PROMPT CACHING

Cut your Claude bill: *5 steps*

Cached text is read back at a fraction of the normal price. Set your prompts up so Claude can reuse the same text.

1. **Put the fixed stuff first**
   Instructions, brand voice, examples and reference files go at the start of the prompt. The question that changes goes last.

2. **Keep the fixed part identical**
   A cache only hits when the repeated part is exactly the same. Keep dates, names and random IDs out of it.

3. **Reuse it before it expires**
   Cached text lasts 5 minutes by default (1 hour is an option) and each reuse refreshes it. Send related requests close together.

4. **Do the math**
   A 5-minute cache write costs 1.25x the normal input price and pays for itself after one cache read. The 1-hour option costs 2x and pays off after two reads.

5. **Check that it is working**
   Look at the cache read and cache write token counts in the response usage. If reads stay at zero, something in your fixed part keeps changing.

Prices: On Sonnet 5.5, cache hits cost $0.10 per million tokens (base input is $2). On Sonnet 5 they cost $0.20.

Source: Anthropic pricing page; Claude Code changelog v2.1.296