

Prompting How-To

Cheat Sheet Operativo A(I)doption

Problema -> Prompt -> Output
-> Feedback -> Revisione

1 Regola madre

- Non ottimizzare a caso: scegli prima se stai migliorando problema, prompt o risultato.
- Definisci sempre output atteso e criteri di qualità prima di eseguire.

2 Decision tree rapido

- Problema confuso: usa D (rewrite) o E (socratico).
- Prompt debole: usa A (CTI) o B (prompt critique).
- Risultato mediocre: usa H (critica focalizzata) o M (rubrica).
- Task complesso: usa J1/J2 con output intermedi e checkpoint.
- Output eseguibile richiesto: usa U (code-first) e V (debug) se necessario.

3 Stack operativo minimo

- 1) D/E: chiarisci il problema e i vincoli reali.
- 2) A: costruisci prompt CTI con output specifica esplicita.
- 3) M/H: valuta con criteri e correggi i punti deboli.
- 4) K/L: genera alternative e scegli con criteri se serve.
- 5) J2/Q: usa checkpoint o plan-act-check nei lavori lunghi.

4 Template CTI standard

- [CONTESTO] ruolo modello, scenario, destinatario.
- [TASK] deliverable misurabile e cosa NON produrre.
- [ISTRUZIONI] vincoli, criteri qualità, tool/fonti, formato.
- [OUTPUT] struttura finale, lunghezza, tono.
- [CHECK] assunzioni, limiti, aderenza al task.

5 Controlli qualità obbligatori

- Guardrail: non inventare dati; se manca info fai domande.
- Rubrica: accuratezza, completezza, chiarezza, aderenza, struttura, actionability.
- Checkpoint: stop esplicito nei passaggi critici prima di proseguire.
- Tool-aware: separa evidenze, sintesi, limiti.

6 Anti-pattern da evitare

- Prompt lungo e generico senza formato output.
- Cambiare troppe leve insieme e perdere causalità.
- Codice senza input/output definiti e senza smoke test.
- Revisione a intuito senza rubrica o criteri.
- Nessuna distinzione tra fatti verificati e inferenze.

Prompting How-To

Cheat Sheet Operativo A(I)doption

Problema -> Prompt -> Output
-> Feedback -> Revisione

7 Checklist finale prima del rilascio

- Output aderente al task e al destinatario.
- Assunzioni e limiti dichiarati chiaramente.
- Formato rispettato e contenuto action-oriented.
- Per codice: smoke test e edge case eseguiti.

8 Pattern operativo: una leva alla volta

- In ogni iterazione cambia solo una componente: contesto, task, istruzioni o output format.
- Mantieni il resto invariato per capire davvero cosa ha prodotto il miglioramento.
- Salva versione prompt + output per tenere traccia della progressione.

9 Domande chiave prima di promptare

- Qual è l'obiettivo operativo concreto di questa richiesta?
- Quali vincoli non sono negoziabili?
- Qual è il formato output realmente utile per agire?
- Quale rischio voglio ridurre (errore, ambiguità, tempo, compliance)?

10 Checklist code-first

- Ambiente e versioni dichiarate (es. Python 3.11).
- Input e output definiti con schema minimo.
- Gestione errori essenziale e logging minimo inclusi.
- Almeno 2 smoke test + 1 edge case obbligatori.