Plan for quiz:

1. Home screen will contain 2 choices; one is for a quiz and the other is a drawing game

* Blue Quiz
* Red Quiz

Score: variable score created

1. Blue Quiz: 2 vacation paths; only right way is valid and left way doesn’t work
2. Mexico path takes to first question: Which flag is Mexico’s flag?
   1. If wrong options are clicked, then taken to a screen showing answer
   2. If correct answer is chosen, then taken to a screen saying “correct,” and 1 point is added to variable score
   3. Button taking to next question is present, on both result screens
3. Question 2: What is the capital of Mexico?
   1. If wrong options are clicked, then taken to a screen showing answer
   2. If correct answer is chosen, then taken to a screen saying “correct,” and 1 point is added to variable score
   3. Button taking to next question is present, on both result screens
4. Question 3: What year was the Mexican revolution? (1910)
   1. If wrong options are clicked, then taken to a screen showing answer
   2. If correct answer is chosen, then taken to a screen saying “correct,” and 1 point is added to variable score
   3. Button taking to next question is present, on both result screens
5. Final screen shows score out of 3
7. Red Quiz: exploring with neo and trinity
8. Taken to a page with drawing game:
   1. Set 2 sliders; one for y and one for x position
   2. Use get command to show turtle where sliders are, based on what user puts x and y as
   3. Change pen color based on button clicked
   4. Ask for pen size in a prompt

**College Board Standards:**

| **Rubric** | **Explanation** |
| --- | --- |
| Program Purpose and Function | * Inputs: user inputs which answer they want for blue choice path, and inputs x/ y value, color, and size of turtle when drawing * Purpose: to create a quiz and drawing game that users can play with * Outputs: score and drawing |
| Managing Complexity | No lists used   * If loops are used to set the pen size in red answer choice path |
| Procedural Abstraction | * Shown on planning doc * All choices/ response plans are mapped out on this docx file (above) |
| Algorithm Implementation | * This platform did not allow for algorithm implementation, so on event blocks were used repeatedly |
| Testing | * Console method outputs were used for debugging purposes and to see where any code went wrong if there were problems |
| Data Abstraction | * I stored a variable for score, and added 1 every time user got the correct answer * There were also variable for x and y position in the drawing game, which were determined by user |