

Курс:  
**«Язык сценариев JavaScript и библиотека jQuery»**

В качестве экзаменационной работы необходимо выполнить одно из следующих двух заданий.

## Задание 1

Необходимо создать одностраничный сайт с прогнозом погоды. Для получения прогноза используйте <https://openweathermap.org/>. Не забудьте зарегистрироваться и получить ключ.

На странице должно быть 2 вкладки:

- **Today** – прогноз погоды на сегодня;
- **5-day forecast** – прогноз погоды на 5 дней вперед.

При загрузке страницы отображается вкладка **Today**. Текущий город определяется по координатам пользователя, а если браузер не поддерживает геолокацию, то отображается город, в котором вы живете. Для выбора другого города пользователь может ввести его название в текстовое поле для поиска.

На вкладке **Today** отображается 3 блока.

### 1. Краткие сведения о текущей погоде:

- дата;
- иконка;
- текстовое описание;
- температура;
- как ощущается температура;
- рассвет;
- закат;
- длительность дня.

### 2. Почасовой прогноз на остаток дня:

- время;
- иконка;
- текстовое описание;
- температура;
- как чувствуется температура;
- скорость и направление ветра.

### 3. Ближайшие города и прогноз для них:

- название;
- иконка;
- температура.

В текстовом поле для поиска всегда должно отображаться название города, по которому выводится прогноз погоды. Даже в том случае, если город определен по геопозиции.

**MY WEATHER**


Baku, Azerbaijan

Today

5-day forecast

CURRENT WEATHER

30.06.2018



Sunny

29°C







Real Feel 30°

Sunrise: 7:04 AM





Sunset: 5:11 PM

Duration: 10:07 hr

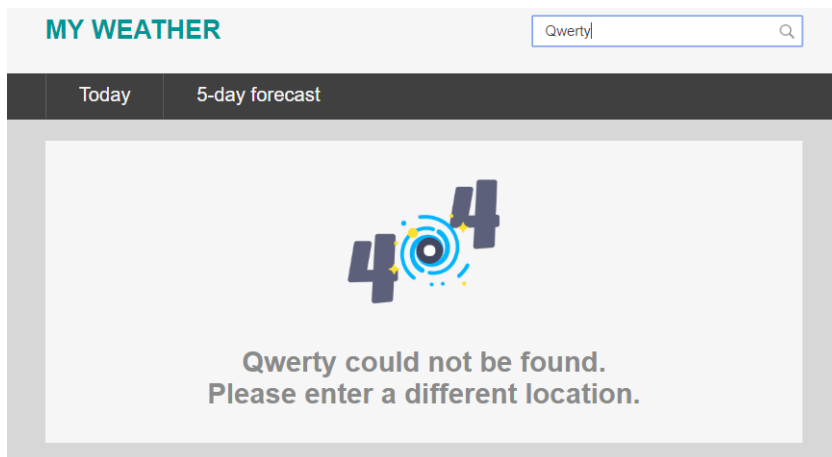
HOURLY

TODAY	7pm	8pm	9pm	10pm	11pm	12am
						
Forecast	Sunny	Sunny	Clear	Clear	Clear	Clear
Temp (°C)	29°	28°	28°	27°	27°	27°
RealFeel	28°	27°	28°	27°	27°	26°
Wind (km/h)	11 ESE	9 ESE	6 SE	6 ESE	6 ESE	7 SE

NEARBY PLACES

Baladzhy		36°C	Bailov		29°C
Badamdar		36°C	Cherni Gorod		29°C

Если пользователь ввел несуществующий город или API не может вернуть вам информацию по введенному городу, то информируем об этом с помощью такой страницы:



На вкладке **5-day forecast** отображается 2 блока.

**1. Краткий прогноз для каждого из пяти дней:**

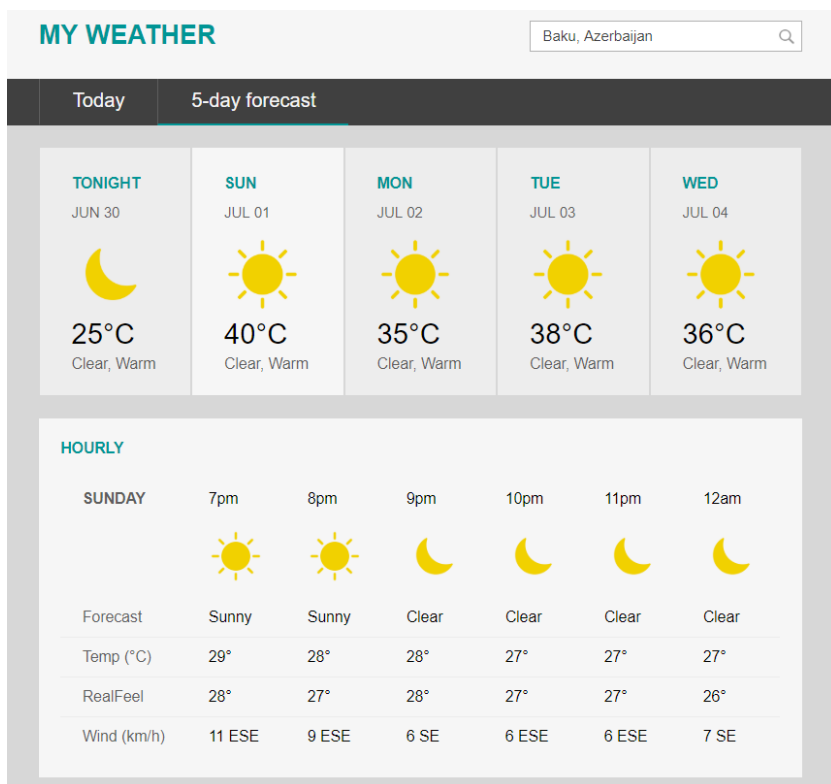
- день недели;
- дата;
- иконка;
- температура;
- текстовое описание.

**2. Почасовой прогноз для выбранного дня:**

- время;
- иконка;
- текстовое описание;
- температура;
- как чувствуется температура;
- скорость и направление ветра.

При клике на блок краткого прогноза одного из пяти дней, необходимо его визуально выделить и ниже отобразить почасовой прогноз на этот день.

При открытии этой вкладки по умолчанию выбранным должен быть сегодняшний день.



**Обращаем ваше внимание** на то, что сайт одностраничный. Это значит, что при любых кликах, при поиске прогноза для другого города или при переходе по вкладкам, не (!) должна обновляться страница в браузере. Вам необходимо изменять структуру страницы с помощью JavaScript.

## Задание 2

**Необходимо создать сайт для библиотекаря, с помощью которого он может вести учет книг.**

**Сайт должен состоять из 4 разделов (пунктов меню).**

### **1. Работа с книгами.**

На этой странице можно:

- посмотреть список всех книг;
- добавить новую книгу;
- изменить существующую книгу;
- удалить существующую книгу.

При просмотре списка всех книг можно осуществлять сортировку и поиск. Сортировать можно по названию, по имени автора или по количеству экземпляров в библиотеке. Для поиска пользователь вводит часть слова, и проверка происходит по наличию этой части в названии книги или в имени автора или в названии издательства.

При добавлении или изменении книги необходимо валидировать данные: все поля обязательны для заполнения и в числовых полях не может быть отрицательных значений.

### **2. Работа с посетителями.**

На этой странице можно:

- посмотреть список всех посетителей;
- добавить нового посетителя;
- изменить существующего посетителя.

При просмотре списка всех посетителей можно осуществлять сортировку и поиск. Сортировать можно по идентификатору или по имени. Для поиска пользователь вводит часть слова, и проверка происходит по наличию этой части в имени посетителя или в номере телефона.

При добавлении или изменении посетителей необходимо валидировать данные: все поля обязательны для заполнения, а в номере телефона допустимы только цифры, пробел и тире.

### 3. Работа с карточками.

**Карточка** – это одна взятая книга одним посетителем, с датой выдачи и датой возврата. Дата возврата появляется только тогда, когда вернули книгу.

На этой странице можно:

- выдать книгу (создать новую карточку);
- забрать книгу (установить дату возврата).

При выдаче книги количество ее экземпляров в библиотеке должно уменьшаться на 1. Соответственно, при возврате – увеличиваться на 1.

При создании карточки в выпадающем списке с книгами должны отображаться только те книги, которые есть в библиотеке на данный момент (то есть количество экземпляров больше 0).

### 4. Статистика.

На этой странице можно:

- посмотреть список из 5 самых популярных книг;
- посмотреть список из 5 самых активных посетителей.

**Основные сущности, которые понадобятся при разработке сайта** (возможно понадобится что-то еще, это минимальный список).

Книга:

- идентификатор;
- название;
- имя автора;
- год издательства;
- название издательства;
- количество страниц;
- количество экземпляров в библиотеке.

Посетитель:

- идентификатор;
- ФИО;
- телефон.

Карточка посетителя:

- идентификатор;
- идентификатор посетителя;
- идентификатор книги;
- дата, когда взял книгу;
- дата, когда вернул книгу.

Данные необходимо хранить в `localStorage` или в `sessionStorage`.

На следующих изображениях показан возможный интерфейс разделов для работы с посетителями и для работы с карточками.



## Список посетителей, их сортировка и поиск

LIBRARY

Books

Visitors

Cards

Statistics

ALL VISITORS:











New visitor

Sort by: ID

Sort

Search:

Search

ID	Name	Phone	Edit
1	Colette Kelley	012 435 45 67	
2	Ruby-Rose Lennon	012 647 34 24	
3	Leanne Gibbons	012 879 78 45	
4	Rumaisa Peel	012 456 64 67	
5	Gene Medrano	012 245 47 89	
6	Sheridan Tucker	012 345 85 90	
7	Christina Mack	012 123 36 46	
8	Everly Moses	012 678 99 74	
9	Kara Feeney	012 456 96 53	
10	Cameron Rennie	012 967 47 85	

## Редактирование посетителя

(такая же форма используется для создания нового посетителя)

**LIBRARY**

Books Visitors Cards Statistics

ALL VISITORS: New visitor

Sort by: ID Sort Search: Search

ID			Edit
1			
2			
3			
4			
5	Gene Medrano	012 245 47 69	
6	Sheridan Tucker	012 345 85 90	
7	Christina Mack	012 123 36 46	
8	Everly Moses	012 678 99 74	
9	Kara Feeney	012 456 96 53	
10	Cameron Rennie	012 967 47 85	

**Edit visitor:**

Full name:

Phone number:

Save

## Список карточек, их сортировка и поиск

В колонке **Return Date** отображается дата возвращения книги или кнопка (стрелочка). По нажатию на кнопку происходит возвращение книги в библиотеку, то есть в колонку **Return Date** устанавливается текущая дата.

## LIBRARY

Books

Visitors

Cards

Statistics

ALL CARDS:

New card

Sort by: Return date ▼

Sort

Search:

Search

ID	Visitor	Book	Borrow Date	Return Date
1	Colette Kelley	JS Part 1	12.01.2019	12.03.2019
2	Ruby-Rose Lennon	JS Part 2	12.01.2019	↩
3	Leanne Gibbons	HTML+CSS book	12.01.2019	↩
4	Rumaisa Peel	Bootstrap	12.01.2019	12.03.2019
5	Gene Medrano	SQL Part 1	12.01.2019	12.03.2019
6	Sheridan Tucker	SQL Part 2	12.01.2019	12.03.2019
7	Christina Mack	SQL Part 3	12.01.2019	12.03.2019
8	Everly Moses	PHP for beginner	12.01.2019	↩
9	Kara Feeney	C# for beginner	12.01.2019	↩
10	Cameron Rennie	C++ for beginner	12.01.2019	↩

## Создание новой карточки

Для создания новой карточки необходимо выбрать посетителя и книгу из выпадающих списков, а в дату выдачи устанавливается текущее время.

The screenshot shows a web application interface for a library. At the top, there's a navigation bar with tabs: 'Books', 'Visitors', 'Cards', and 'Statistics'. The 'Cards' tab is active. Below the navigation bar, there's a section titled 'ALL CARDS:' with a 'New card' button. A 'Sort by:' dropdown menu is set to 'Relevance'. A 'Search' button is also present. A modal dialog box titled 'New card:' is open in the center. It contains two dropdown menus: 'Visitor:' and 'Book:'. Below these is a 'Save' button. In the background, a table of library cards is visible. The table has columns for ID, Name, Book Title, Issue Date, and Return Date. The first four rows are partially obscured by the dialog box. The last six rows are visible and contain data.

ID	Name	Book Title	Issue Date	Return Date
1	Cc			12.03.2019
2	Ru			12.03.2019
3	Le			12.03.2019
4	Ru			12.03.2019
5	Gene Medrano	SQL Part 1	12.01.2019	12.03.2019
6	Sheridan Tucker	SQL Part 2	12.01.2019	12.03.2019
7	Christina Mack	SQL Part 3	12.01.2019	12.03.2019
8	Everly Moses	PHP for beginner	12.01.2019	12.03.2019
9	Kara Feeney	C# for beginner	12.01.2019	12.03.2019
10	Cameron Rennie	C++ for beginner	12.01.2019	12.03.2019