**Enhance – Junior Developer: Coding Problems**

We would like you to have a go at the coding problems below plus the short project described at the bottom of this document. You do not have to complete all of the problems; we would just like to see some example of your code on smaller problems as well as get a sense of how you go about solving problems in general. We would prefer that you attempt each problem in either Ruby, Python, JavaScript, or TypeScript, as these are the programming languages that we use most commonly as a business.

Please push any solutions and your project up to your GitHub accounts and provide a link to your account.

**Problem 1 – FizzBuzz:**

Write a short program that prints each number from 1 to 100 on a new line.

For each multiple of 3, print "Fizz" instead of the number.

For each multiple of 5, print "Buzz" instead of the number.

For numbers which are multiples of both 3 and 5, print "FizzBuzz" instead of the number.

**Problem 2 – Anagram Checker:**

Given two strings, write a function to check if they are an anagram of each other, i.e. they have the same characters but in a different order. Your function should return true for the case when the two strings are anagrams for each other and false in all other cases.

Examples:

anagram(triangle, integral) => true

anagram(wool, howl) => false

**Problem 3 – Deck of Cards:**

Write a program that creates a standard deck of playing cards in memory, shuffles it, and deals a number of cards, specified in standard input, as text to the command line as well as the sum of the card values (treat ace as one and picture cards as 10). This is a good opportunity to demonstrate your understanding of Object Oriented Programming. I won’t provide the expected output of this problem as I want to see what you come up with.

**Problem 4 – Largest Dot Product:**

Given two arrays of equal length, write a function that returns the largest value that can be obtained by multiple each number in one array by one number in the other, using each number in each array only once.

Examples:

largest\_dot\_product([2, 1, 4], [1, 6, 4]) => should return 33, i.e. 4\*6 + 2\*4 + 1\*1

largest\_dot\_product([1, 2, 3], [3, 2, 1]) => should return 14, i.e. 3\*3 + 2\*2 + 1\*1

**Problem 5 – Prime Number Checker:**

Write a function that accepts a single integer as its argument and returns true if the number is a prime number and false if it isn’t.

Examples:

prime\_checker(4) => false

prime\_checker(3) => true

prime\_checker(13 => true

**Project: Calendar Application**

As a project, we would like you to attempt making a simple calendar application in Ruby on Rails. The application should render a calendar that you can schedule appointments against. You can decide on how you implement this problem and architect it in the way that you wish.

If you wish to complete an extension to this, consider adding one or more of the following features:

* Users to the app, each with their own calendars and appointments.
* Email notifications of appointments or reminders of appointments that are soon.
* Notes that can be saved against appointments.
* The ability to add recurring appointments.