**Overview**

Implement a basic calculator using HTML, CSS and JavaScript. The objective of this exercise is to demonstrate fluency with modern web tools and techniques. We ask that you complete everything in the requirements section. You may choose to complete items in the bonus points section but not doing them will not be counted against you.

**Requirements**

* Please user ember.js as your framework.
* Please implement your own solution. ***Using a npm package like js-calculator is not acceptable.***
* The calculator should be able to add, subtract, divide, and multiply both integers and decimals.
* The calculator should be able to combine multiple operations (i.e. 2 x 10 + 5), and have a button for clearing results.
* There should be a page in the application where the following api is hit "<https://s3.amazonaws.com/gavant-public/themes>” to load in 3 different themes from a server. After selecting one of those 3 themes, the user will automatically navigate back to the calculator, show the title of the current theme at the top of the page, and change the styles to match the theme name. The actual styles are up to the developer. There should be 1 default theme that would be displayed on page load. After you load in the themes from the api the default theme should remain an option. Please use ember data. The following are relevant/helpful links: <https://guides.emberjs.com/v2.16.0/tutorial/ember-data/>, <https://guides.emberjs.com/v2.16.0/models/customizing-adapters>
* Include all supporting materials such as unit tests and/or documentation in your submission.

**Bonus points**

* Using a CSS pre-processor like SASS.
* Using ES6 JavaScript concepts.
* Incorporating sound, animation or other elements that elevate user experience
* The calculator should be responsive. It should work in all screen sizes.

**Considerations**

* ***This exercise is not timed and you are not rewarded for a fast turnaround.***This exercise should be treated as any development task – code should be clean, robust and production-worthy. Ultimately, you should consider “Would I commit this to a repository for all to see?”
* The conventions, standards, and best-practices of any frameworks you are utilizing should be adhered to.
* Thoughtful design paradigms should be considered. Be prepared to defend your design choices and/or the details of your implementation.

**Deliverables**

Your submission should be submitted as a link to a GitHub or GitLab repository - both services offer free accounts for hosting repositories.