



Kammavari Sangham (R) 1952, K.S.Group of Institutions

K. S INSTITUTE OF TECHNOLOGY

9 No.14, Raghuvanahalli, Kanakapura Road, Bengaluru - 560109

Affiliated to VTU, Belagavi & Approved by AICTE, New Delhi, Accredited by NBA , NAAC & IEI

Department of Computer Science and Engineering **Project Phase – II (17CSP85)** **Review – 1**

Chrome Extension for Dropdown Code Editor

Group No.: G1

Batch No.: 25

1KS17CS012 ARITRA RAY

1KS17CS073 SAURAV SUMAN

1KS17CS080 SOUVIK MANDAL

Guided By:
Mrs. Geetha R.
Assistant Professor

Contents

- Introduction
- Comparison with similar work
- Problem Statement and Objectives
- Methodology Proposed/ Design
- Technologies / Tools Used
- Implementation of Modules with codes
- Snapshots
- References

Introduction

- Extensions are small bits of software that allow you to customize the look and feel of your browser. HTML, CSS, and JavaScript can be used to customize Chrome's functionality and behaviour. While any application or software can create timetables, to-do lists, and set notifications, Chrome extensions make it easy to keep organized while browsing the web.
- **This project “Chrome Extension for online code editor” will provide the features of an online code editor as an chrome extension.**

Comparison with similar work

- ☐ Our extension will provide all the same features as an online IDE.
- ☐ No need to open websites every time to write code.
- ☐ Our extension will be easier to access compared to other online IDEs.

Problem Statement and Objectives

1. Project Goals

This project will help developers to check for outputs for a given code without opening an external IDE or website. They could tap on the icon of this extension and paste the code in the dropdown that will appear after and get the desired output without any hassle.

It can also identify the language in which the code is being written. Keywords get highlighted to help developers analyze and code easily.

2. Project Objectives

- To fetch data easily from the server.
- To make it accessible for the user.
- Make it easy to execute through the extension.

Methodology Proposed/ Design

1. About Extension

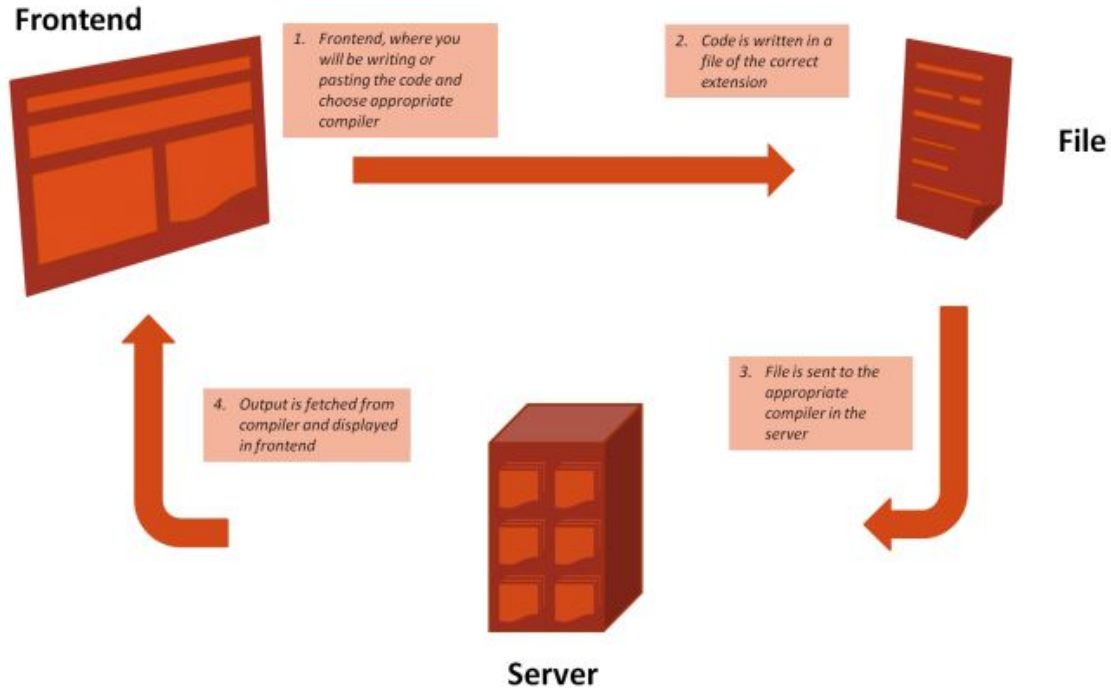
Chrome Extension for Dropdown Code Editor is a browser-based tool for writing and executing codes for multiple programming languages. In this IDE extension, we can select any specific programming language from various languages and write and execute codes of that particular programming language. To develop our IDE we used HTML, CSS, JavaScript for the front end/UI, and for the server-side, we used PHP. Every extension that uses the Web Extension APIs must include the manifest.json file. One can describe basic metadata about their extension, such as the name and version, as well as features of the extension's functionality, using manifest.json (such as background scripts, content scripts, and browser actions).

2. Manifest V3

Since its inception a decade ago, Manifest V3 has represented one of the most significant changes to the extensions platform. Extensions that use Manifest V3 will benefit from improved security, privacy, and performance, as well as the ability to employ more modern Open Web technologies like service workers and promises. Developers can upgrade their extensions now to take advantage of these Manifest V3 features; as we phase out Manifest V2 in the future, this will become essential.

3. Our Extension

The user can access the extension IDE by clicking the extension icon present at the top bar of the web browser which will drop down a text area where the user can write or paste their code. They are allowed to choose a specific programming language of their choice from the menu beside the text area. According to the chosen programming language, the user can execute their code by clicking the submit button at the bottom of the editor. The output will be fetched from the server after compiling without a hassle. The user can perceive the result which will be displayed at the bottom of the editor.



Working Model

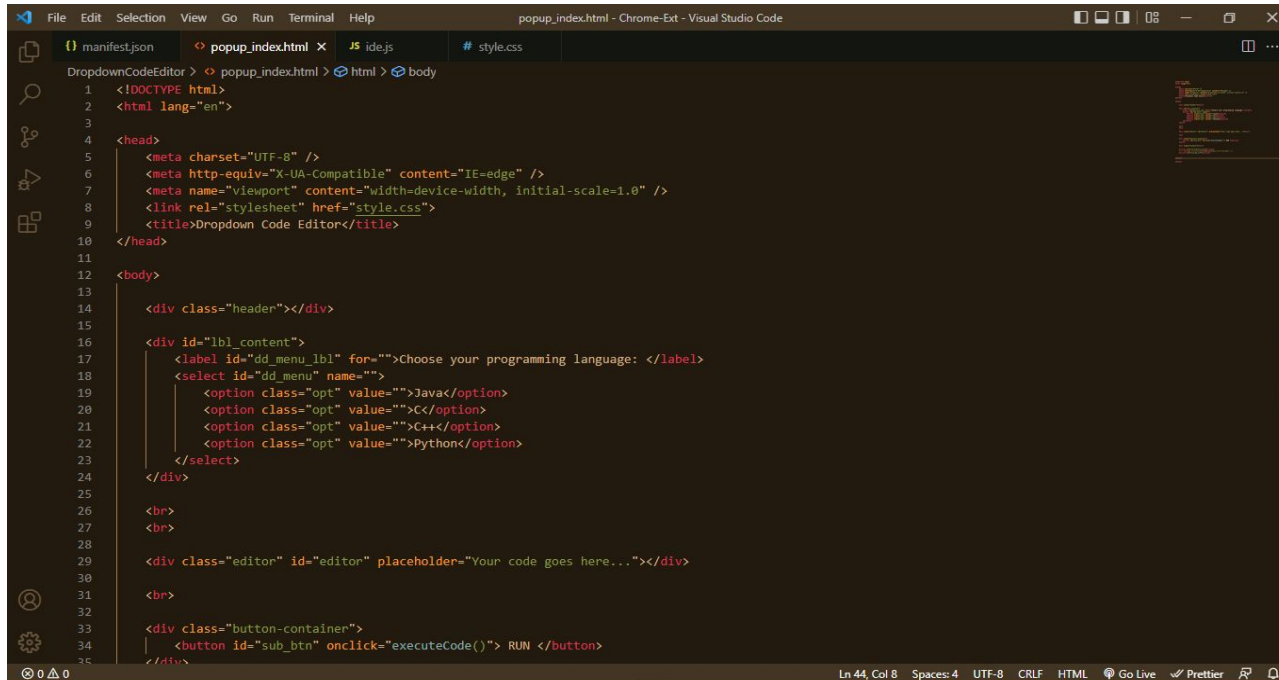
Technologies / Tools Used

- ☐ HTML
- ☐ CSS3 + Bootstrap
- ☐ JavaScript
- ☐ Manifest.JSON
- ☐ Ace Editor

Implementation of Modules with codes

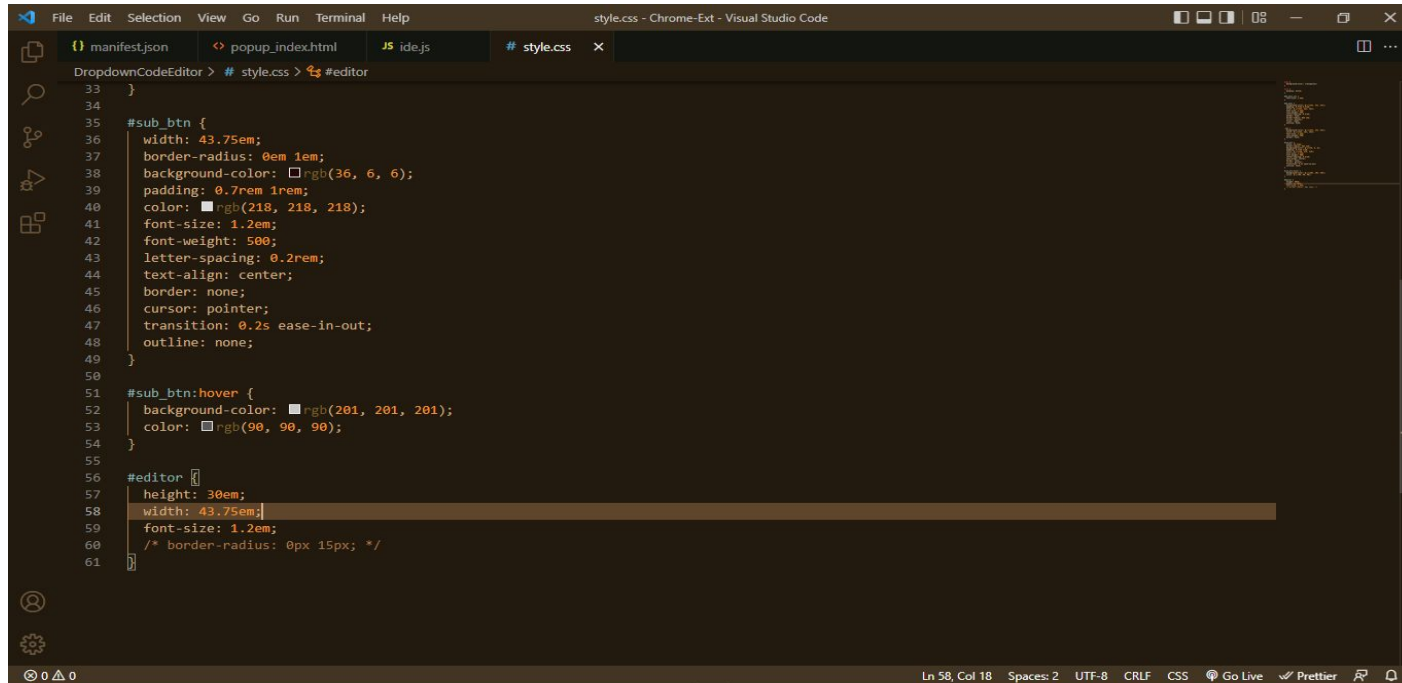
Frontend

- popup_index.html



```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8" />
6   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
7   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
8   <link rel="stylesheet" href="style.css">
9   <title>Dropdown Code Editor</title>
10 </head>
11
12 <body>
13
14   <div class="header"></div>
15
16   <div id="lbl_content">
17     <label id="dd_menu_lbl" for="">Choose your programming language: </label>
18     <select id="dd_menu" name="">
19       <option class="opt" value="">Java</option>
20       <option class="opt" value=""></option>
21       <option class="opt" value="">C++</option>
22       <option class="opt" value="">Python</option>
23     </select>
24   </div>
25
26   <br>
27   <br>
28
29   <div class="editor" id="editor" placeholder="Your code goes here..."></div>
30
31   <br>
32
33   <div class="button-container">
34     <button id="sub_btn" onclick="executeCode()"> RUN </button>
35   </div>
```

- style.css



The screenshot shows the Visual Studio Code editor with the 'style.css' file open. The editor has a dark theme. The file explorer on the left shows 'manifest.json', 'popup_index.html', 'JS ide.js', and 'style.css'. The main editor area displays the following CSS code:

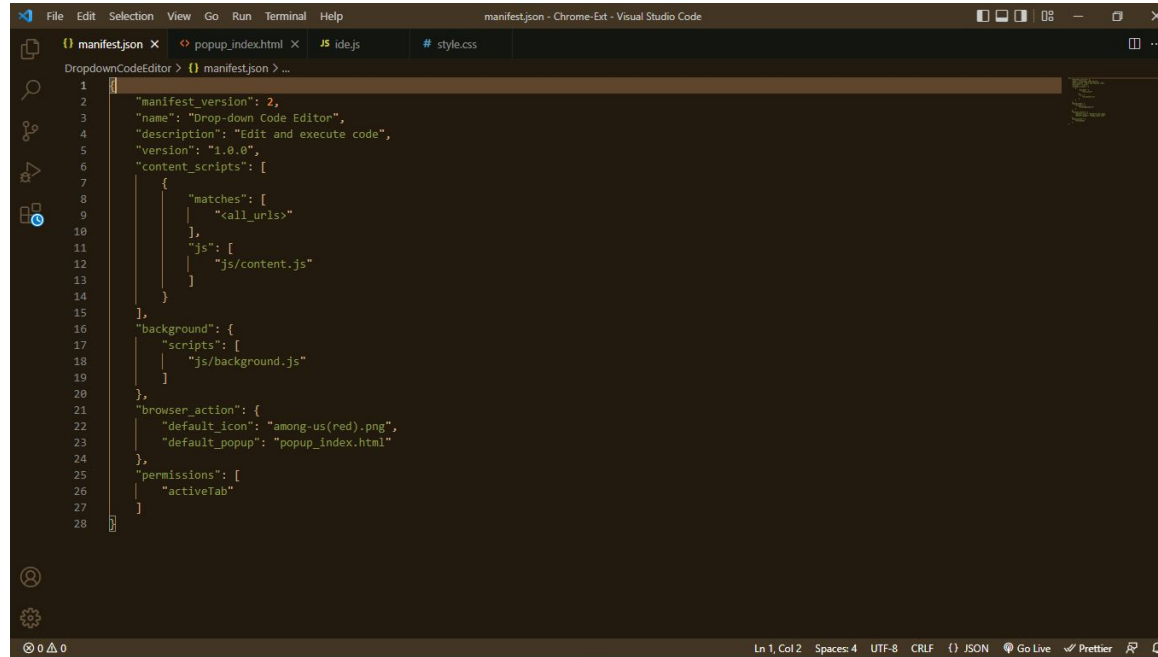
```
33 }
34
35 #sub_btn {
36   width: 43.75em;
37   border-radius: 0em 1em;
38   background-color: rgb(36, 6, 6);
39   padding: 0.7rem 1rem;
40   color: rgb(218, 218, 218);
41   font-size: 1.2em;
42   font-weight: 500;
43   letter-spacing: 0.2rem;
44   text-align: center;
45   border: none;
46   cursor: pointer;
47   transition: 0.2s ease-in-out;
48   outline: none;
49 }
50
51 #sub_btn:hover {
52   background-color: rgb(201, 201, 201);
53   color: rgb(90, 90, 90);
54 }
55
56 #editor {
57   height: 30em;
58   width: 43.75em;
59   font-size: 1.2em;
60   /* border-radius: 0px 15px; */
61 }
```

The status bar at the bottom indicates 'Ln 58, Col 18', 'Spaces: 2', 'UTF-8', 'CRLF', 'CSS', 'Go Live', 'Prettier', and a search icon.

Implementation of Modules with codes

Conversion into web extension

- manifest.json

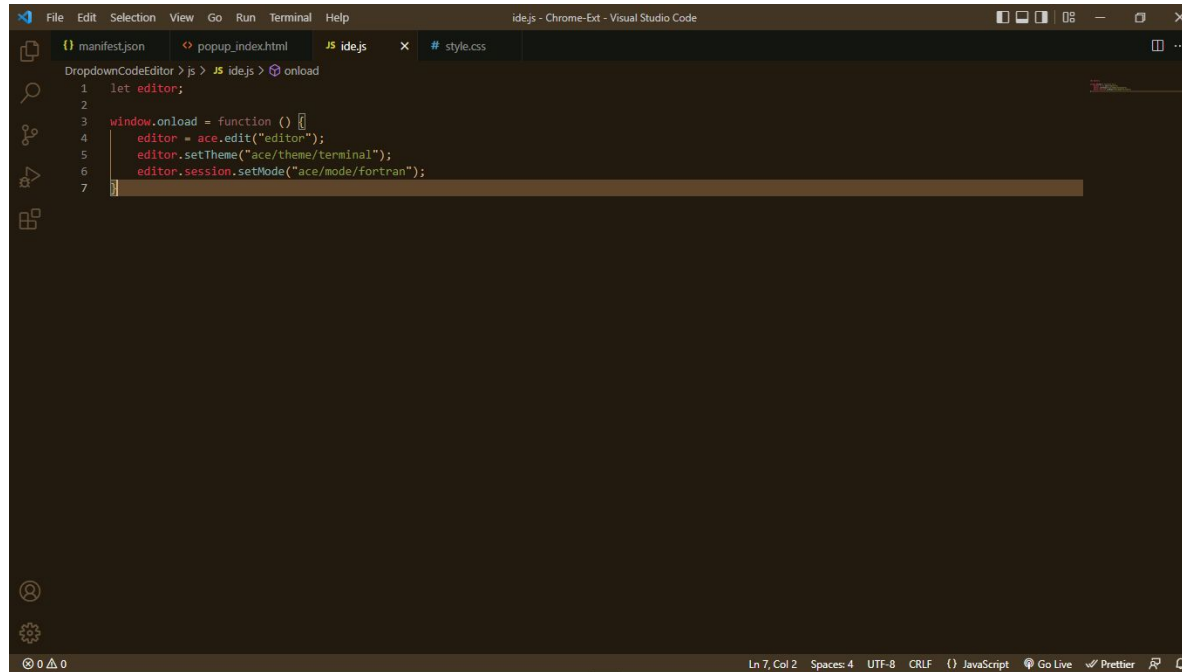


```
1  {
2    "manifest_version": 2,
3    "name": "Drop-down Code Editor",
4    "description": "Edit and execute code",
5    "version": "1.0.0",
6    "content_scripts": [
7      {
8        "matches": [
9          "<all_urls>"
10         ],
11        "js": [
12          "js/content.js"
13        ]
14      }
15    ],
16    "background": {
17      "scripts": [
18        "js/background.js"
19      ]
20    },
21    "browser_action": {
22      "default_icon": "among-us(red).png",
23      "default_popup": "popup_index.html"
24    },
25    "permissions": [
26      "activeTab"
27    ]
28  }
```

Implementation of Modules with codes

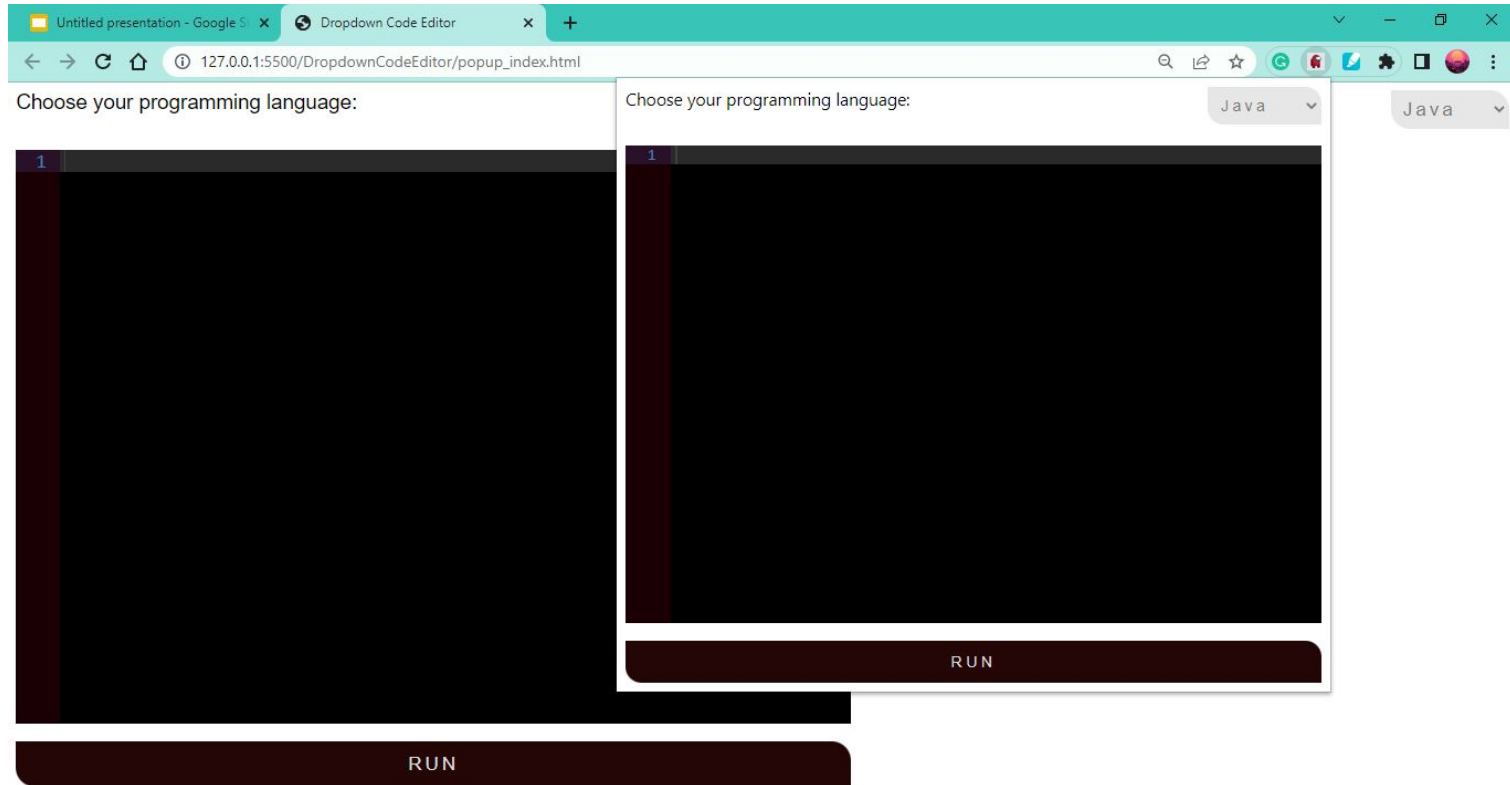
Editor theme

- ide.js



```
File Edit Selection View Go Run Terminal Help
ide.js - Chrome-Ext - Visual Studio Code
manifest.json popup_index.html JS ide.js x # style.css
DropdownCodeEditor > js > JS ide.js > onload
1 let editor;
2
3 window.onload = function () {
4   editor = ace.edit("editor");
5   editor.setTheme("ace/theme/terminal");
6   editor.session.setMode("ace/mode/fortran");
7 }
```

Snapshots



Choose your programming language:

1

Choose your programming language:

Java

```
1 interface Animal {
2     public void animalSound(); // interface method (does not have a body)
3     public void sleep(); // interface method (does not have a body)
4 }
5
6 class Pig implements Animal {
7     public void animalSound() {
8         System.out.println("The pig says: wee wee");
9     }
10    public void sleep() {
11        System.out.println("Zzz");
12    }
13 }
14
15 class Main {
16     public static void main(String[] args) {
17         Pig myPig = new Pig();
18         myPig.animalSound();
19         myPig.sleep();
20     }
21 }
22
```

RUN

RUN

References

Journal Papers:

- [1] Muhammad Nomani Kabir, Omar Tayan, Yasser Alginahi, “On the development of a web extension for text authentication on Google Chrome”, 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), 7-9 February 2019.

Website:

- https://www.w3schools.com/js/js_json_php.asp
- <https://ace.c9.io/>