

# Front-end: Coding Exercise

You are also welcome to submit your answers in whatever format works for you. [Codepen](#), or just a Zip file containing your code are all fine. There are total 3 questions.

## JS Algorithms and Performance

### Problem 1

Given an array of objects that define a person's name and their skills, write a javascript function that takes the given array and returns an object that maps each skill to an array of people that has that skill.

For example:

```
const people = [
  {
    id: 0,
    name: "John",
    skills: ["javascript", "html", "css", "c#"]
  },
  {
    id: 1,
    name: "Brian",
    skills: ["javascript", "java", "c", "c#", "c++", "html"]
  },
  {
    id: 2,
    name: "Michael",
    skills: ["c", "c++", "go", "rust"]
  }
];
```

Expected result:

```
{
  "c": ["Brian", "Michael"],
  "c#": ["John", "Brian"],
  "c++": ["Brian", "Michael"],
  "css": ["John"],
  "go": ["Michael"],
  "html": ["John", "Brian"],
  "java": ["Brian"],
  "javascript": ["John", "Brian"],
  "rust": ["Michael"]
};
```

## Problem 2

Write a function that takes a number up to 999,999,999.99 and returns the amount in words. This function will be used to automatically write a check after the amount in number is entered.

Examples:

1000 => "One thousand"

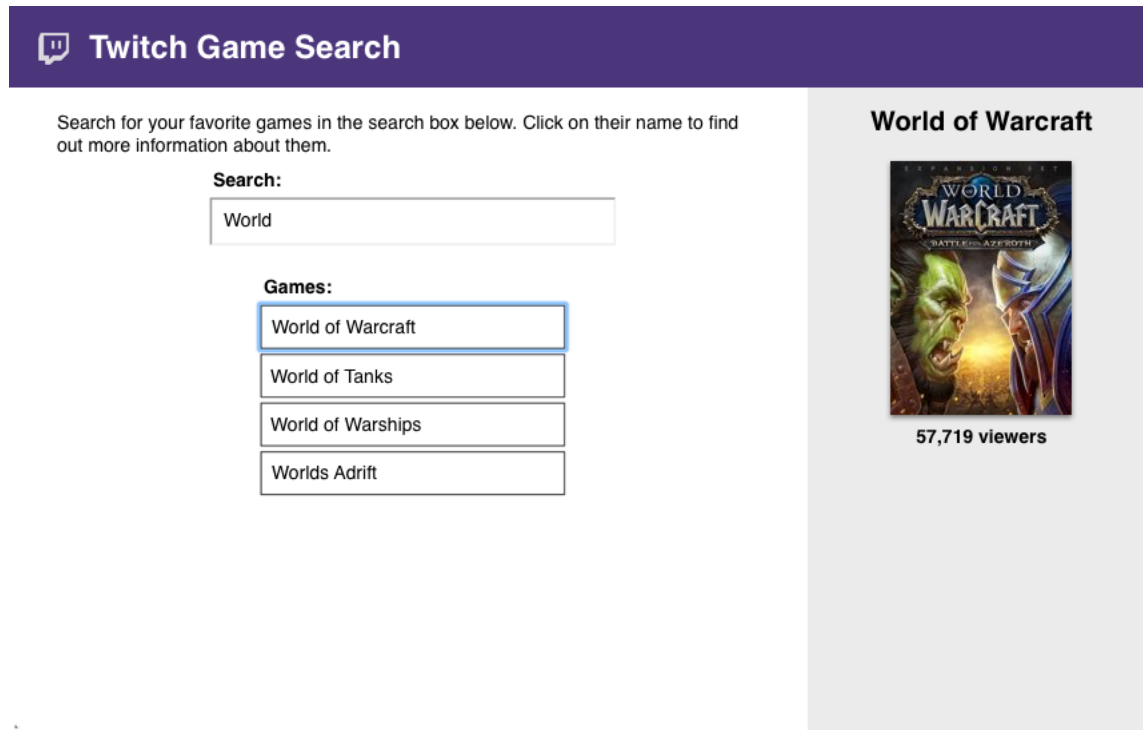
1000000 => "One million"

1545120 => "One million, five hundred and forty-five thousand, one hundred and twenty"

9999999.99 => "Nine hundred and ninety-nine million, nine hundred and ninety-nine thousand, nine hundred and ninety-nine and 99/100"

245.13 => "Two hundred and forty-five and 13/100"

### 3. React Page Layout and State Management




#### Instructions

Implement the layout pictured above using React. Feel free to use create-react-app if you find that easy to use.

- You can target modern browsers and you can use CSS pre/post-processors if desired.
- The browser window should not scroll.
- The “Main Content” and “Side Bar” portions of the page should scroll independently if necessary.
- The exact logo in the top left is unimportant.
- You don’t need to use pagination, but you can if you want.

\*\*Pixel perfection paddings, margins, margins, font-sizes, etc is **not** important. Just ensure that what you use is sane and consistent.

The user should be able to type anything into the search box. Twitch game search can be obtained for free from <https://dev.twitch.tv/docs/v5/reference/search/#search-games>. To use the Twitch API you need to register a developer account at <https://dev.twitch.tv/> and register your application. You can use <http://localhost> in the OAuth Redirect URL when registering the application.

Get StartedProductsDocsForumBlogConsole

# Dashboard

OverviewExtensionsAppsGamesDrops

## Register Your Application

Register a new application that uses the Twitch API to interact with Twitch

**Name**

Displayed to users when authorizing your application.

**OAuth Redirect URL**

Will receive the result of all client authorizations: either an access token or a failure message. This must exactly match the redirect\_uri parameter passed to the authorization endpoint. When testing locally, you can set this to http://localhost.

**Category**

Website Integration

Create

Once the game list is returned, add the game names to a list below the box. The user can click on the game's name to see more information in the side bar. The response from Twitch API should include the data you need to add to the side bar.