CodeX Remix Task

MISSION 10: Reaction Tester	**PROJECT: Reaction Tester v.2**	**# HOURS:** 1-2
PROJECT GOALS: Students will apply their knowledge of lists and random numbers to build a problem-solving application.	**EXTRA PROJECT MATERIALS:** ● **none**	**VOCABULARY:** ● Loop ● Logic operators ● Parameter

LEARNING TARGETS:

- I can write a function to make my code more efficient and readable.
- I can utilize multiple variables to a new program and describe their purposes.
- I can utilize loops to make my code more efficient.

SUCCESS CRITERIA:

□ Include two images to correspond to Button A and Button B.
□ Require player to hit the matching image to win.
□ Use conditions to branch a “win” or “lose” option.

RUBRIC:

CSTA Standard	Basic (3)	Proficient (4)	Mastered (5)
Documentation			
2-AP-10 Use flowcharts and/or pseudocode to address complex problems as algorithms.	Incomplete flowcharts.	Flowcharts provided for each process.	Flowcharts provided for each process. Evidence of revisions and improvements made.
Algorithms and Programming			
2-AP-11 Create clearly named variables that represent different data types and perform operations on their values.	No variables; variables not named appropriately.	Variables used and named correctly in most instances.	Variables are used and named correctly in each process as needed.
2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.	No loops or conditionals.	Loops or conditionals used correctly in most instances.	Loops or conditionals are used correctly in each process as needed.
2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.	Code is not organized or readable. No comments.	Code is sometimes organized into problems and subproblems in order to make it organized and readable. Comments used inconsistently.	Code is decomposed into problems and subproblems, making it easy to follow and read. Comments are clear and easy to understand.
2-AP-14 Create procedures with parameters to organize code and make it easier to reuse.	No procedures; procedures not named appropriately.	Procedures used and named correctly in most instances.	Procedures used efficiently to organize code and reused as needed.
Debugging			
2-CS-03 Systematically identify and fix problems with computing devices and their components.	Code bugs not identified; little or no documentation of fixes.	Code bugs mostly identified and fixed; adequate documentation of fixes.	Code bugs identified and fixed; extensive documentation of fixes.