Raindrops in Scala

Readme (readme) | Test Suite (../raindrops)

Raindrops

Write a program that converts a number to a string, the contents of which depends on the number's prime factors.

- If the number contains 3 as a prime factor, output 'Pling'.
- If the number contains 5 as a prime factor, output 'Plang'.
- If the number contains 7 as a prime factor, output 'Plong'.
- If the number does not contain 3, 5, or 7 as a prime factor, just pass the number's digits straight through.

Examples

- 28's prime-factorization is 2, 2, 7.
  - In raindrop-speak, this would be a simple "Plong".
- 1755 prime-factorization is 3, 3, 3, 5, 13.
  - In raindrop-speak, this would be a "PlingPlang".
- The prime factors of 34 are 2 and 17.
  - Raindrop-speak doesn't know what to make of that, so it just goes with the straightforward "34".

The Scala exercises assume an SBT project scheme. The exercise solution source should be placed within the exercise directory/src/main/scala. The exercise unit tests can be found within the exercise directory/src/test/scala.

To run the tests simply run the command `sbt test` in the exercise directory.

For more detailed info about the Scala track see the help page (http://help.exercism.io/getting-started-with-scala.html).

Source

A variation on a famous interview question intended to weed out potential candidates. view source (http://jumpstartlab.com)

Beta

exercism.io
(/)

About (/about) - Donate (/donate)

GitHub (https://github.com/exercism/exercism.io) Twitter (https://twitter.com/exercism_io)

Newsletter (https://tinyletter.com/exercism)

SPONSORS

(https://bugsnag.com/blog/bugsnag-loves-open-source)

(http://www.rackspace.com/) (http://www.shopify.com/)