>_ A Legolcsóbb és Leggyorsabb Reverse Engineering Környezet

[INFO] Optimalizált AI munkafolyamatok: LeanREA + Deepseek V4.1 Flash + PandaCode

Gemini Notebook

A Cenzúra Fal: Miért Tört El a Régi Rendszer?

A Szükséges Feladatok

- ✓ Electron app kommunikációjának elemzése (native add-on).
- ✓ Függőségek (könyvtárak) verzióinak ellenőrzése binárisokban.
- ✓ Forráskód nélküli szoftverek hibakeresése (debugging).



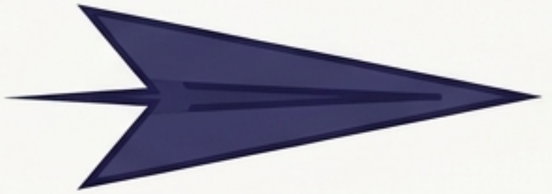
Megtagadva.

A Claude Code és a CodeEx elkezdték blokkolni a standard REA-stílusú promptokat. A normál reverse engineering munkafolyamat ma már fennakad a biztonsági szűrőkön.

Gemini Notebook

A Token-Pazarlás Csapdája





34,000 Token

A Régi REA: Az eredeti MCP szerver 207 beépített eszközt tölt be. Egyetlen 'Okay' válasz is százezres költséggel indul a definíciók miatt.

A Cél: Egy könnyített, felesleges definíciók nélküli rendszer, amely azonnal a tényleges munkára fókuszál.

A Következmény: Olcsóbb modellekre váltva a kontextusablak azonnal megtelik felesleges definíciókkal, még mielőtt a tényleges munka elkezdődne.

Gemini Notebook

Az Új Paradigma: A Tökéletes Felépítés



Ez a hármas kombináció áttöri a cenzúrát, és töredékére csökkenti a token-költségeket.

Gemini Notebook

1. Pillér: LeanREA – A Kivonás Művészete

$$\left[\begin{array}{c}\text{REA}\\(207\ \text{Eszköz})\end{array}\right] + \left[\begin{array}{c}400\text{k}\\\text{Token}\end{array}\right] - \left[\begin{array}{c}\text{Felesleg}\end{array}\right] = \left[\begin{array}{c}\text{LeanREA}\\(1.8\text{kb}) +\\34\text{k Token}\end{array}\right]$$



+38% Gyorsabb



-35% Olcsóbb



-56% Kevesebb interakció (turn)

Nincsenek beépített eszközök. A 1.8kb-os fájl csupán megmondja az ügynöknek, melyik parancs a leggyorsabb (pl. Assar, Otool, Jadx), és kész parancsokat ad át futtatásra.

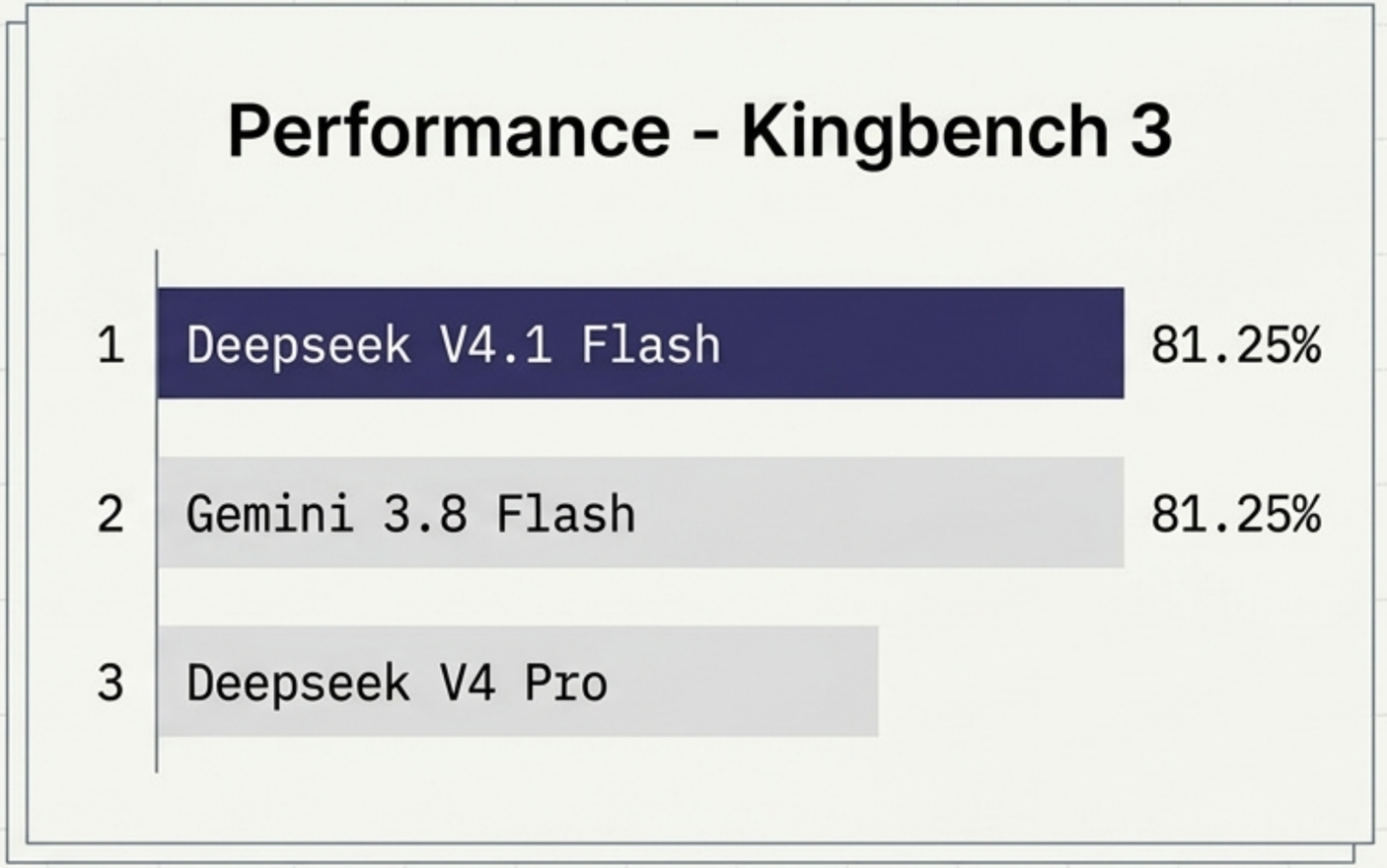
Gemini Notebook

2. Pillér: Deepseek V4.1 Flash – Szabadság és Intelligencia



Nincs visszautasítás (Refusal).

Zökkenőmentesen hajt végre kódkivonást, visszafejtést és elemzést a normál reverse engineering munkafolyamatok során. Saját kódolási környezetén (DSH) lett kiképezve.



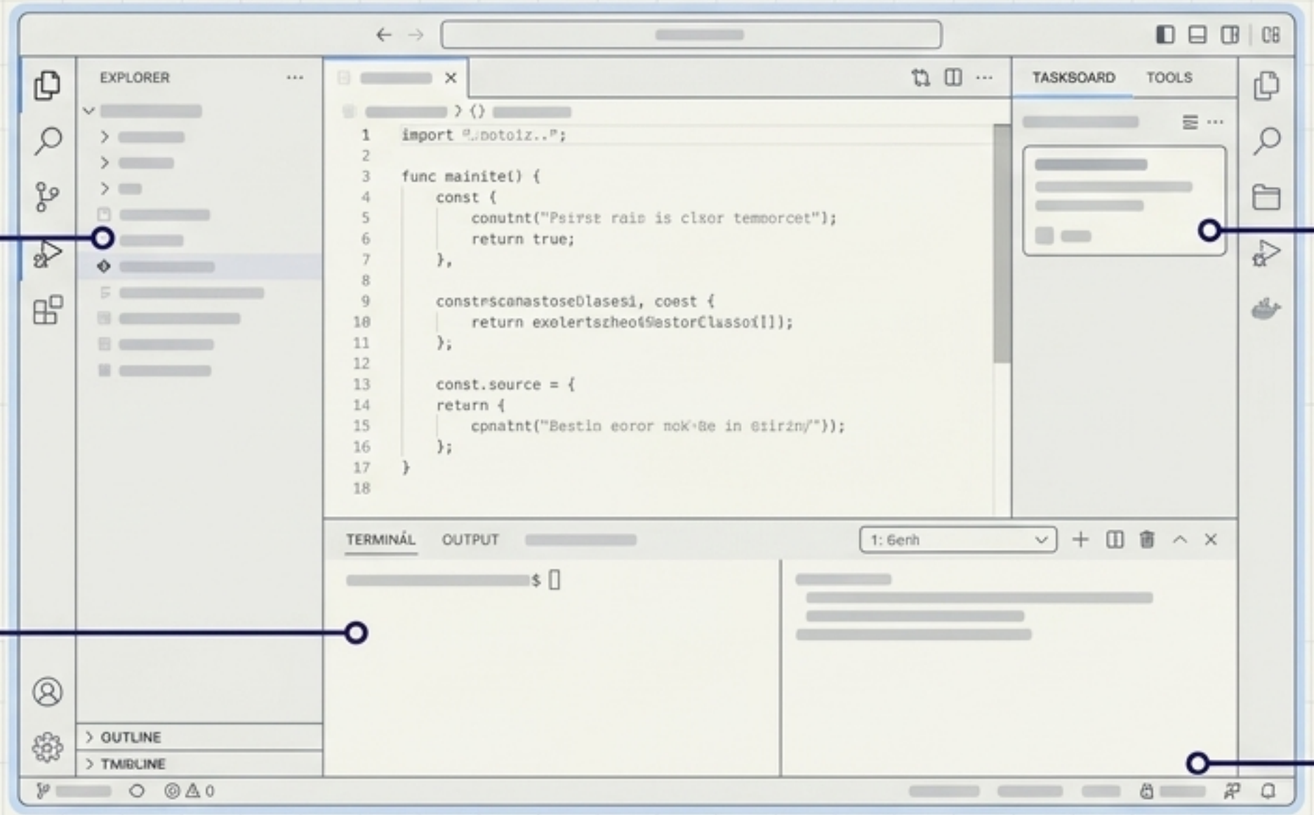
A maximális teljesítményhez a 'Thinking' (Gondolkodás) funkciót maximumra kell állítani.

Gemini Notebook

3. Pillér: PandaCode – A Központi Irányítópult

Git Panel: Beépített verziókövetés és fájlváltozás-követés.

Terminál: Közvetlen parancssori hozzáférés.



Taskboard: A folyamatban lévő feladatok áttekintése.

Élő Költségkövetés: API költségek valós idejű becslése és monitorozása.

Natív támogatás a Deepseek DSH (Deepseek Harness) rendszeréhez, mindez egyetlen ablakban.

Gemini Notebook

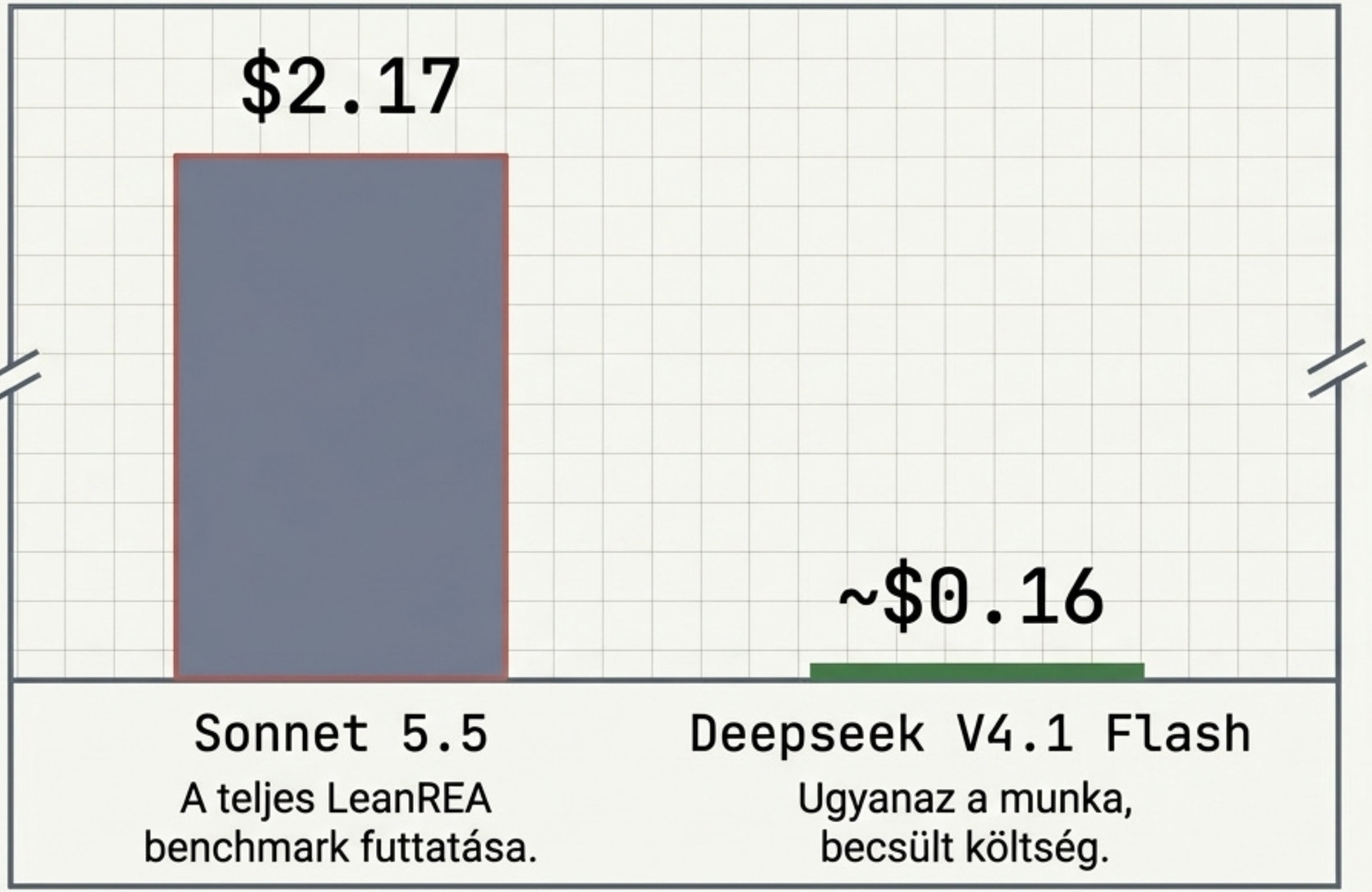
Ár-Összehasonlító Mátrix ($ / 1 Millió Token)

	Input	Output	Cached Input
Sonnet 5.5	$2.00	$10.00	$0.20
Opus 5.5	$4.00	$20.00	$0.40
Deepseek V4.1 (Off-Peak)	$0.15	$0.60	$0.003

A kódoló ügynökök folyamatosan ugyanazt a kontextust küldik újra (turns), így a tokenek nagy része Cached Input. Ebben a metrikában a Deepseek **~67-szer olcsóbb**, mint a Sonnet. (Csúcsidőben a Deepseek árai duplázódnak, de még így is töredékébe kerülnek).

Gemini Notebook

Mit Jelent Ez a Gyakorlatban?



Egy cenzúrázott, drága környezetből átlépünk egy olyan rendszerbe, amely a feladatokat korlátozások nélkül, mindössze néhány centért végzi el.

Gemini Notebook

Telepítési Playbook 4 Lépésben

Step 1: PandaCode Telepítése

Letöltés a GitHub Releases oldalról (OS specifikus fájl).

Step 2: Deepseek Harness (DSH)

```
npm install -g @deepseek-ai/dsh
```

A PandaCode automatikusan megtalálja és futtatja a háttérben.

Step 3: API Kulcs Beállítása

PandaCode Settings -> Providers. Vagy környezeti változó: DEEPSEEK_API_KEY. (A kulcs lokális marad).

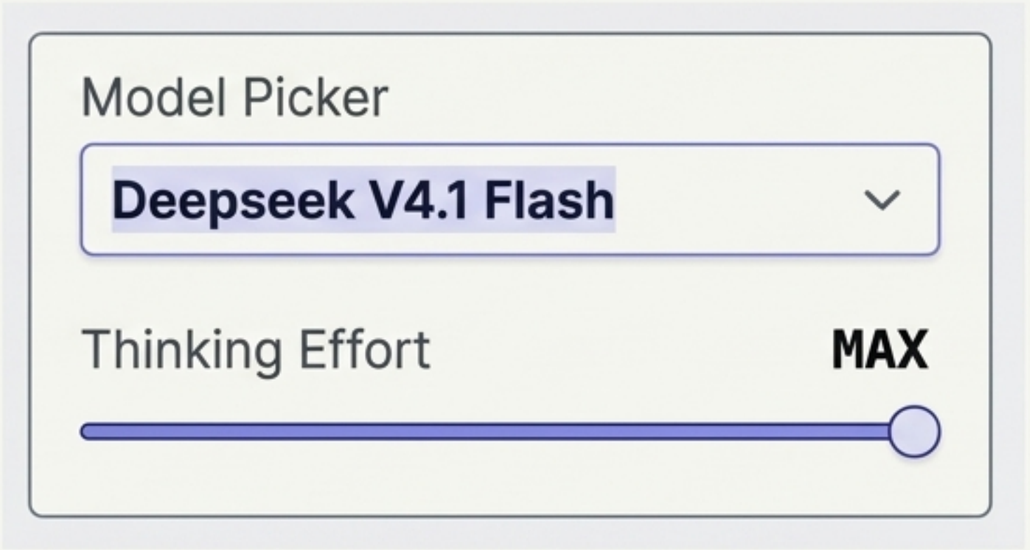
Step 4: LeanREA Hozzáadása

```
npx skills add AICodeKing/leanrea -g
```

A projekt mappában futtatva, a -g flag a globális eléréshez.

Gemini Notebook

Konfiguráció és Futtatás



1. Modell Kiválasztása

A **Model Picker**-ben válaszd a **Deepseek V4.1 Flash** opciót.

2. Kritikus Beállítás

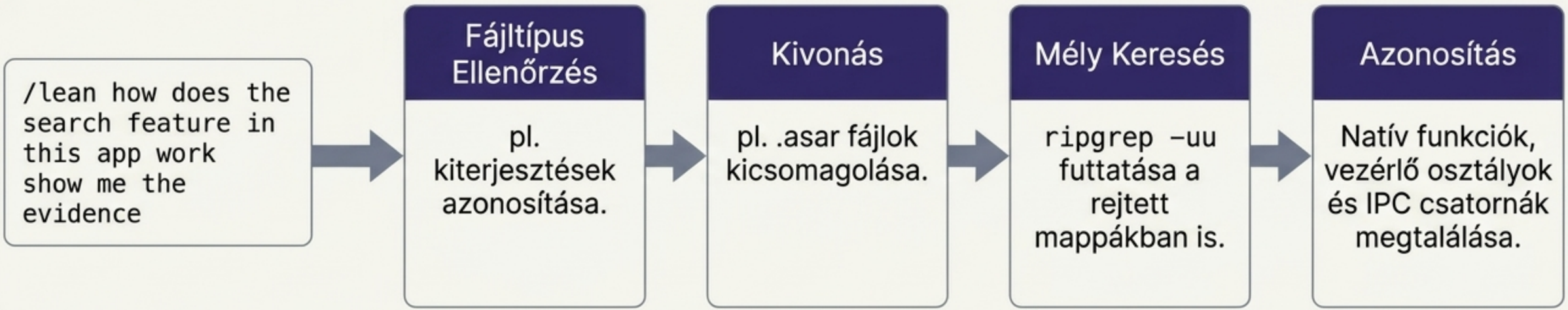
Állítsd a **Thinking Effort** (Gondolkodás) csúszkát MAX-ra. (Tesztelés alapján e nélkül a teljesítmény majdnem 30 pontot esik).

3. A Készség Behívása

Gépelj egy / (slash) jelet, és válaszd a `/leanrea` opciót a listából. (Ha nem jelenik meg, csak utasítsd az ügynököt a 1.8kb-os lean fájl elolvasására).

Gemini Notebook

Az Ügynök Munkafolyamata a Gyakorlatban



A folyamat végén a rendszer az állításokat kategorizálja: Látott (Seen), Kikövetkeztetett (Inferred), vagy Ismeretlen (Unknown). A LeanREA utasítására a kis ellenőrzéseket **egyetlen shell hívásba (batch)** foglalja, drasztikusan csökkentve az interakciók számát.

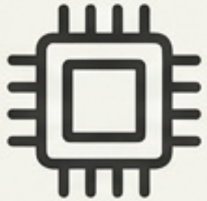
Gemini Notebook

Korlátok és Kompromisszumok



Adatvédelem

A kód a DeepSeek szervereire kerül. Vállalati vagy szigorúan privát projektek esetén ellenőrizni kell az adatvédelmi szabályzatokat.



A Legnehezebb Feladatok

Bár a Deepseek kiváló, a legmélyebb, legkomplexebb visszafejtési feladatokban a Claude Opus 5.5 továbbra is pontosabb (több interakciót vagy iránymutitást igényelhet a Deepseek).



Időzítés és Árazás

Csúcsidő vs. Csúcsidőn kívüli (Off-peak) árazás. A nagy volumenű feladatokat érdemes csúcsidőn kívülre időzíteni, mivel olyankor az API költség pontosan a fele.

Gemini Notebook

Összegzés: A Tökéletes Fejlesztői Környezet

```
  [ LeanREA (Helyes Parancsok / Nincs Pazarlás) ]
+ [ Deepseek V4.1 Flash (Okos / Olcsó / Cenzúrázatlan) ]
+ [ PandaCode (Minden-egyben UI) ]
```

= Verhetetlen Fejlesztői Hatékonyság (Aprópénzért)

A jelenlegi legoptimálisabb megoldás a reverse engineering feladatok automatizálására, megszüntetve a token-pazarlást és a modellszintű korlátozásokat. Felkészülve a jövő komplex feladataira.

Gemini Notebook