1. API Choice

* Browse through the [provided list](https://github.com/appbrewery/public-api-lists) and choose an API of interest. This choice should be guided by the potential to retrieve, manipulate, and present data in a meaningful and interactive way. I recommend choosing an API that does not require authentication and is CORS enabled. ([What is CORS?](https://medium.com/@electra_chong/what-is-cors-what-is-it-used-for-308cafa4df1a))

**Have decided to use a Game of Thrones API -** [**https://anapioficeandfire.com/**](https://anapioficeandfire.com/)

**No authentication is required on this API, so it only supports GET requests.**

[**https://github.com/joakimskoog/AnApiOfIceAndFire/wiki/Characters**](https://github.com/joakimskoog/AnApiOfIceAndFire/wiki/Characters) **is the root path for pulling up information on characters. Essentially want to generate a ‘what game of thrones character am I’ website which makes a request to get a particular character at random from this API.**

**Look at the ‘get a specific character’ section of** [**https://github.com/joakimskoog/AnApiOfIceAndFire/wiki/Characters**](https://github.com/joakimskoog/AnApiOfIceAndFire/wiki/Characters) **- essentially want to use math.floor(math.random) to generate a character number at random and add it to the API url to make the request. Don’t need to know the exact amount of characters in order to set this up – I know that they go at least up to 823 so can set it up for the first 500 for example. I HAVE FIGURED OUT THAT THERE ARE 2134 CHARACTERS IN THE API.**

**What data do I want to pull in about the character to show on the screen? I think the following: -**

* **Name**
* **Gender**
* **Born**
* **Died**
* **Aliases**

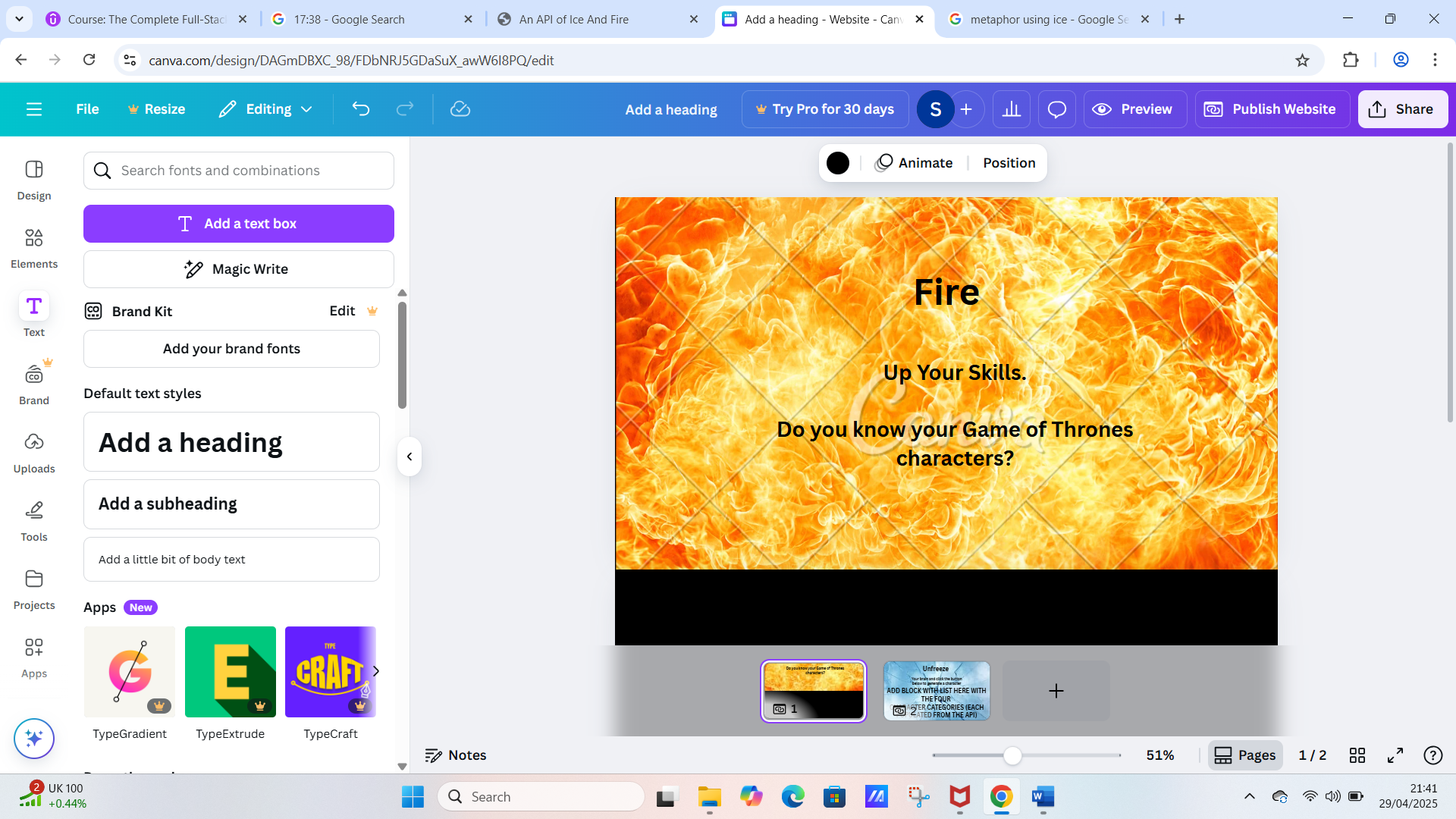
**Can I use this information to populate a table?**

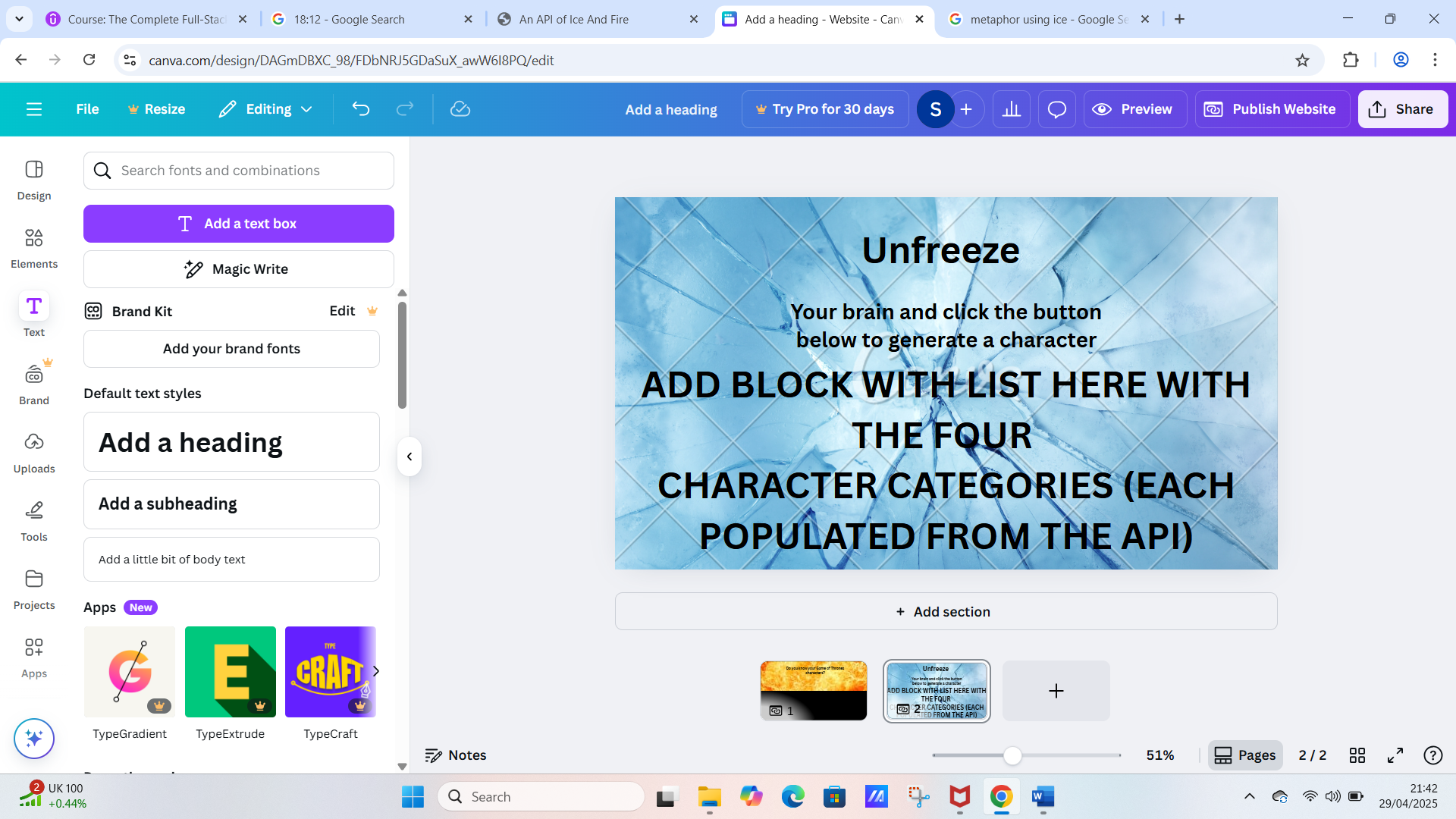
**I could have a static image that is game of thrones related, such as a background of an image of ice and one of fire, which I could use as my background in some way.**

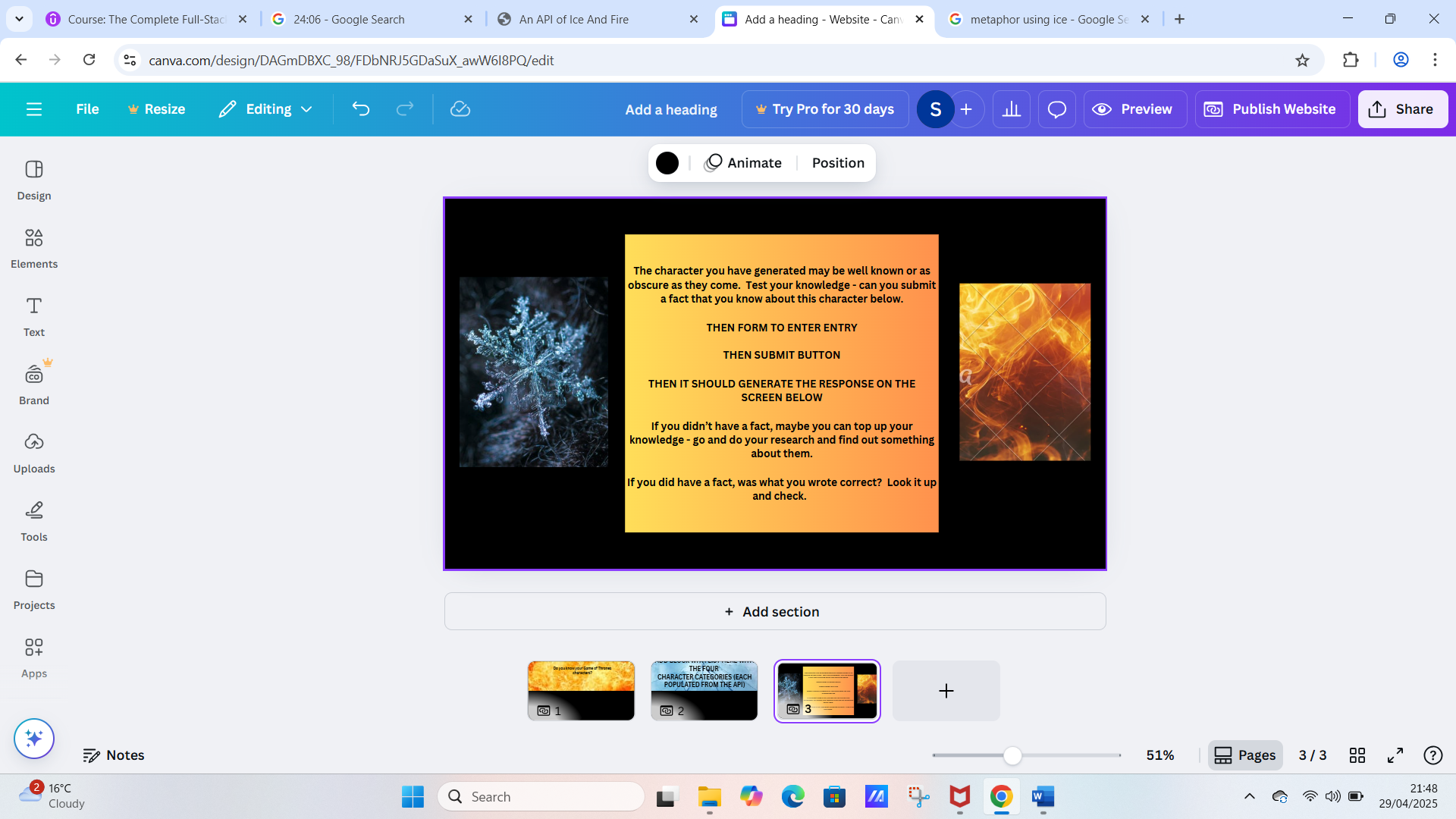
**Then I could add a form where the user could do something regarding the character, such as …. It’s a test your knowledge of GoT characters (including very obscure ones) challenge. There will be a form where it gets people to write down a fact about the person and then go and check it’s accuracy – or if they didn’t know a fact to go and look one up.**

2. Project Planning

* Think through your project, researching the chosen API, its features, what data it will provide, and how it will be used in your web application.







**Then I also want a header and footer as partials. The footer can just contain my email address and copyright statement. On the header, could I link to an external site to help people in their research – e.g. https://www.cosmopolitan.com/entertainment/tv/g12022444/minor-game-of-thrones-characters/**

3. Project Setup

* Set up a new Node.js project using Express.js.
* Include Axios for making HTTP requests.
* Include EJS for templating.
* Ensure that the project has a structured directory and file organization.

**Notes as to what to do to start with**

**Initialise Node DONE**

**Install Express, Axios, EJS, Node, body-parser (in the command line and in the index.js file).**

**App.use on body-parser.**

**Set the static folder to be the public folder and stick a blank style-sheet in there.**

**Create a ‘views’ folder and put my index.ejs file in it.**

4. API Integration

* Implement at least a GET endpoint to interact with your chosen API.
* Use Axios to send HTTP requests to the API and handle responses.

5. Data Presentation

* Design the application to present the retrieved data in a user-friendly way. Use appropriate HTML, CSS, and a templating engine like EJS.

6. Error Handling

* Ensure that error handling is in place for both your application and any API requests. You can console log any errors, but you can also give users any user-relevant errors.

7. Documentation

* Include comments throughout your code to explain your logic.

8. Code Sharing

* Use what you have learnt about GitHub to commit and push your project to GitHub so that you can share it with other students in the Q&A area, I'd love to see what you've build too! You can tweet at me @yu\_angela
* Include a Readme.md file that explains how to start your server, what commands are needed to run your code. e.g. npm i  and then nodemon index.js