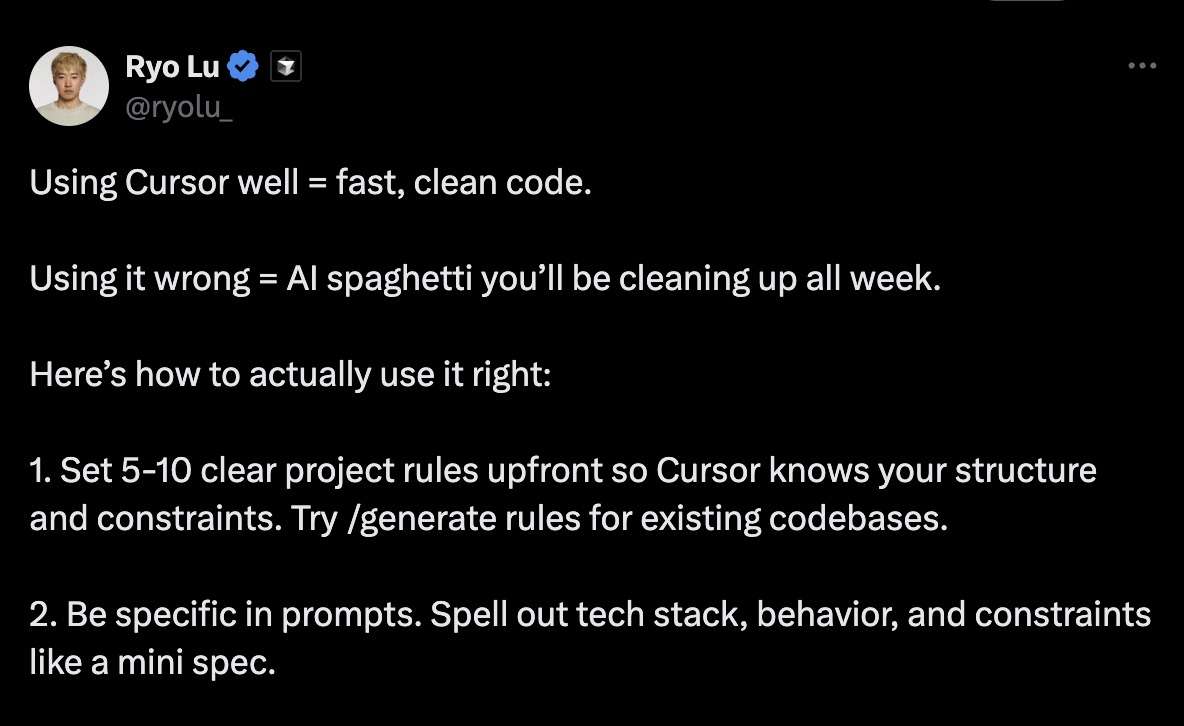
**Cursor官方下場談Cursor正確用法，以及我的實踐解讀**

<https://www.youtube.com/watch?v=iE2VkoTp3x8>



**Cursor官方下場談Cursor正確用法，以及我的實踐解讀**

[](https://www.zhihu.com/people/cai-lang-ri)

[**ai呀蔡蔡**](https://www.zhihu.com/people/cai-lang-ri)

讀書、碼字、聽播客、學AI，關注個人成長和健康

**[來自專欄 · AI程式設計](https://www.zhihu.com/column/c_1875212143512662016)**

536 人贊同了該文章

Cursor首席設計師 [Ryo Lu](https://zhida.zhihu.com/search?content_id=257125192&content_type=Article&match_order=1&q=Ryo+Lu&zd_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ6aGlkYV9zZXJ2ZXIiLCJleHAiOjE3NDc3MTQxNTMsInEiOiJSeW8gTHUiLCJ6aGlkYV9zb3VyY2UiOiJlbnRpdHkiLCJjb250ZW50X2lkIjoyNTcxMjUxOTIsImNvbnRlbnRfdHlwZSI6IkFydGljbGUiLCJtYXRjaF9vcmRlciI6MSwiemRfdG9rZW4iOm51bGx9.mrJ88KCB3zywcYsxqKMsnSk5bIhAFv5TsjnOCd4u3Fo&zhida_source=entity) 上周發佈了一條內容，**關於如何正確使用Cursor**，一共 12 條方法，可能部分小夥伴已經看過了。

不過沒關係。今天這篇文章，除了分享 Ryo Lu 提到的這 12 條方法，還會基於我的Cursor實踐對這些方法進行解讀（解讀內容會用灰字進行標注），這些方法同樣適用於Windsurf、Trae的開發實踐。

**正確使用 Cursor = 快速、乾淨的代碼。**

**使用錯誤 Cursor = 你將整個星期都在清理“**[**AI義大利面**](https://zhida.zhihu.com/search?content_id=257125192&content_type=Article&match_order=1&q=AI%E6%84%8F%E5%A4%A7%E5%88%A9%E9%9D%A2&zd_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ6aGlkYV9zZXJ2ZXIiLCJleHAiOjE3NDc3MTQxNTMsInEiOiJBSeaEj-Wkp-WIqemdoiIsInpoaWRhX3NvdXJjZSI6ImVudGl0eSIsImNvbnRlbnRfaWQiOjI1NzEyNTE5MiwiY29udGVudF90eXBlIjoiQXJ0aWNsZSIsIm1hdGNoX29yZGVyIjoxLCJ6ZF90b2tlbiI6bnVsbH0.vPXkXBCSGKoyZgd9JXqpeDV2fDe4NHJiRAkss4X7wHI&zhida_source=entity)**”（即亂糟糟的代碼）。**

**這就是正確使用它的方法：**

**1. 在一開始就設定 5-10 條清晰的[專案規則](https://zhida.zhihu.com/search?content_id=257125192&content_type=Article&match_order=1&q=%E9%A1%B9%E7%9B%AE%E8%A7%84%E5%88%99&zd_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ6aGlkYV9zZXJ2ZXIiLCJleHAiOjE3NDc3MTQxNTMsInEiOiLpobnnm67op4TliJkiLCJ6aGlkYV9zb3VyY2UiOiJlbnRpdHkiLCJjb250ZW50X2lkIjoyNTcxMjUxOTIsImNvbnRlbnRfdHlwZSI6IkFydGljbGUiLCJtYXRjaF9vcmRlciI6MSwiemRfdG9rZW4iOm51bGx9.XfaYF8Q8EA1nv42NVWom68vYiyOh4Qcs5o75sOo3n4M&zhida_source=entity" \t "_blank)，讓 Cursor 瞭解你的結構和約束。嘗試為現有代碼庫執行/generate rules。**

設定專案規則很重要，我之前對比過設定了專案規則，以及沒有設定專案規則去推進同一個專案，後者會明顯更吃力，因為代碼生成更不可控。

為什麼是 5-10 條專案規則？如果大家之前看過這篇文章 [Cursor Rules在實際開發中的三種層級&實際應用（附20個常用Rules）](https://link.zhihu.com/?target=https%3A//mp.weixin.qq.com/s%3F__biz%3DMzIyMzk3MTEwNQ%3D%3D%26mid%3D2247485919%26idx%3D1%26sn%3D9ec1ed6066effc793797d89de768eaee%26scene%3D21%23wechat_redirect)，應該還記得：

隨著項目壯大，我們已經沒法用一條通用的.cursorrules來跑完整個專案，大家想要更精准地控制代碼生成，就需要更細緻的專案規則。在Cursor實際開發中，專案規則可以分為三個層級，即**通用規則、程式設計語言規則**和**框架規則**。根據專案類型、規模的不同，需要的專案規則數量也會有所區別。

如果是全新的專案，建議先從通用規則（對應rule type的Always）開始；

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

AI 產生的內容可能不正確。

也可以在完成專案腳手架後，在 Agent 模式輸入中執行/generate rules。

注：專案腳手架提供**專案初始結構**和**基礎配置**（如目錄、工具鏈、依賴等），它的核心是**“加速啟動”**

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

關於/generate rules（自動生成規則）功能，可以看這篇文章的詳細介紹：[Cursor v0.49.x更新：自動生成Rules、MCP上下文可添加圖像、新增5種強大程式設計模型](https://link.zhihu.com/?target=https%3A//mp.weixin.qq.com/s%3F__biz%3DMzIyMzk3MTEwNQ%3D%3D%26mid%3D2247485810%26idx%3D1%26sn%3De5f32a3608b4f738174309aaf503aa0b%26scene%3D21%23wechat_redirect" \t "_blank)

**2. 在提示中要具體。詳細說明技術棧、行為和約束，就像一個迷你規範。**  
  
先給大家看三個提示詞案例：

**案例1：**“web版”限定了這個貪吃蛇遊戲的技術棧範圍，其中“儘量不引入外部依賴”也降低了依賴的安裝，基本就是一個最簡單的HTML+CSS+JavaScript的技術棧組合

請幫我開發一個web版貪吃蛇遊戲，儘量不引入外部依賴

**案例2：**“Chrome流覽器外掛程式”類別限定了技術棧範圍，功能描述則做了進一步的約束

請幫我開發一個“圖片轉png”Chrome流覽器外掛程式，這個外掛程式的功能是：

1、開啟外掛程式後，使用者在流覽器圖片滑鼠右鍵後會出現外掛程式入口"下載為png"，點擊就可以把選中的任意格式的圖片下載為png格式的圖片；

2、支援下載為png的圖片格式包括JPG、JPEG、PNG、BMP、webp、svg；

3、使用OffscreenCanvas避免阻塞主執行緒；

4、通過Blob URL減少記憶體佔用。

**案例3：**這是指定特定 Page 進行 UI 優化，同時 UI 優化方向進行了約束

請優化 LoginPage 的 UI，使其符合現代簡約風格，

具體要求：

1、佈局：登錄表單居中，使用卡片式設計（圓角 8px，輕微陰影）

2、配色：主色調深藍 (#1E3A8A），背景淺灰（#F9FAFB）

3、字體：標題用 Poppins 加粗，正文用 Inter 常規

4、交互：輸入框聚焦時邊框變藍，按鈕懸停時輕微變深

5、參考風格：類似 Notion 的登錄頁面，但更簡潔一些。

整體功能架構保持不變

大家不難發現，每一輪的提示詞對話，其實都是在做一次小的規範，是在User Rules、.cursorrules、Project Rules之下的更細顆粒度的一個規範。

之前分享過10個場景的提示詞範本，包括：代碼生成、功能拓展、UI調整、代碼解釋、項目重構，代碼調試、代碼修復、測試用例、代碼翻譯、API調用，感興趣的小夥伴可以點擊查看>>>[和Cursor協作450+小時後，我整理了這10個常見場景的Cursor提示詞範本](https://link.zhihu.com/?target=https%3A//mp.weixin.qq.com/s%3F__biz%3DMzIyMzk3MTEwNQ%3D%3D%26mid%3D2247485550%26idx%3D1%26sn%3Dc92c333819d6dcd9a4cae0a054b97d42%26scene%3D21%23wechat_redirect)

**3. 按文件逐個工作；在小塊、集中的部分生成、測試和審查。**  
  
如果只是做功能簡單的小項目，那麼完全可以用Cursor一次性生成全部代碼，一般不會有什麼大問題。

但如果是做功能相較複雜的中大型項目，那麼一次性搞定基本是不可能的。之前看過這麼一個案例，就是團隊用AI IDE花了3-4個小時進行開發，結果後面程式完全沒法運行。

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

AI 產生的內容可能不正確。

這種案例並不少見，他們的問題都很相似，就是前期沒有做好專案和結構拆解，中期任由AI IDE自己去跑代碼也不做太多調試和審查，到了後期代碼就成了一團改不動也運行不了的 。

這個問題解決起來的核心就是功能模組化，然後按逐個模組解決。之前在知識星球其實也有提到：

一張含有 文字, 螢幕擷取畫面, 文件, 字型 的圖片

AI 產生的內容可能不正確。

而關於前期專案和代碼結構拆解，之前其實也有一期主題分享，感興趣的可以移步查看>>>[Cursor專案後期維護全是坑？前期一定要做好專案和代碼結構拆解](https://link.zhihu.com/?target=https%3A//mp.weixin.qq.com/s%3F__biz%3DMzIyMzk3MTEwNQ%3D%3D%26mid%3D2247485753%26idx%3D1%26sn%3D21273e06f00875762dae6081eedf3bb7%26scene%3D21%23wechat_redirect)

**4. 先寫測試，鎖定它們，然後生成代碼，直到所有測試通過。**  
  
由於 Agent 可能會修改測試檔去讓測試通過，所以在測試前，建議使用 [.cursorignore](https://zhida.zhihu.com/search?content_id=257125192&content_type=Article&match_order=1&q=.cursorignore&zd_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ6aGlkYV9zZXJ2ZXIiLCJleHAiOjE3NDc3MTQxNTMsInEiOiIuY3Vyc29yaWdub3JlIiwiemhpZGFfc291cmNlIjoiZW50aXR5IiwiY29udGVudF9pZCI6MjU3MTI1MTkyLCJjb250ZW50X3R5cGUiOiJBcnRpY2xlIiwibWF0Y2hfb3JkZXIiOjEsInpkX3Rva2VuIjpudWxsfQ.akL8S5Pc1VZBPgMhnvTZZXkVnW5aAUSd2AGmc8JzTDE&zhida_source=entity) 功能去防止 Agent 修改測試檔。

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

AI 產生的內容可能不正確。

.cursorignore 可以在【Cursor Settings-Features-Hierachical Cursor Ignore】中開啟，這個功能主要用於控制 .cursorignore 檔的作用範圍。

注：類似.gitignore，用於指定哪些檔和目錄應該被 Cursor 忽略（也就不會被Cursor的智慧功能如代碼補全、檔搜索等處理）。

當啟用【Hierachical Cursor Ignore】選項後，一個目錄中的 .cursorignore 檔的規則將會應用到其所有子目錄，大家可以在專案根目錄或任何子目錄創建 .cursorignore 文件，然後在文件中列出想要忽略的檔模式，比如:

node\_modules/

\*.log

.DS\_Store

build/

dist/

對於大型專案，建議啟用此選項以便更好地管理忽略規則，可以在不同層級的目錄放置 .cursorignore 檔來實現更細細微性的控制。

注意：如果修改了此設置，需要重啟 Cursor 才能生效。

**5. 總是審查 AI 輸出，並對任何錯誤的內容進行硬修復，然後告訴 Cursor 將其作為示例使用。**

Cursor 每次給出的代碼並不完全都是正確的或符合要求的，所以需要我們進行人工審查。

舉個最簡單的例子，比如你所在企業對代碼注釋有一定規範，但Cursor每次生成都和這種企業規範有一定差距，這時候你就需要自己按照企業代碼注釋規範進行調整。

當然你不用每次都手動調整，最方便的方法就是將這種規範作為示例喂給Cursor，這樣Cursor就能學會這種示例並應用到下次程式設計任務中。

**6. 使用@file、@folder、@git，將 Cursor 的關注點定位到代碼庫的正確部分。**

這個功能一般會搭配第三點“按模組推進開發”使用，因為隨著專案代碼越來越多，我們不可能每次都遍歷專案檔案去做內容調整，一方面受上下文限制，另一方面遍歷代碼往往會導致Cursor改動一些我們並不希望它改動的代碼。

所以要善用@file、@folder、@git 這些上下文功能，它們相當於一個定位工具，實現指哪打哪。

**7. 將設計文檔和檢查清單保存在 .cursor/ 中，以便 Agent 能夠全面瞭解下一步要做什麼。**

.cursor/ 目錄一般包含專案規則（即第一點提到的rules）以及的代碼索引（比如這裡提到設計文檔、檢查清單等），都是用於加速 AI 的代碼分析和上下文檢索。

如果從實現目的看，設計文檔和檢查清單都是為了讓 Agent 更全面瞭解當前專案進度和下一步計畫，那檔不一定要保存在 .cursor/ 目錄下。

在此之前，我們的一般實踐可能記錄在README.md檔中，比如：

./

└── docs/

├── RAEDME-design.md

├── RAEDME-product.md

每次調整都及時更新這些README.md檔即可，和放在 .cursor/ 目錄下實現的效果都是一樣的。

**8. 如果代碼錯誤，就自己寫。Cursor 從編輯中學習比從解釋中學習更快。**  
  
我個人認為這一點和第五點有點類似。

有時候Cursor會在某個問題上反復打轉，這時候建議自己動手去做一些代碼調整，你的編碼內容會成為Cursor自我學習的上下文（相當於成為一種示例），這樣Cursor會更清楚自己接下來要做什麼。

不過這一點對於零程式設計基礎的用戶來說，會比較難。

所以我之前在文章評論區和知識星球都反復提到一個觀點，**你想在AI程式設計上走得遠，最好懂一點代碼知識**。你不需要先掌握某種程式設計語言再入門AI程式設計，而是在學習AI程式設計的過程中，在一個個具體的開發場景中，去補齊自己的程式設計知識。

**9. 使用聊天歷史來反覆運算舊的提示，而不需要從頭開始。**

之前有小報童用戶問我，

“提了個問題後AI輸出一半時發現思路跑歪了，這時候要不要stop並繼續追加對話？還是不用stop直接進行新的提問？還是stop並重新編輯這個問題進行提問？”

我當時的回答是，“可以繼續讓它跑下去，根據它的回答再二次追問，調整。”

一張含有 文字, 螢幕擷取畫面, 字型, 設計 的圖片

AI 產生的內容可能不正確。

在我看來，我們和Cursor（或任何一個AI）的每一次對話其實都是有用的，無論這次對話 Cursor 給到的回答是否滿意。

滿意固然很好，不滿意的話至少我們知道自己不滿意的點在哪，這樣下次對話就有了調整的方向。所有的聊天歷史，都可以成為我們優化提示詞的上下文。

**10. 有意選擇模型。Gemini 用於精確，Claude 用於廣度。**

我之前分享過一篇文章[【經驗分享】你在AI程式設計中可能會犯的9個錯誤](https://link.zhihu.com/?target=https%3A//mp.weixin.qq.com/s%3F__biz%3DMzIyMzk3MTEwNQ%3D%3D%26mid%3D2247485666%26idx%3D1%26sn%3Db184f8544df8b1715f6e8f93a2fadb17%26scene%3D21%23wechat_redirect)，其中有一個常見使用誤區，**就是用 1 個 AI 模型解決所有事情**。

與之相對的，正確的使用方法，應該是根據不同場景使用不同的模型。當然，每個人的編碼體驗可能有所差別：

在@Cj Z看來：在Cursor/Windsurf：

* 使用 Claude sonnet 3.5 進行編碼（是的，對於執行代碼來說，它比 3.7 更好。）
* 使用 GPT o1/o3-mini-high 調試複雜錯誤。
* 使用 Gemini Flash 2.0 掃描整個代碼庫並更新文檔。

在@Ryo Lu看來：模型是獨一無二的，就像人類一樣，在 Cursor 中感受它們：

* ‍ Gemini 2.5：高級軟體工程師 1，需要推動
* ‍ Claude 3.7：過度思考者，熱衷於使用工具，需要馴服
* ‍ Claude 3.5：全能高手，依然如此
* ‍ GPT-4.1 / o3：開始意識到程式設計不僅僅是關於基準測試

我之前也在知識星球分享過類似的觀點，Claude 3.7 thinking適合做規劃，Claude 3.7因為喜歡“畫蛇添足”所以適合腦補最佳實踐的場景，Claude 3.5能比較好遵循指令。

至於Gemini 2.5，它現在是我用Claude解決不了難題時一定會去使用的模型，一個被大家低估的程式設計模型。

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

**11. 在新的或不熟悉的技術棧中，粘貼文檔連結。讓 Cursor 逐行解釋所有錯誤和修復。**

要是放在以前，對於新的或不熟悉的技術棧，粘貼文檔連結確實是個不錯的方法，它其實就是使用@Web 的聯網功能，去讀取相應的文檔連結。

怎麼獲取這些最新的文檔連結呢？可以考慮大家可以考慮使用 [Context7](https://zhida.zhihu.com/search?content_id=257125192&content_type=Article&match_order=1&q=Context7&zd_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ6aGlkYV9zZXJ2ZXIiLCJleHAiOjE3NDc3MTQxNTMsInEiOiJDb250ZXh0NyIsInpoaWRhX3NvdXJjZSI6ImVudGl0eSIsImNvbnRlbnRfaWQiOjI1NzEyNTE5MiwiY29udGVudF90eXBlIjoiQXJ0aWNsZSIsIm1hdGNoX29yZGVyIjoxLCJ6ZF90b2tlbiI6bnVsbH0.GI9s0LhbOSt1bwKOErKZ2XI7alamZHhi_boco0Vp8VA&zhida_source=entity)。

：[https://context7.com/](https://link.zhihu.com/?target=https%3A//context7.com/)

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

你可以在context7上獲取最新文檔連結，然後粘貼到Cursor進行中代碼修復。

一張含有 文字, 電子產品, 多媒體, 軟體 的圖片

AI 產生的內容可能不正確。

當然，你也可以使用 Context7 MCP，讓Cursor直接調用最新的文檔，進而降低幻覺。

：[https://github.com/upstash/context7-mcp/tree/master](https://link.zhihu.com/?target=https%3A//github.com/upstash/context7-mcp/tree/master)

它的使用方法也很簡單，就是在成功安裝 Context7 MCP後，每次在對話中後加入 “使用 context7”（或類似的表達，如“使用 context7 獲取文檔”）即可：

新的 Next.js after() 函數是如何工作的？使用 context7

**12. 讓大型專案整夜索引，並限制上下文範圍以保持性能敏捷。**

Cursor有個 Codebase Indexing（代碼庫索引）功能。

索引是一個耗時且資源密集的過程，在處理大型專案時（如包含大量檔、資料或內容的系統）可能需要數小時甚至更久。在夜間運行索引任務，是因為夜間系統負載較低，使用者活動較少，計算資源更充足，可以避免影響白天的工作效率。

一張含有 文字, 螢幕擷取畫面, 軟體, 作業系統 的圖片

AI 產生的內容可能不正確。

在【Codebase Indexing】這裡，是可以設置 Ignore files 的，這樣 Cursor 在索引項目目代碼時就會忽略這些問題，這就是前面提到的通過限制上下文範圍，減少計算資源的消耗，進而降低記憶體和 CPU 的使用，從而提高索引或查詢的速度。

Ryo Lu 最後提到，**“結構和控制是勝利之道（目前如此）。將 Cursor 代理視為一位元強大的初級員工——如果你指明方向，它能夠快速走得很遠。”**

Cursor 12 種正確用法的介紹和實踐解讀就到這裡了，希望對大家有啟發。