Andela

Andela Nigeria Cycle 43 Technical Challenge.

Instructions for submission

Create an account on codepen.io and attempt **any 1** of the questions. You are required to make use of **only** HTML, CSS and JavaScript, and **NO FRAMEWORKS**. Please submit via this form before **4pm on Monday, March 4th, 2019.**

Question 1

Armstrong	
Context	An Armstrong number of 3 digits is an integer that the sum of the cubes of its digits is equal to the number
Task	Write a program to check whether an integer is Armstrong number or not. Example: **371 is an Armstrong number since 3**3 + 7**3 + 1**3 = 371**
UI Design	The App should have ● An input field which takes in the number to be checked. ● A button to run a check and a div to display the result. If the number is not a 3 digit integer, the button should be deactivated and if the user tries to click on it while deactivated, a message should appear explaining why the button can't be clicked. ● A div to display the result

Question 2

Circular primes	
Context	Circular primes are primes in which numbers formed from the rotation of the digits are themselves primes e.g. 197, which has rotations, 197, 971 and 719. For more context, visit this link.
Task	Write an algorithm that takes a number and returns the number of circular primes below the given number, following these conditions: 1. The number **must** range between **100 - 100,000**. 2. Return the number of circular primes below the number given. Example: ● For **100**, the number of circular primes is **13**
UI Design	The app should have ● An input text field to collect the number. ● A button to perform the operation. ● A div to display the result.

Question 3

Kaprekar's constant	
Context	Take any four-digit number (whose digits are not all identical) and do the following; 1. Rearrange the string of digits to form the largest and smallest 4-digit numbers possible. 2. Take these two numbers and subtract the smaller number from the larger. 3. Use the number you obtain and repeat the above process. **The most amazing thing is this: *Every* four-digit number whose digits are not, all the same, will eventually hit 6174, in at most 7 steps, and then stay there!** For more information, use this link.
Task	Write an algorithm to compute the Kaprekar's constant on any four-digit number. The algorithm must do the following: ● Take **only** four-digit numbers whose digits are not identical. ● Return the number of steps taken to get to the Kaprekar's constant (6174). Example: using 3524. 5432 – 2345 = 3087 8730 – 0378 = 8352 8532 – 2358 = **6174** 7641 – 1467 = **6174** **This above example should return "3", because it took 3 steps to get to the constant.**
UI Design	The app should have ● An input text field to collect the four-digit number. ● A button to perform the operation. ● A div to display the result.