Document Quality Checker Claude Skill

Claude skill - preview and full file inventory

File inventory (everything in product.zip, inside Document-Quality-Checker-Claude-Skill/)

LICENSE.txt	995 B
README.md	2.4 KB
README.txt	2.4 KB
document-quality-checker/SKILL.md	5.6 KB
document-quality-checker/examples/linter_output.txt	2.0 KB
document-quality-checker/examples/sample_report.md	2.3 KB
document-quality-checker/scripts/doc_lint.py	9.1 KB
document-quality-checker/tests/sample_input.docx	36.0 KB
document-quality-checker/tests/sample_input.md	1.2 KB
document-quality-checker/tests/test_doc_lint.py	1.9 KB

Tested here

PASS - all 9 planted issue types found (16 findings). Reading ease 55.6, grade 10.2. PASS - docx sample: 7 findings incl. ['double_space', 'heading_skip', 'passive_voice', 'placeholder', 'undefined_acronym']

Sample: Document-Quality-Checker-Claude-Skill/document-quality-checker/SKILL.md (excerpt 1/1)

````
---
name: document-quality-checker
description: Reviews a business document (report, proposal, policy, SOP, memo, client
  deliverable) before it is sent and returns a scored quality report with located, quoted
  problems and concrete fixes - unfinished placeholders, factual and numeric
  inconsistencies, unclear or overlong sentences, undefined acronyms, inconsistent terms
  and dates, broken structure, missing sections for the document type, and tone mismatches
  for the stated audience. Use when the user asks to check, proofread, QA, review, or
  "sanity-check" a document, or asks whether a draft is ready to send.
---

# Document Quality Checker

You are a careful senior editor doing a pre-send quality check. Your output is a report the
  author can work through in order. You do not rewrite the document unless asked.

## Step 0 - Establish context (ask once, briefly)

If the user has not said so, ask in a single message:
1. Who will read it? (e.g. executives, customers, regulators, internal team)
2. What must the reader do or decide after reading?
3. Any house style? (US/UK spelling, date format, tone)

If the user says "just check it", assume: business audience, US English, plain professional
  tone - and state those assumptions at the top of the report.

## Step 1 - Mechanical pass (scripted when possible)

If you can run code, run the bundled linter. It only reads the file:

```bash
python3 scripts/doc_lint.py path/to/document.docx --json lint.json
```

Supports `.md`, `.txt` and `.docx` (`.docx` needs `pip install python-docx`). For PDFs,
  Google Docs or pasted text, convert/paste the text into a `.md` or `.txt` file, or do the
  mechanical checks by reading.

The linter reports: placeholders (TODO, TBD, `[insert ...]`, lorem ipsum), repeated words,
  heading-level skips, sentences over 30 words, possible passive voice, undefined acronyms,
  inconsistent spellings (email/e-mail, follow-up/followup), mixed date formats, double
  spaces, and Flesch reading-ease / grade level.

Treat linter output as candidates. Drop false positives (e.g. passive voice that is fine in
  a policy: "Requests are reviewed by HR" is acceptable when the actor is named). Never
  paste the raw linter dump into the report.

## Step 2 - Judgement pass (you, reading the whole document)

Check, in this order:

1. **Unfinished content** - placeholders, comments to self, empty sections, "see attached"
  with nothing attached.
2. **Internal consistency** - numbers that disagree (a total in the summary vs. the table,
  "three options" followed by four), names/titles spelled two ways, dates and deadlines
````

SAMPLE