# Build your own Scan Manager

# Stage 3 — Robust Error Handling & Retries

Real‑world services are resilient. Add fault tolerance by detecting job failures, retrying, and reporting a summary.

## Your Task

* Jobs sometimes return errors, check for status != “success” or a raised exception.
* Retry a failed job up to 3 times using exponential back‑off delays (1 s, 2 s, 4 s).
* If all retries fail, write a log entry with `status=FAILED` and the exception message.
* After all tasks finish, print (and log) a summary: total tasks, successes, failures, total runtime, and average execution time per task.

## Deliverables

* Extend your Stage 2 solution (or create `process\_tasks\_resilient.py`) to run jobs resiliently.
* `tasks.log` demonstrating at least one simulated failure and its retries.
* Console or separate `summary.log` with the aggregated statistics.