Cover

====================================================

This project was a challenge to see what I could do with API’s and ReactJS with no prior knowledge of React. It is an example of what I can accomplish in a short period of time.

### The Challenge

I was challenged by a company to build a small project. This project would involve using some new technologies, and it was a test to see what I could do. I was very interested in using this new technology and was excited to add it to my project list.

### The Instructions

Create a basic website that shows launch data for SpaceX rocket missions in a grid/table.  Your web application should:

* Display the date and time of each mission, the rocket name, a link to the cargo manifest, and whether the launch was a success.
* Display the time of the mission in Central Time (CT).
* Order the launches in reverse chronological order.
* For each launch, include a column that ranks the payload mass, with the heaviest payload being rank 1.
* Work in any modern browser (i.e. Chrome, Firefox, IE10+, Safari, etc.).
* Use the React JavaScript framework.

### My Story

I have a background in Java Core and Visual Basic. I had just finished Centriq Training, learning Full Stack web development with .Net. I had one other course in web technologies the used NodeJS. I have some good skills, but this project was definitely challenging every step of the way.

I chose AG-Grid as my grid product. The recommended grid product only came with a 30 day free trial. Since I wanted this project on my website for more 30 days, I went with a product that had a community version.

The requesting company gave me more time for the project. They wanted me to see if I could replicate my HTML version in React with a table, without AG-Grid.

### My Timeline

* **Friday Afternoon** – I received the challenged and started looking at the date in Json format and working on the Javascript call to retrieve the data.
* **Sunday** – I decided to make sure I could complete all the data requirements, so I built a version out of HTML which I have displayed on the website.
* **Monday** – I finished the HTML version. I started working with NodeJS and React.
* **Tuesday** – I finished working with various grid products in the morning and settled on AG-Grid. I figured how to get a React program deployed on my website.
* **Wednesday** – I hit the wall. I started to work with the way React handles Json. This is where all the problems occurred, probably because I have never worked with React or JSX. From what I have seen, React will load the Json and store it in the state. The issue is, the SpaceX data had lots of changes that needed to take place before the grid displayed it. All examples of setState were basic and would not come close to what I needed to do. Some of the setState examples never worked in my React program. I spent a couple of hours building and playing with a REST API with express. I wanted to see if I could filter the SpaceX data. It was apparent that this was not the right way to go. Time to call it a day and go with the fields I could display without issues. I realized this project required much more time and knowledge. I contacted the company for further instruction.
* **Friday** –The requesting company gave me more time to complete the project without using AG-Grid.
* **Monday** – I reviewed several YouTube React lectures. I removed AG-Grid. I ran into issues with the current code not wanting to load the SpaceX data.
* **Tuesday –** I fixed the SpaceX data loading issue. Got UTC time, flight number, mission name, rocket name and launch status added to the table.
* **Wednesday -**   Payload mass calculation and cargo manifest added to the table. I had issues with displaying HTML in React. I researched and added NPM package ReactHtmlParser. I had issues with building the rank table and getting function to see the data.
* **Thursday –** I adjusted date/time to display local time. I finished the rank table by adding it to the State so functions could process it. I wrapped up the website and code.

### Conclusion

This was a really cool challenge, and I am grateful for the opportunity to take on this project. I was able to complete the project successfully. At first, I was unable to get AG-Grid to do some of the more complex work, but I was completely successful using a HTML table in React. It was very apparent this project would have flowed smoother if I had previous experience in using React. Every step was a challenge, including getting React to work, working with React State and learning to use NodeJS. I am looking forward to taking classes to acquire this knowledge for future projects.

### Project Links

Here are some links of the resources I used for the project.

* API info - [SpaceX API](https://github.com/r-spacex/SpaceX-API)
* API Call - [Json List](https://api.spacexdata.com/v3/launches)
* React installing and getting demo running - [Learn React - Programming with Mosh](https://www.youtube.com/watch?v=Ke90Tje7VS0)
* Restful API - [How to create an API RESTfull in Express (Node.js) without database](https://medium.com/@etiennerouzeaud/how-create-an-api-restfull-in-express-node-js-without-database-b030c687e2ea)
* Source Code - [HTML and React source code.](https://github.com/jercoinc/SpaceXapi)
* NPM Package ReactHtmlParser - <https://www.npmjs.com/package/react-html-parser>