27	Вебинары	Открывает страницу с Бесплатные вебинары Cloud	
 sber 28	Все продукты	Открывает страницу с перечнем продукции сервиса	
 sber 29	Блог	Открывает страницу Блог Cloud	
 sber 30	Новости	Открывает страницу Новости Cloud	
 sber 31	IaaS	Открывает страницу с описанием о IaaS	
 sber 32	PaaS	Открывает страницу с описанием о PaaS	
 sber 33	Machine Learning	Открывает страницу с описанием о Machine Learning	
 sber 34	Разработчикам	Открывает страницу – Продукты Сбера для разработчиков	
 sber 35	Ссылки на соц.сети, платформы.	открывает В контакте, Телеграмм, Ютуб, Хабр.	

Кросс платформа			
НОМЕР	ПРОВЕРКА	ОЖИДАЕМЫЙ РЕЗУЛЬТАТ	
 sber 36	windows 11 pro , версия 22H2	Все работает.	
 sber 37	Браузер Opera, Версия Chromium:111.0.5563.147	Все работает	
 sber 38	Браузер Microsoft Edge Версия 112.0.1722.39 (Официальная сборка) (64-разрядная версия)	Все работает	
 sber 39	Google Chrome, Версия 112.0.5615.87 (Официальная сборка), (64 бит)	Все работает	
 sber 40	Мобильное Устройство андроид версия 13 Браузер яндекс Версия 23.32 сборка 230302003	Все Работает	



Экспортная версия из ситичко. xls.

Задание 3.

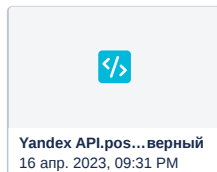
API Яндекс.Погоды позволяет получать погодные данные в автоматизированном режиме. Объем предоставляемых данных зависит от [выбранного тарифа доступа](#)

Необходимо с помощью Postman протестировать работу метода со следующими параметрами:

- тариф Тестовый
- lon 50
- lat - 55
- язык русский

Позитивный запрос.

https://api.weather.yandex.ru/v2/forecast?lat=-55&lon=50&lang=ru_RU

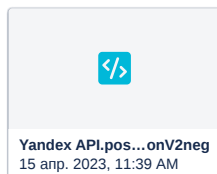


Импорт позитивного запроса.

Также создать несколько негативных сценариев и убедиться, что при неправильных данных запрос возвращает соответствующий результат

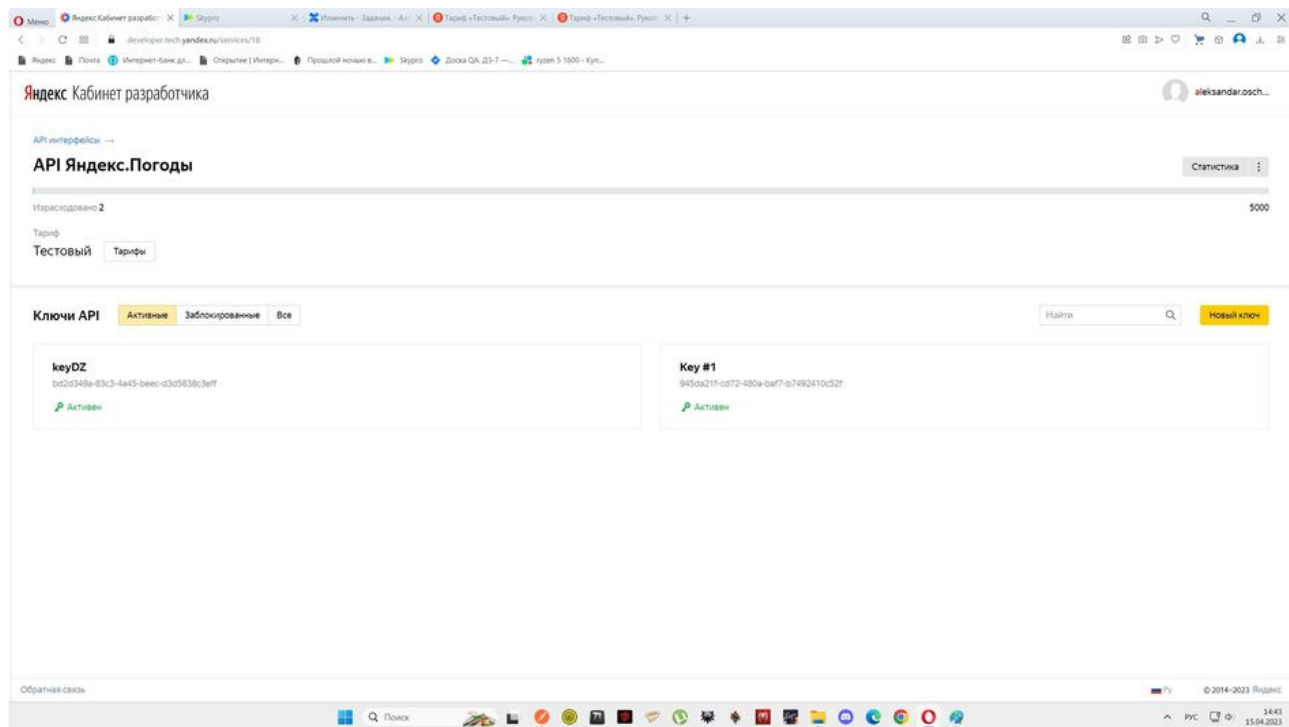
Негативный запрос.

Get: https://api.weather.yandex.ru/v2/forecast?lat=50&lon=555&lang=ru_RU так выглядит негативный запрос. Запрос можно ломать по разному в разных настройках. Думаю 1 примера достаточно что бы была ясность что я это понимаю.



Импорт негативного запроса.

При выполнении задания была ситуация. Все сделал как в 1 позитивном запросе но выпадала ошибка **403 Forbidden**. Используя Гугл, нашел информацию что ключ может работать не сразу, а через сутки. Я не знаю правда ли это или нет. В моем случае, через час примерно, стало работать. После выполнения задания, я создал второй ключ, который работал сразу.



Задание 4. [↗](#)

Есть вот такие вводные данные:

table animal_classes	
id	class
1	кошка
2	собака

table animal_info		
name	class	owner
Кити	1	Ваня
Мити	2	Ваня
Пити	1	Петя

Ответь, используя запросы SQL, на следующие вопросы:

1. сколько всего животных у Вани

```
Select count (*)  
from (select owner  
from animal_classes inner join animal_info on animal_classes.class=animal_info.class  
where animal_classes owner = 'Ваня'  
group by owner);
```

2. имена всех кошек

```
select animal_info. name, animal_classes. class,  
from animal_classes inner join animal_info on animal_info.class=animal_classes.id  
where id = 1;
```

3. Уникальные имена всех хозяев, отсортированные по алфавиту

```
select DISTINCT owner  
from animal_info  
order by owner;
```