Your assessment Meets Some Expectations

This was a good attempt and working submission. However, I would like you to go over my notes and checkin with me on data structure work and iteration.

1. Can create a local git repository and submit the assessment with a complete git log of relevant commits for each feature. Please leave more meaningful commit messages. I would like to see a description for each feature you write.

2. Can create a method and use parameters to return the desired calculation result. Good work! Solid control flow and you kept it simple. Refactor to name your variables num_1, operator, num_2. This will help improve the readability of your code.

3. Provide a UI that takes user input which can be formatted and added to the calculate method.

4. Use looping in UI run many calculations. Use a controlled loop to end when the user is finished. Nice work with the looping. Instead of using a loop do in your user interface try and allow the loop to break naturally each time. 'break' is very abrupt and should be avoided when at all possible. e.g until input == 'done' Good string formatting. I like that you made an extra method to do this.

5. Use a data structure to store user input and calculation results. Use Ruby data structure enumeration to display calculation history with results. Lets refactor this to be a single data structure. I suggest a hash { ""3 + 5"" => 8}. This will also improve your iteration at the end to use .each

6. Create validation protections in code to ensure proper user input. No validations present.

7. Create a zipped folder of your work and upload the file to canvas. Meets expectations