Project #3

You have been hired by a bank to manage their loan department. One of your responsibilities is to write a JavaScript program that will compare loans with various interest rates.

Your program should first consist of a form that allows the user to enter the loan amount (float), a starting yearly interest rate (int), an ending yearly interest rate (int), and the number of years of the loan (int). The goal of your JavaScript program is to compute what the monthly payment will be for each interest rate beginning with the starting rate and finishing with the ending rate in increments of 1. Your output will be presented within a table.

To compute monthly payment, use the formula..

$$\text{Monthly Payment} = \frac{c * r * (1 + r)^n}{(1 + r)^n - 1}$$

where
c = loan amount
r = monthly interest rate (yearly interest rate / 1200) this also converts to a decimal for your formula
n = number of months of loan (years * 12)

For the formula, you will need to use a few of the JavaScript Math functions.

You must also compute the total payment of the loan. This is a very simple calculation, so I will let you figure it out.

Your JavaScript must properly obtain and parse the user's input from the form. Next, create a loop to create the table. Your output must display all of the user's input, and the table. Your table must alternate row colors. **You must alternate the row colors of your table using CSS.** Look this up, easy to find and easy to do.

Also, the CSS file is located at…
http://newton.ncc.edu/gansonj/ite154/templates/ite154proj3sp19template.txt

You will also be graded on the following.

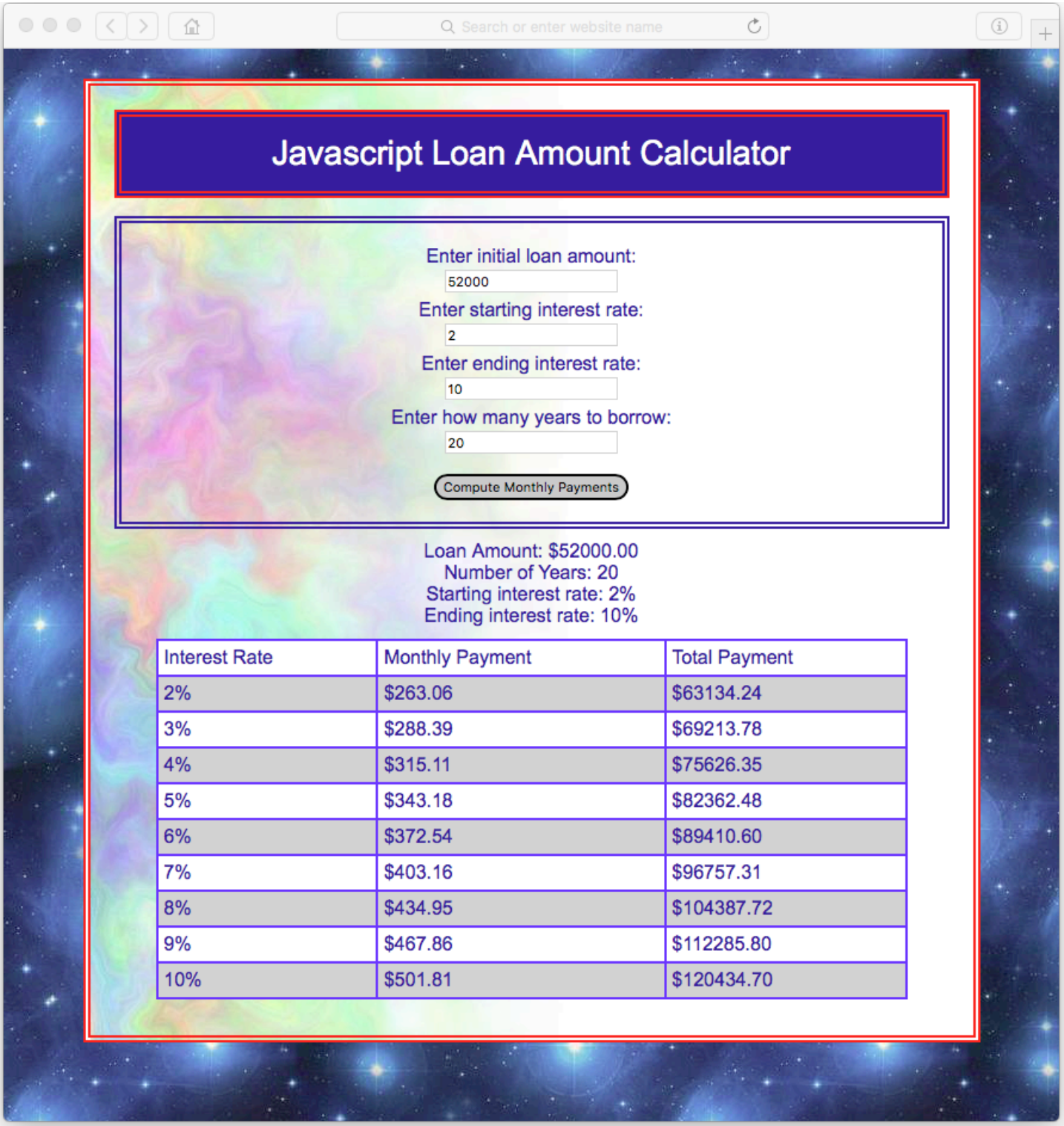
- Code must be properly indented. All brackets must line up and all method bodies must be uniformly indented.
- Your code must contain a comment at the top specifying your name, project number, date, and course and section.
- Variable names must be intuitive.
- All closing brackets MUST be commented as to what they are closing as shown in class.
- All numbers must be displayed to 2 decimal places by properly using the toFixed() function.
- All monetary values must be displayed with a $ sign.

Your output should look exactly like the two sample outputs listed on the following pages.

Sample run 1:

Interest Rate	Monthly Payment	Total Payment
4%	$87.55	$12607.61
5%	$92.49	$13318.42
6%	$97.59	$14052.24
7%	$102.84	$14808.69
8%	$108.25	$15587.32
9%	$113.80	$16387.64

Sample run 2:



Interest Rate	Monthly Payment	Total Payment
2%	$263.06	$63134.24
3%	$288.39	$69213.78
4%	$315.11	$75626.35
5%	$343.18	$82362.48
6%	$372.54	$89410.60
7%	$403.16	$96757.31
8%	$434.95	$104387.72
9%	$467.86	$112285.80
10%	$501.81	$120434.70