

## W14-P1: Get Cocktail data by first letter

The screenshot displays a web application running in a browser, with the source code visible in the background. The application is designed to fetch cocktail data from an API and display it, filtered by the first letter of the cocktail name.

**Code Snippets:**

```
const btn = document.querySelector('.btn');

btn.addEventListener('click', async () => {
  const url = 'https://www.thecocktaildb.com/api/json/v1/1/search.php';
  try {
    const response = await fetch(url);
    const data = await response.json();
    console.log('data', data.drinks);
    displayItems(data.drinks);
  } catch (error) {
    console.error(error);
  }
});

const displayItems = (items) => {
  const displayData = items
    .map((item) => {
      const { strDrink } = item;
      return `<p>${strDrink}</p>`;
    })
    .join('');
  const element = document.createElement('div');
  element.innerHTML = displayData;
  document.body.appendChild(element);
};
```

**Browser Output:**

The browser shows a list of cocktails starting with 'A':

- A1
- ABC
- Ace
- ACID
- Adam
- AT&T
- A. J.
- Avalon
- Apello
- Affair
- Abilene
- Almeria
- Addison
- Applear
- Acapulco
- Affinity
- Aviation
- After sex
- Applejack
- Afterglow
- Afternoon
- Alexander
- Autodafé
- Allegheny
- Americano

The console shows the API response, which is an array of objects. Each object contains the cocktail name (strDrink) and other details like the category (strCategory) and tags (strTags).

# W14-P2: Get data from MealDB, SportDB, AudioDB one by one as in P1

## MealDB

The screenshot shows a web application running in a browser. The application has a button labeled "show json". When clicked, it displays a list of meals from the MealDB API. The list contains four items: "Apple Frangipan Tart", "Apple & Blackberry Crumble", "Apam balik", and "Ayam Percik".

The browser's developer console shows the following JSON response from the API:

```
{
  "data": [
    {
      "idMeal": "52708",
      "strMeal": "Apple Frangipan Tart",
      "strCategory": "Dessert",
      "strTags": "Tart",
      "strSource": "https://www.themealdb.com/recipes/52708"
    },
    {
      "idMeal": "52893",
      "strMeal": "Apple & Blackberry Crumble",
      "strCategory": "Dessert",
      "strTags": "Crumble",
      "strSource": "https://www.themealdb.com/recipes/52893"
    },
    {
      "idMeal": "53049",
      "strMeal": "Apam balik",
      "strCategory": "Dessert",
      "strTags": "Apam",
      "strSource": "https://www.themealdb.com/recipes/53049"
    },
    {
      "idMeal": "53050",
      "strMeal": "Ayam Percik",
      "strCategory": "Chicken",
      "strTags": "Ayam",
      "strSource": "https://www.themealdb.com/recipes/53050"
    }
  ]
}
```

# SportsDB

app\_40.js - 1112-IN-js-demo-211410740 - Visual Studio Code

styles\_40.css U index.html U app\_40.js X 2-supabase\_40.js

demo > w14-api-more > app\_40.js > displayItems > displayData > items.map() callback

```
1 const btn = document.querySelector('.btn');
2
3 btn.addEventListener('click', async () => {
4   const url = 'https://www.thesportsdb.com/api/v1/json/3/eventsseason.php?';
5
6   const response = await fetch(url);
7   const data = await response.json();
8   console.log('data', data.events);
9   displayItems(data.events);
10 } catch (error) {
11   console.error(error);
12 });
13
14
15 const displayItems = (items) => {
16   const displayData = items
17     .map((item) => {
18     const { strEvent } = item;
19     return `<p>${strEvent}</p>`;
20   })
21   .join('');
22   const element = document.createElement('div');
23   element.innerHTML = displayData;
24   document.body.appendChild(element);
25 };
26
```

問題 輸出 偵錯主控台 終端機 JUPYTER

yumen@LAPTOP-N0RM7KF0 MINGW64 /d/1112-IN-js-demo-211410740 (main)

\$

bash

JavaScript Tutorial

127.0.0.1:5502/demo/w14-api-more/ind 80%

AJAX

show json

Oakland Athletics vs Seattle Mariners  
Oakland Athletics vs Seattle Mariners  
Seattle Mariners vs Boston Red Sox  
Los Angeles Dodgers vs Los Angeles Angels  
New York Yankees vs Baltimore Orioles  
Washington Nationals vs New York Mets  
Milwaukee Brewers vs St. Louis Cardinals  
Philadelphia Phillies vs Atlanta Braves  
Toronto Blue Jays vs Detroit Tigers  
Tampa Bay Rays vs Houston Astros  
Oakland Athletics vs Los Angeles Angels  
Cincinnati Reds vs Pittsburgh Pirates  
Los Angeles Dodgers vs Arizona Diamondbacks  
Miami Marlins vs Colorado Rockies  
Minnesota Twins vs Cleveland Indians  
San Diego Padres vs San Francisco Giants  
Kansas City Royals vs Chicago White Sox  
Texas Rangers vs Chicago Cubs  
Tampa Bay Rays vs Houston Astros  
Seattle Mariners vs Boston Red Sox  
Toronto Blue Jays vs Detroit Tigers

Inspector Console

Array(100) [ (-), (-), (-), (-), (-), (-), (-), (-), (-), (-), ... ]

Object { idEvent: "598443", strEvent: "Oakland Athletics vs Seattle Mariners", strEventAlternate: "Seattle Mariners @ Oakland Athletics", ... }

1: Object { idEvent: "598443", strEvent: "Oakland Athletics vs Seattle Mariners", strEventAlternate: "Seattle Mariners @ Oakland Athletics", ... }

2: Object { idEvent: "618875", strEvent: "Seattle Mariners vs Boston Red Sox", strEventAlternate: "Boston Red Sox @ Seattle Mariners", ... }

3: Object { idEvent: "595947", strEvent: "Los Angeles Dodgers vs Los Angeles Angels", strEventAlternate: "Los Angeles Angels @ Los Angeles Dodgers", ... }

4: Object { idEvent: "595948", strEvent: "New York Yankees vs Baltimore Orioles", strEventAlternate: "Baltimore Orioles @ New York Yankees", ... }

5: Object { idEvent: "595949", strEvent: "Washington Nationals vs New York Mets", strEventAlternate: "New York Mets @ Washington Nationals", ... }

6: Object { idEvent: "595950", strEvent: "Milwaukee Brewers vs St. Louis Cardinals", strEventAlternate: "St. Louis Cardinals @ Milwaukee Brewers", ... }

7: Object { idEvent: "595951", strEvent: "Philadelphia Phillies vs Atlanta Braves", strEventAlternate: "Atlanta Braves @ Philadelphia Phillies", ... }

8: Object { idEvent: "595952", strEvent: "Toronto Blue Jays vs Detroit Tigers", strEventAlternate: "Detroit Tigers @ Toronto Blue Jays", ... }

9: Object { idEvent: "595953", strEvent: "Tampa Bay Rays vs Houston Astros", strEventAlternate: "Houston Astros @ Tampa Bay Rays", ... }

10: Object { idEvent: "595954", strEvent: "Oakland Athletics vs Los Angeles Angels", strEventAlternate: "Los Angeles Angels @ Oakland Athletics", ... }

11: Object { idEvent: "595955", strEvent: "Cincinnati Reds vs Pittsburgh Pirates", strEventAlternate: "Pittsburgh Pirates @ Cincinnati Reds", ... }

12: Object { idEvent: "595956", strEvent: "Los Angeles Dodgers vs Arizona Diamondbacks", strEventAlternate: "Arizona Diamondbacks @ Los Angeles Dodgers", ... }

13: Object { idEvent: "595957", strEvent: "Miami Marlins vs Colorado Rockies", strEventAlternate: "Colorado Rockies @ Miami Marlins", ... }

# AudioDB

app\_40.js - 1112-IN-js-demo-211410740 - Visual Studio Code

styles\_40.css U index.html U app\_40.js X 2-supabase\_40.js

demo > w14-api-more > app\_40.js > btn.addEventListener('click') callback

```
1 const btn = document.querySelector('.btn');
2
3 btn.addEventListener('click', async () => {
4   const url = 'https://www.theaudiodb.com/api/v1/json/2/search.php?s=coldp';
5
6   const response = await fetch(url);
7   const data = await response.json();
8   console.log('data', data.artists);
9   displayItems(data.artists);
10 } catch (error) {
11   console.error(error);
12 });
13
14
15 const displayItems = (items) => {
16   const displayData = items
17     .map((item) => {
18     const { strArtist } = item;
19     return `<p>${strArtist}</p>`;
20   })
21   .join('');
22   const element = document.createElement('div');
23   element.innerHTML = displayData;
24   document.body.appendChild(element);
25 };
26
```

問題 輸出 偵錯主控台 終端機 JUPYTER

yumen@LAPTOP-N0RM7KF0 MINGW64 /d/1112-IN-js-demo-211410740 (main)

\$

bash

JavaScript Tutorial

127.0.0.1:5502/demo/w14-api-more/ind 80%

AJAX

show json

Coldplay

Inspector Console

data

Array [ (-) ]

Artist: "111239" strArtist: "Coldplay"

strLabel: "Parlophone", ... }

length: 1

<prototype>: Array []

09ee368 yuwen1213 Thu May 18 19:52:51 2023 +0800 W14-P2: Get data from MealDB, SportDB, AudioDB one by one

## All logs

```
$ git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2023-05-17"
```

09ee368 yuwen1213 Thu May 18 19:52:51 2023 +0800 W14-P2: Get data from MealDB, SportDB, AudioDB one by one

7acca10 yuwen1213 Thu May 18 19:42:31 2023 +0800 W14-P1: Get Cocktail data by first letter