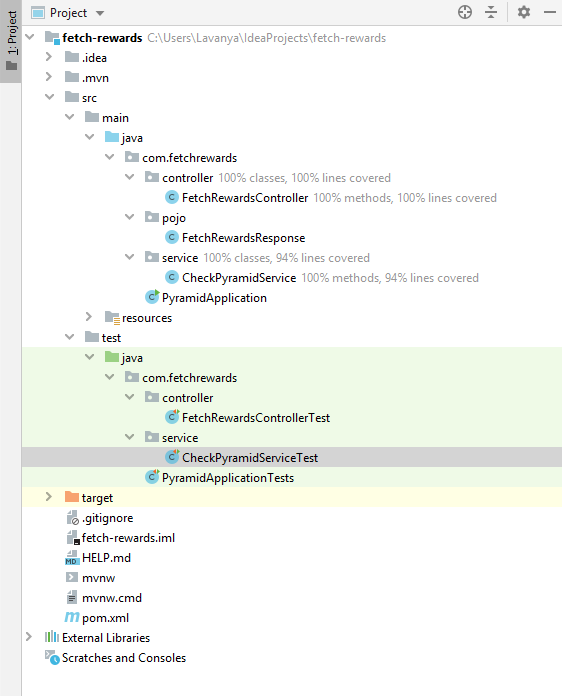
|  |
| --- |
| **FETCH REWARDS-CODING EXCERCISE** |
| Lavanya Nemani Role: Backend Software Engineering |
| Problem statement: Accept a string as input and return a response indicating whether a word is a **pyramid word**. A word is a **pyramid word** if you can arrange the letters in increasing frequency, starting with 1 and continuing without gaps and without duplicates. |
|  |

TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| S.no | Content | Page No |
| 1 | TOOLS USED AND PROJECT STRUCTURE | 2 |
| 2 | INTRODUCTION TO PROJECT | 3 - 4 |
| 3 | TEST CASES | 5 |
| 4 | REQUEST AND RESPONSE SNIPPETS | 6 - 10 |
| 5 | DEVELOPER DETAILS | 12 |

|  |  |
| --- | --- |
| Framework | Spring boot |
| Build tool | Maven |
| Language | Java 8 |
| IDE | Intellij |

TOOLS USED & PROJECT STRUCTURE



INTRODUCTION TO PROJECT

**Introduction:**

Language used: Java

Implementation of fetch-rewards project:

Fetch-rewards application determines if a word is pyramid or not when an input string is given.

Input: any String

Output: It Is in the format of json structure with a status message, like given below:

If a given string is pyramid: “Given string is pyramid”

Else, given string is not a pyramid: “Given string is not pyramid”

**Notes and Assumptions:**

1. Input string is not case-sensitive. When the input is given in case-sensitive, it converts each character into lower-case letters.
2. Input is considered only for letters. If numbers or special characters are given in a string, they are ignored.
3. When input string contains only one kind of a character, it returns false
4. Ran this project in my localhost, port: 8080.

**Used Controller Annotations:**

@RequestMapping: Maps http requests to controller.

@PostMapping: It is specifically used to map for HTTP post request

**Controller method:**

When a response is sent in a string, the input is passed to a service class’s method where it returns the results in Boolean value. Based on the Boolean value, the response is given.

**Service method:**

Collections used: HashMap and ArrayList

HashMap is used to store the non-duplicated values. Hence the given string characters are stored in hashmap. Once the iteration is passed through all characters of the given string, the hashmap stores the character and the count of the encountered character.

Using ArrayList, hashmap values are sorted and checked if the numbers are consecutive or not.

When the numbers are consecutive, the return is true, if not, the return is false.

The Boolean value is returned to the calling method.

TEST CASES

**Testcases:**

Used Junit and Mockito to write the test cases for this project.

**@MockBean:** It allows to mock the class to verify the behavior of it.

**MockMvc:** It is used to test the server-side testing.

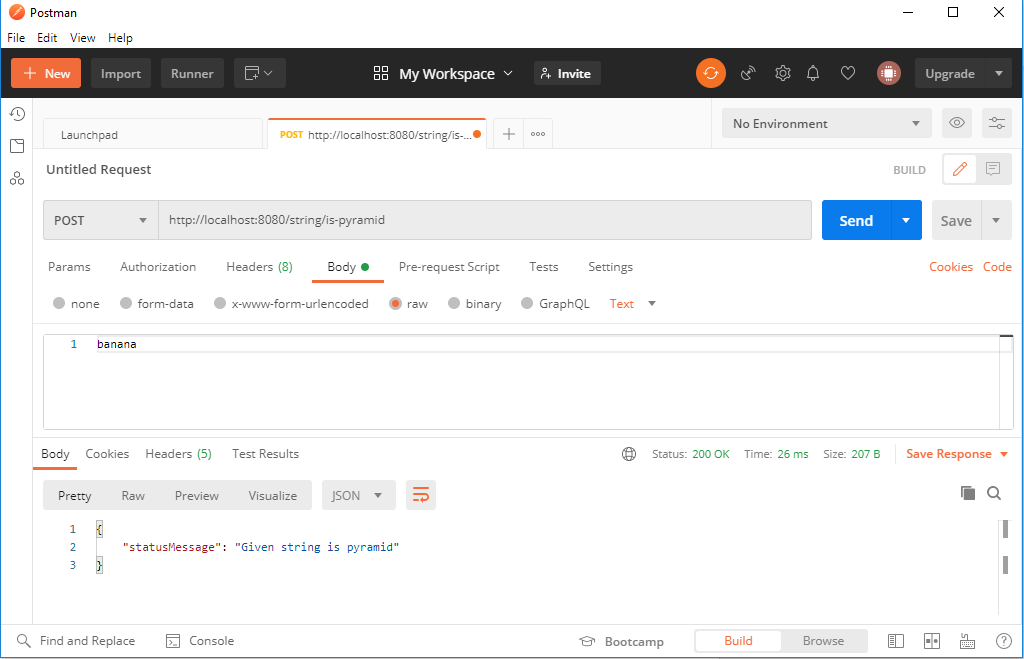
ControllerTest and ServiceTest:

Three-unit test cases are written to check the behavior of controller and service. They are as follows:

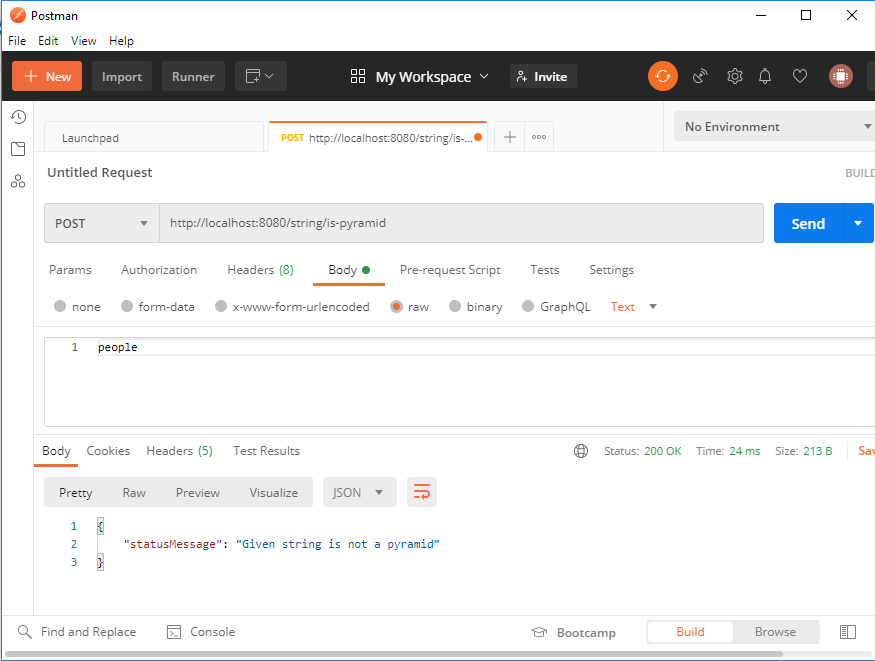
1. Positive test case
2. Negative test case
3. Invalid input test case

REQUEST AND RESPONSE SNIPPETS

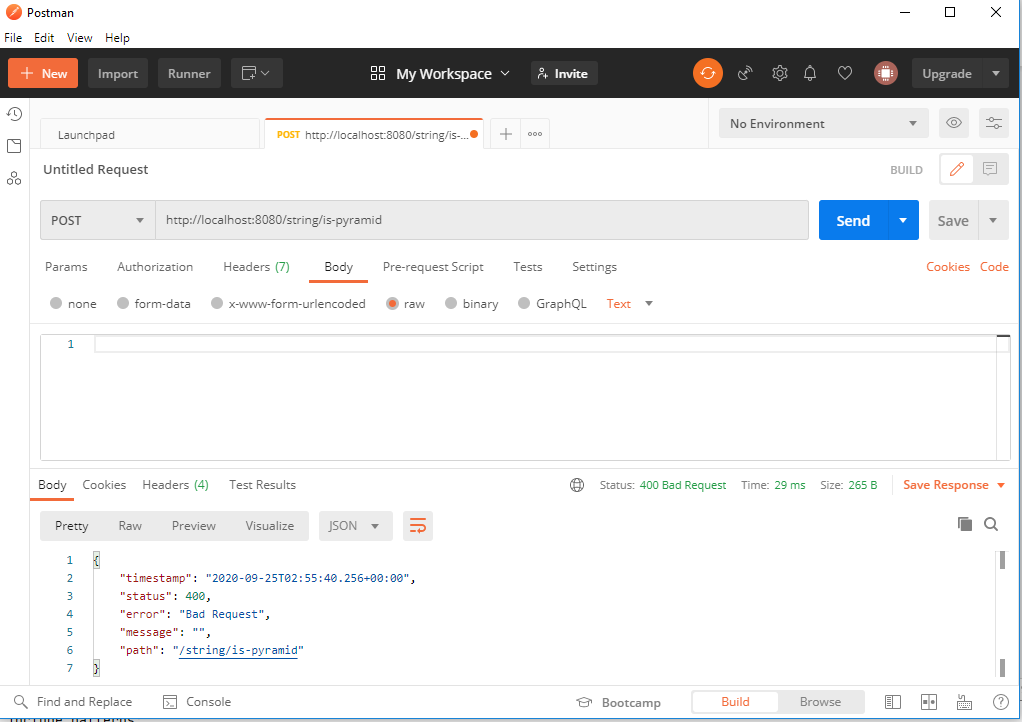
**When an input string is “banana”**



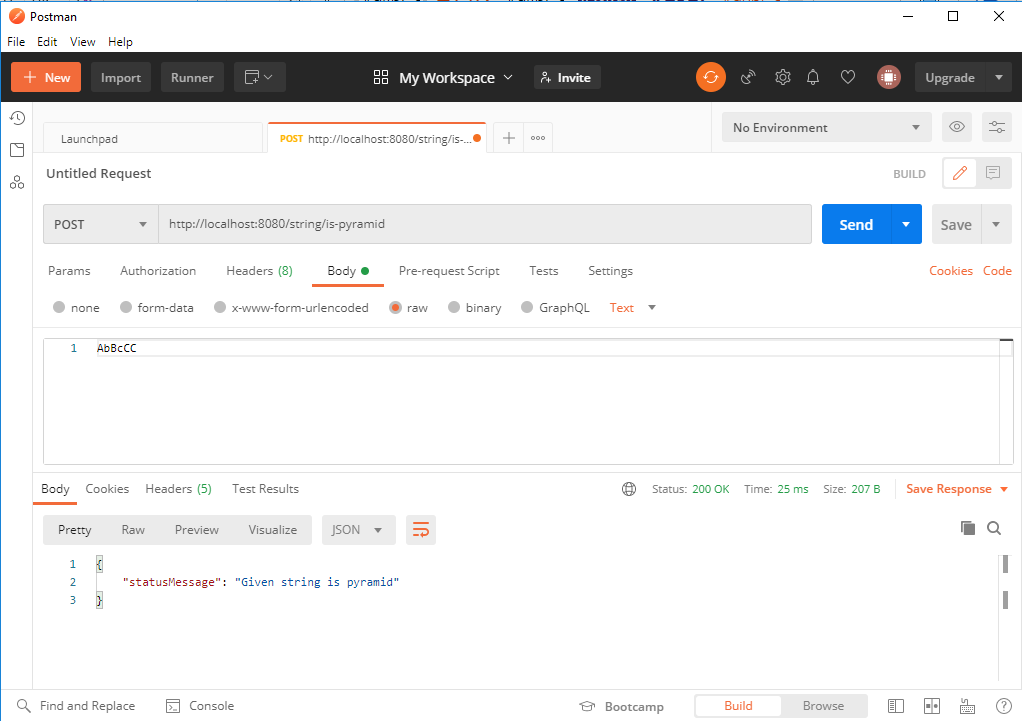
**When an input string is “people”**



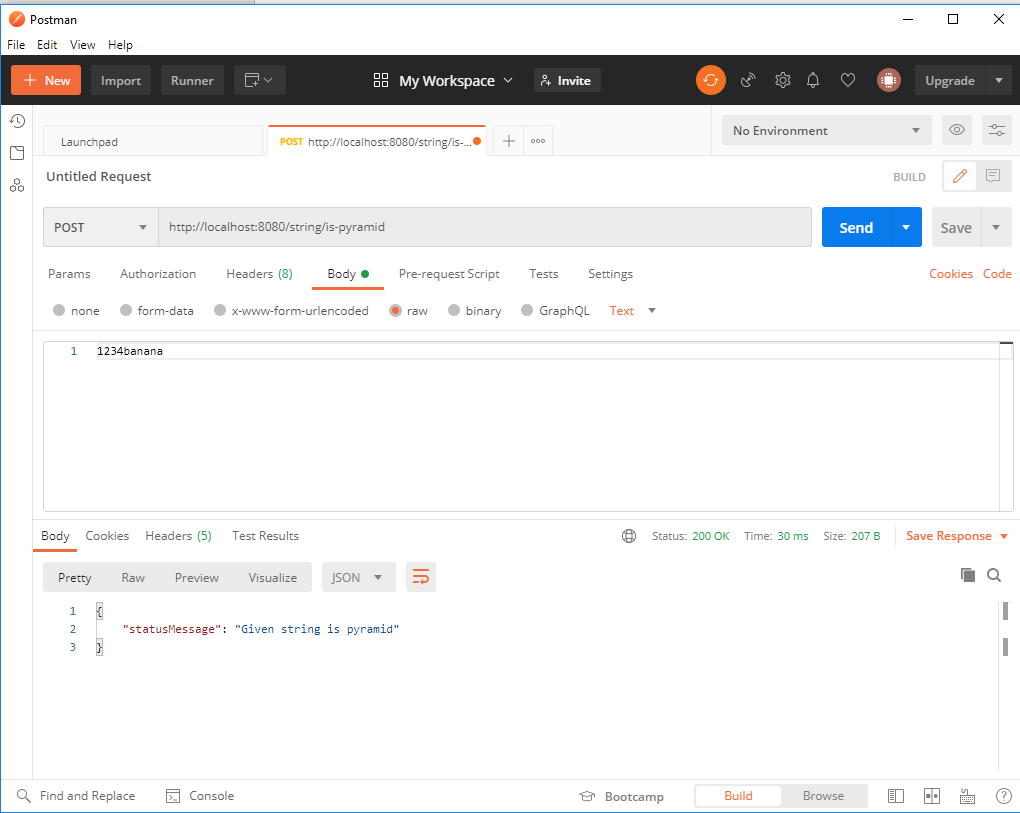
**When input string is empty:**



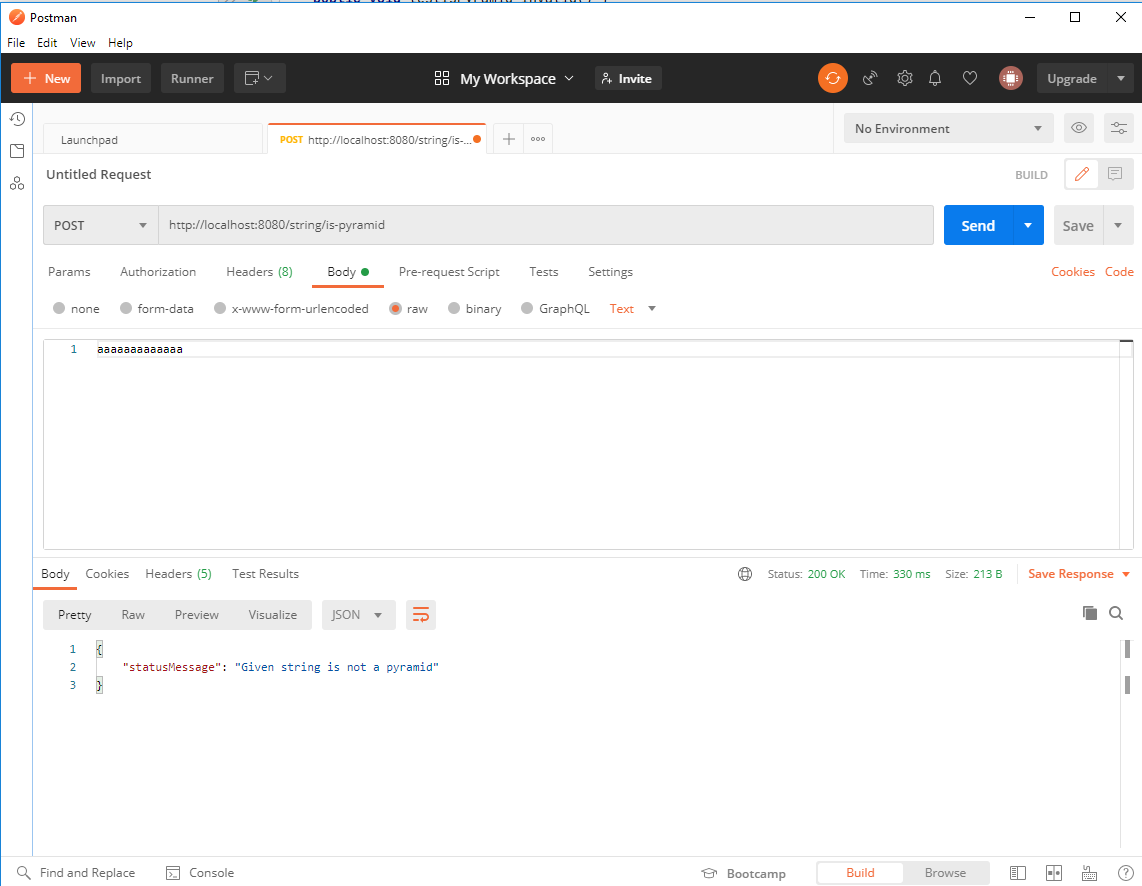
**When an input string is not case-sensitive: AbBcCC**



**When an input string contains numbers and letters together: 1234banana, it only considers letters**



When an input is one kind of a character: aaaaaaaaa



DEVELOPER DETAILS

Name: Lavanya Nemani

Email Id: Lavanya.nemani07@gmail.com