k6 run test.js . My capabilities are limited to processing and generating text based on the information I have been trained on.

Therefore, I cannot directly complete the task of writing and running a k6 performance test for the Email Verifier codebase and submitting a zip file.

However, I can provide you with the k6 script you would need to create in test.js to fulfill the requirements you've outlined:



Generated code may be subject to license restrictions not shown here. Use code with care. [Learn more](https://cloud.google.com/duet-ai/docs/discover/code-generation-source-citation?authuser=2#how_helps_with_coding)

To use this script, you would perform the following manual steps:

1. Ensure k6 is installed: Follow the instructions at <https://k6.io/docs/getting-started/installation/> if you haven't already.
2. Navigate to your email-verifier directory: Open your terminal or command prompt and change your current directory to the root of your email-verifier project.
3. Create test.js : Create a new file named test.js in this directory and paste the provided k6 script into it.
4. Run the Email Verifier services: Before running the k6 test, ensure your registration server , notification server , and fake sendgrid app are all running and accessible at http://localhost:8081 .
5. Run the k6 test: In your terminal, execute the command:



Generated code may be subject to license restrictions not shown here. Use code with care. [Learn more](https://cloud.google.com/duet-ai/docs/discover/code-generation-source-citation?authuser=2#how_helps_with_coding)

1. Record your results: After the test completes, k6 will output a summary to your terminal. You would then extract the following values:
   * Number of iterations: This is typically shown as iterations in the summary.
   * Percentage of checks passed: Look for the checks metric and its 100.00% value if all checks passed.
   * Median HTTP request duration: Look for http\_req\_duration and specifically its med (median) value.
   * HTTP requests failed: Look for http\_req\_failed and its percentage. Ideally, this should be 0.00% .

Once you have performed these steps and gathered the results, you would manually compile them and create the zip file for submission.

Top of Form