

المحطة الثانية

# هندسة الأوامر المتقدمة

Advanced Prompting for Code

صياغة الأوامر البرمجية الاحترافية لانتزاع أفضل الأكواد وأكثرها أماناً واستقراراً من نماذج الذكاء الاصطناعي.

---

# القسم الأول

تقنيات صياغة الأوامر البرمجية (Prompting Techniques)

كيف تختار الاستراتيجية البرمجية المناسبة لحجم وتعقيد مهمتك؟

# تقنية Zero-Shot Prompting

Cyberpunk glowing neon code representing direct AI programming instruction

**المفهوم الأساسي:** طلب الكود مباشرة من النموذج الذكي دون إرفاق أي أمثلة مسبقة للحل.

**متى تُستخدم؟** في المهام البرمجية القياسية، كتابة الدوال البسيطة، أو توليد أكواد شائعة.

**الميزة الكبرى:** سرعة التطبيق ولا تستهلك مساحة كبيرة من نافذة السياق (Context Window).

**الحدود:** قد لا يلتزم النموذج بالأسلوب الدقيق لهندسة مشروعك الخاص.

# تقنية Few-Shot Prompting

**المفهوم الأساسي:** تزويد النموذج بمثال أو عدة أمثلة (مدخلات ومخرجات متوقعة) قبل طلب الكود الأساسي.

**متى تُستخدم؟** عند الحاجة لاتباع نمط تصميم معين (Design Pattern)، أو الالتزام بمعايير كتابة خاصة بشركتك.

**الميزة الكبرى:** تضمن دقة المخرجات بنسبة هائلة ومطابقتها التامة لمعاييرك البرمجية والبيانية المعقدة.

**مثالي لـ:** معالجة هياكل البيانات غير المألوفة وتحويل التنسيقات الصعبة بشكل آمن.

## ⚡ تأثير الأمثلة

الأمثلة البرمجية تعمل كقناة توجيه مغناطيسية، تمنع النموذج من الخروج عن حدود الصياغة المنطقية المستهدفة وتوفر فحصاً ذاتياً للمخرجات.

# مقارنة بصرية: Zero-Shot ضد Few-Shot

وجه المقارنة	تقنية Zero-Shot Prompting	تقنية Few-Shot Prompting
الاستهلاك (Tokens)	اقتصادي وموفر جداً للذاكرة المؤقتة.	يستهلك كمية أكبر بسبب تضمين الأمثلة.
تعقيد المهمة برمجياً	للمهام القياسية والمباشرة (مثال: دالة فرز).	للمهام المخصصة والمعمارية الهيكلية المعقدة.
الالتزام بالنمط (Pattern Style)	منخفض؛ يعتمد النموذج على أسلوبه الافتراضي.	عالي جداً؛ ينسخ النموذج نمط الكود المرفق بدقة.

---

# القسم الثاني

معمارية بناء الأوامر الخلفية (System Prompts)

كيف تضع الدستور والقواعد الحاكمة لسلوك النموذج وتصرفاته الذكية؟

# هيكل الـ System Prompt الاحترافي



## 3. أسلوب المخرجات (Output)

### (Format)

إجبار النموذج على نمط محدد: "أريد الكود البرمجي النقي فقط دون مقدمات أو شروحات نصية طويلة، مع تعليقات مقتضبة لشرح المنطق الصعب".



## 2. القيود والأمان

### (Constraints)

فرض معايير أمان صارمة مثل: "لا تستخدم مكتبات خارجية غير مستقرة، وامنع الثغرات الأمنية الشائعة مثل SQL Injection".



## 1. تحديد الدور (Role)

صياغة هوية واضحة للنموذج لرفع مستوى أدائه، على سبيل المثال: "أنت مهندس برمجيات محترف (Senior Tech Lead) وخبير في لغة X".

# تخصيص الـ System Prompt لكل لغة

## لغات الـ Dynamic Type (Python / JavaScript)

التركيز في أمر النظام على كفاءة الأداء ونقاء الكود المكتوب لمنع أخطاء وقت التشغيل.

التركيز على مقروئية الكود ومبادئ Clean Code.

تأمين المداخل والمخارج لمنع Runtime Exceptions.

## لغات الـ Static Type (Go / TypeScript / Java)

التركيز في أمر النظام على فرض القواعد الصارمة لبيئة التطوير وعملية البناء كـ `컴파일`.

الالتزام التام بتعريف الأنواع (Strict Typing).

معالجة الأخطاء (Error Handling) بشكل فوري ومنع الـ `null`.

# ورشة عمل تطبيقية (مثال عملي لأمر نظام)

## تحليل الهيكل البرمجي للأمر

يظهر هذا التحديد الهيكل الصارم كيف يجبر المطور نموذج الـ AI على تفادي الهلوسة والالتزام بالأمان.

**النتيجة العملية:** كود خالي من الثغرات، سريع المراجعة، مهياً  
مباشرة لإعادة الاستخدام والدمج بمستودع الأكواد.

```
# SYSTEM PROMPT DEFINITION role: "Senior Python Developer" rules: - "Output ONLY executable code block" - "Do not write conversational text" - "Use clean docstrings & type hinting" - "Enforce strict exception handling" security: - "Ensure no hardcoded API keys" - "Sanitize inputs before file processing"
```



# اكتملت المحطة الثانية بنجاح!

هندسة الأوامر ليست مجرد كلمات وتخمين، بل هي تحديد فني دقيق للقيود، المعايير، والتعليمات الصارمة.

المحطة القادمة: هندسة الأكواد وتنظيفها مع الـ AI (Code Refactoring & Optimization)

# Image Sources

[https://images.stockcake.com/public/5/6/4/5649a43a-1033-4dc8-a3df-dbcdf74bd211\\_large/neon-hacker-paradise-stockcake.jpg](https://images.stockcake.com/public/5/6/4/5649a43a-1033-4dc8-a3df-dbcdf74bd211_large/neon-hacker-paradise-stockcake.jpg)

Source: [stockcake.com](https://stockcake.com)



---

[https://png.pngtree.com/thumb\\_back/fh260/background/20221205/pngtree-d-podium-abstract-blue-circles-hud-screen-system-technology-futuristic-innovation-image\\_1480441.jpg](https://png.pngtree.com/thumb_back/fh260/background/20221205/pngtree-d-podium-abstract-blue-circles-hud-screen-system-technology-futuristic-innovation-image_1480441.jpg)

Source: [pngtree.com](https://pngtree.com)

