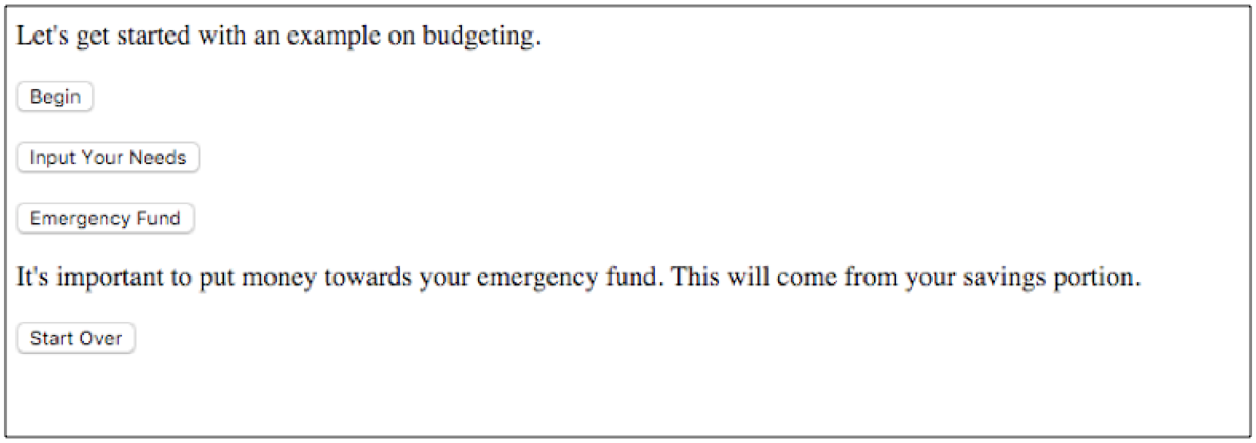
AnnieCannons Function Exercise

Your first assignment is to write a program to help with budgeting.

1. Create the starting page.


Let's get started with an example on budgeting.

It's important to put money towards your emergency fund. This will come from your savings portion.

Create an entire folder system on your Desktop called “JavaScriptBudgetingProgram”. Add an HTML file, a scripts folder, a custom.js file, a styles folder, and a main.css file. Link the files together in your HTML, and add the jQuery library.

In the HTML file, create four buttons in order: “Begin”, “Input Your Needs”, “Emergency Fund”, and “Start Over”.

You will also add two lines of text, as seen in the screenshot. You should add IDs to each line, as we will need to access them individually with our JavaScript.

You also will need to create an empty paragraph after each button (with a corresponding ID). This is where the user’s answers to the information should be stored. EX) <p id=”part1”></p>

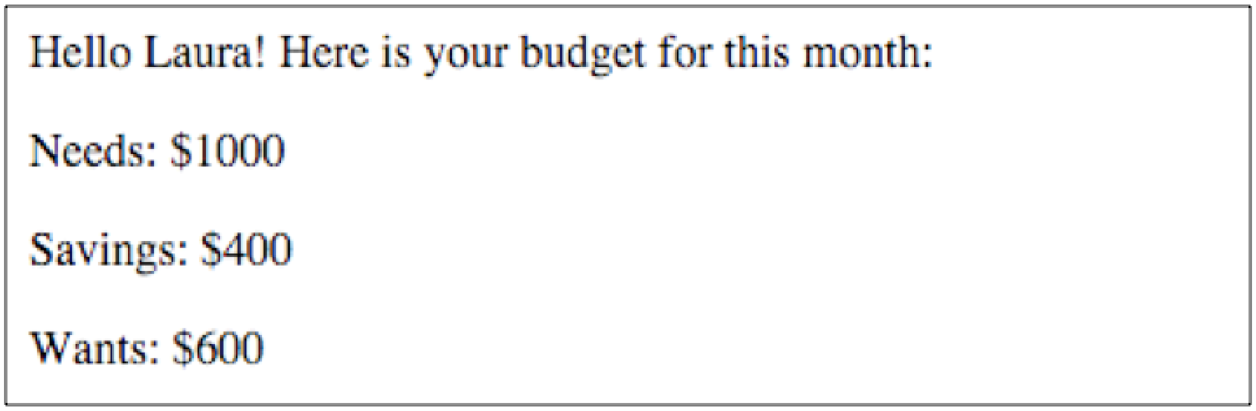
2. Work on the “Begin” button.

In the JavaScript file, create five variables: “person”, “budget”, “needs”, “savings”, and “wants”. “person” should be set initially to the blank string, “”. All other variables should be set to 0.

The function for the “Begin” button should have the following steps:

1. We will create a function called myBudget and set it to run when the button ‘Begin’ is clicked (in our HTML).
2. In the function myBudget:
    a. Prompt the user for their name or ID. Store the user input in the variable “person”.

b. Prompt the user for their monthly salary. Store the user input in the variable "budget". (Note, when testing out this part, please don't use dollar signs...only express the salary as a number.)
c. "needs" should be changed to 50% of "budget"; "savings" should be changed to 20% of "budget"; and "wants" should be changed to 30% of "budget".
d. Using .append(), print out this information in the following format so that is shows in the browser. (For this example, the user name is "Laura" and the user's monthly salary is 2000. Different user inputs should have different results.)



```
Hello Laura! Here is your budget for this month:

Needs: $1000

Savings: $400

Wants: $600
```

3. Work on the "Input Your Needs" button.

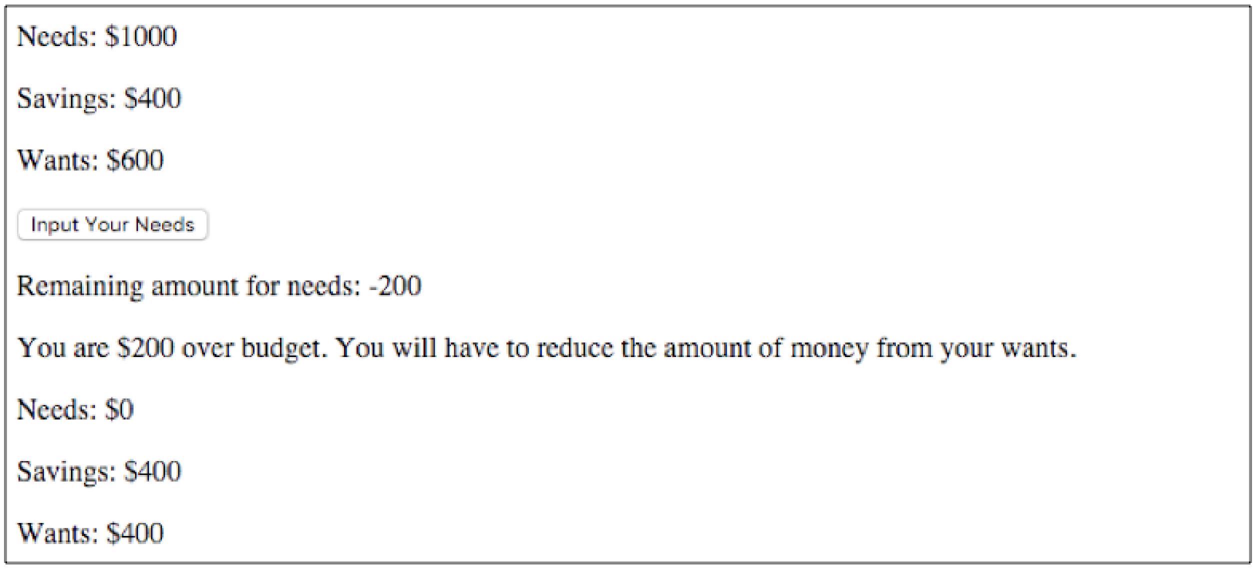
Create a function called inputYourNeeds. This function should be called when you click on the button "Input Your Needs" (onclick="inputYourNeeds").

The function for the "Input Your Needs" button should have the following steps:

1. Prompt the user on how much they spend on food. Subtract this from the "needs" variable.
2. Prompt the user on how much they spend on housing. Subtract this from the "needs" variable.
3. Prompt the user on how much they spend on bills and loans. Subtract this from the "needs" variable.
4. Prompt the user on how much they spend on healthcare. Subtract this from the "needs" variable.
5. Prompt the user on how much they spend on transportation. Subtract this from the "needs" variable.
6. Prompt the user on how much they spend on other essentials. Subtract this from the "needs" variable. Use the append() to print out how much is remaining in their "needs".

If “needs” is less than zero (the user spent more on needs than they had in the budget), use append () to print out how over budget they are and subtract it from the “wants” variable. If “needs” is over zero, you should use append () to print “You are doing great!”.

Print out the remaining needs, savings, and wants.



```
Needs: $1000

Savings: $400

Wants: $600
```

```
Remaining amount for needs: -200

You are $200 over budget. You will have to reduce the amount of money from your wants.

Needs: $0

Savings: $400

Wants: $400
```

(example for over-budget)

4. Work on the “Emergency Fund” button.

Create a function called “Emergency Fund”. This function should be called when you click on the button Emergency Fund.

1. Once the user clicks the “Emergency Fund” button, the line of text about emergency funds should go away.
2. Prompt the user on how much they will put towards their emergency fund and store the answer in a variable called funds. If the user inputs more than they currently have in their “savings” budget, prompt the user again by saying “That’s too much. Try a smaller amount” and then subtract this from the “savings” variable. Else, just subtract the initial amount the user enters from the savings variable.
3. Print out their “needs”, “wants”, and “savings” in a similar table as the ones in the previous buttons.

6. Work on the “Start Over” button.

When you click on the “Start Over” button, you should run a function called startOver. Create this function, and then do the following:

1. Set all global variables (“person”, needs”, “wants”, “savings”, “budget”) back to their original states: 0 or the empty string.
2. Clear all text the text that has been created with the .empty() method. The screen should look like the original in Part I.

BONUS: Add Bootstrap or other CSS to make the program look a little bit nicer for your user.