

Claude

دليل استخدام Claude

كل شيء في مكان واحد

د.م. حماده بساط

استشاري التحول الرقمي والذكاء الاصطناعي

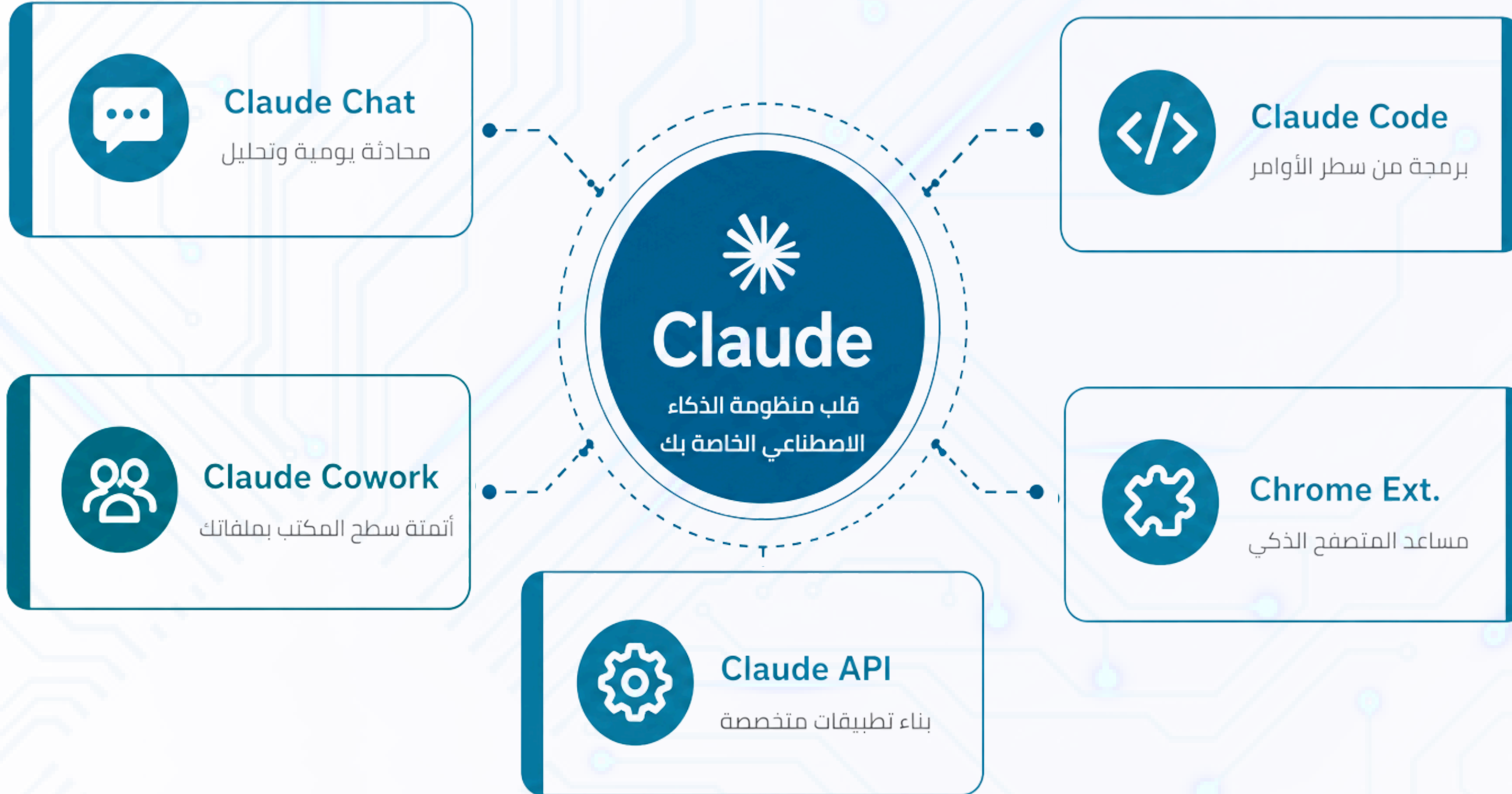
@albosat — www.al-bosat.com

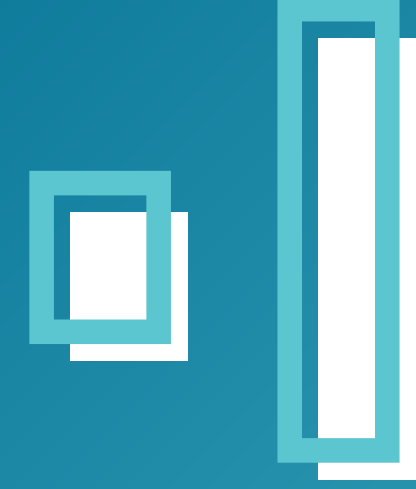
من بعض اقتباسات
م.عبدالرحمن سليم



المقدمة

خريطة منظومة Claude الكاملة





أساسيات Claude



أساسيات Claude

ما ستكتسبه في هذا الجزء

فلسفة Claude وبنيته

فهم ما الذي يجعله مختلفاً جوهرياً عن أي أداة اصطناعي أخرى

مقارنة عملية وصادقة

بين Claude و ChatGPT بمعايير واضحة تخدم قرارك – بلا تضليل

اختيار خطة الاشتراك

التي تناسب عملك فعلاً

بوابات الوصول

فهم كل بوابة والفرق الحقيقي بين كل أداة في المنظومة



Claude ليس أداة محادثة – إنه منصة تنفيذ



الحقيقة: "Claude = منصة تنفيذ ذكية متكاملة"

- يتذكرك ويتعلم أسلوبك عبر الوقت
- يقرأ ملفاتك المحلية وينقذ مهاماً حقيقية
- يتكامل مع Gmail و Drive و Notion وغيرها
- يعمل في الخلفية بلا إشراف مستمر



المفهوم الشائع الخاطئ: "Claude = Chatbot للأسئلة والأجوبة"

- أداة بسيطة تُجيب على أسئلتك فقط
- تحتاج تكرار كل معلومة في كل محادثة
- لا تتذكر من أنت أو كيف تعمل
- لا تستطيع إنجاز مهام حقيقية بملفاتك

مبني على **Constitutional AI** من Anthropic – الذكاء الاصطناعي الآمن والمسؤول



الفروق الجوهرية

Claude **VS** ChatGPT

ChatGPT (OpenAI)

التركيز على النمو السريع والقدرات العامة

يميل للموافقة المفرطة وإرضاء المستخدم

تعتمد على فلاتر خارجية (RLHF)

قد يهلوس بثقة أو يعطي أعذاراً عامة

يعتمد بشكل كثيف على التفضيلات البشرية

مساعد شخصي للمهام اليومية والعصف الذهني

Claude (Anthropic)

ذكاء اصطناعي آمن ومسؤول (Constitutional AI)

صريح – قد يرفض الأفكار الضعيفة بلطف

مدمجة في النواة (Constitutional AI)

يوضح بدقة حدود معرفته وأسباب رفضه

موجّه بمبادئ أخلاقية صارمة لتقليل التحيز

مستشار للتحليل المعقد والكتابة والبرمجة

المعيار

الفلسفة

الإجابات

السلامة

الشفافية

التدريب

الاستخدام الأمثل

ChatGPT كالمساعد الذي يقول "نعم سيدي" لكل شيء. Claude كالمستشار الذي يحترمك كفاية ليقول لك الحقيقة.



خطط الاشتراك – أيها يناسب عملك؟

مجاني – Free

للتجربة فقط

- Chat محدود جداً
- لا مشاريع
- لا ذاكرة متقدمة

★ شهر/\$20 – Pro

الأنسب للأغلبية

- Chat غير محدود
- Projects + Memory
- Styles مخصصة

شهر/\$100-200 – Max

الأداء الأقصى

- كل شيء في Pro
- Claude Cowork
- رصيد أعلى بكثير

مستخدم/\$30 – Team

للفرق

- إدارة مركزية
- بيانات آمنة
- مشاريع مشتركة

تفاوضي – Enterprise

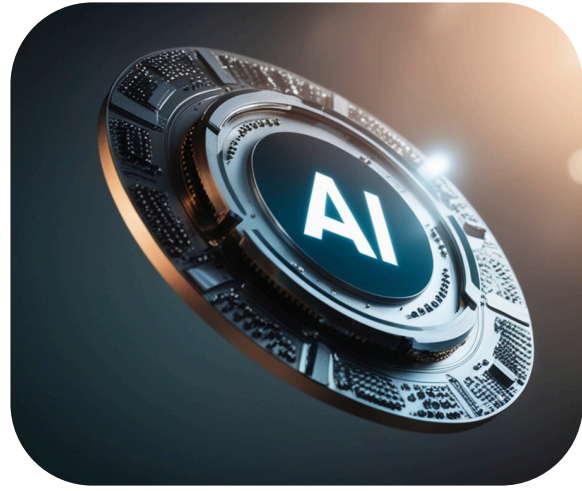
للمؤسسات الكبرى

- SLA مخصصة
- أمان مؤسسي
- تدريب مخصص

القرار العملي: ابدأ ب Pro لشهر كامل قبل القفز ل Max – معظم الناس يكتشفون أن Pro كافٍ تماماً.



بوابات الوصول ل Claude J – كل واحدة لها دورها



claude.ai

للمحادثات السريعة، والبدء، ومن أي جهاز. لا تثبيت، لا إعداد.



Desktop App

للعمل الجدي: ملفات محلية، مهام Cowork، يبقى في ال Dock جاهزاً.



Mobile App

تبدأ فكرة في الطريق وتنتهيها على الحاسوب. لإعطاء مهام ل Claude بعيداً عن المكتب.



Chrome Extension

يقرأ الصفحات، يلخص YouTube، يتفاعل مع Gmail وأي موقع تفتحه.



Desktop App **VS** Chrome Extension

وجه المقارنة	التطبيق المكتبي (Desktop)	إضافة المتصفح (Chrome)
البيئة	تطبيق مستقل (macOS / Windows)	داخل نافذة متصفح Chrome
الملفات المحلية	وصول كامل (بإذن المستخدم)	لا يمكنه الوصول للملفات المحلية
الأتمتة	مهام طويلة ومعقدة – يعمل بالخلفية	مهام فورية وسريعة على الصفحة
الاستخدام الأمثل	إدارة المشاريع – تحليل البيانات – Code	النماذج – Gmail – تلخيص المقالات
المتطلبات	خطة مدفوعة + مساحة (~2GB VM)	حساب Claude (يفضل خطة Pro)

السيناريو الأقوى: الإضافة تجمع البيانات من الويب – والمكتبي (Cowork) يحلها ويكتب التقرير.



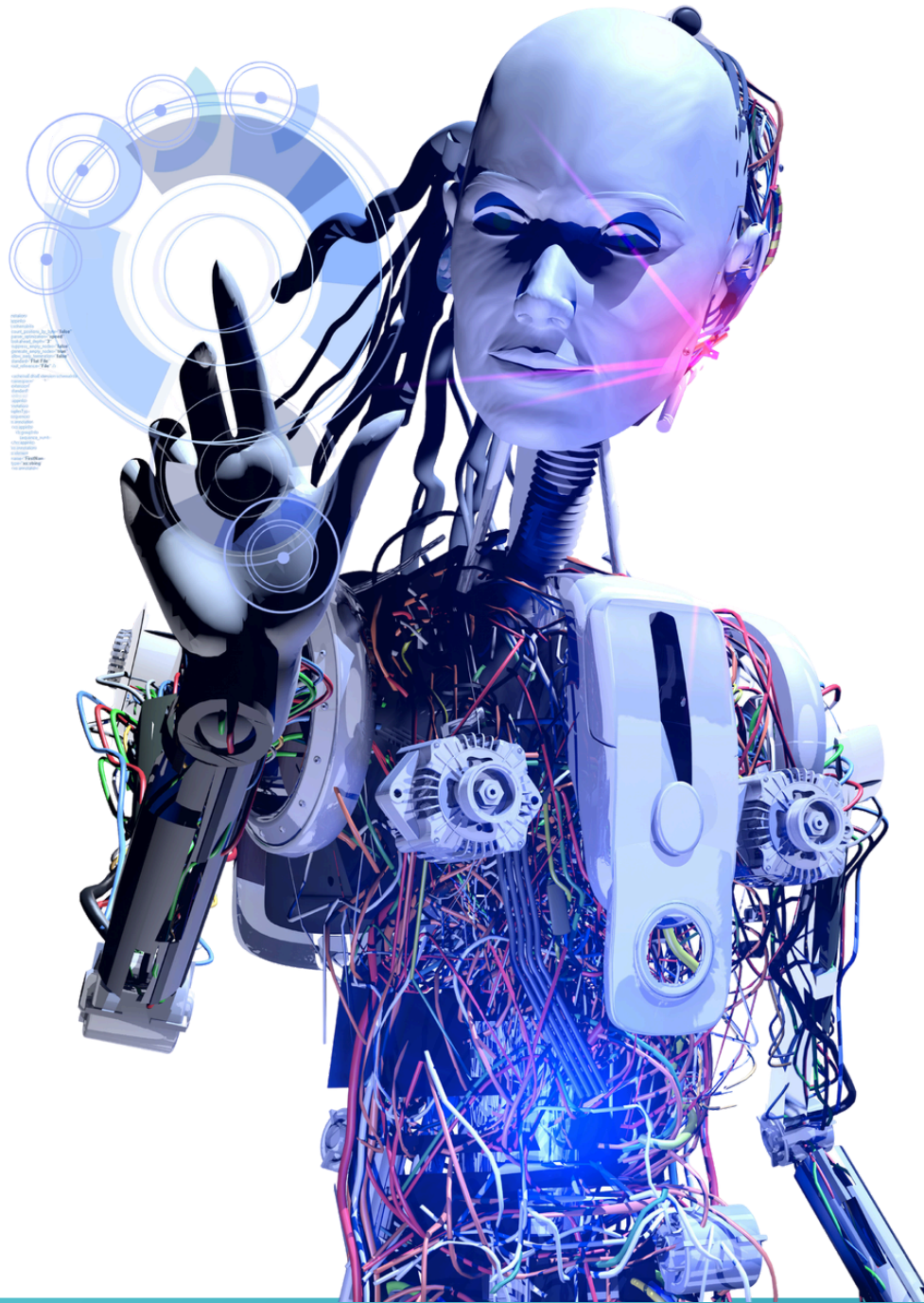
Chat – أكثر مما تتوقع

6 قدرات أساسية تُحوّل Chat من مجرد محادثة إلى أداة عمل قوية

- 01 بحث الويب المدمج**
يبحث تلقائياً عند الحاجة لمعلومات حديثة: أخبار اليوم، أسعار الأسهم، آخر الإصدارات
- 02 ذاكرة عبر المحادثات**
يتذكر من أنت وكيف تعمل بين المحادثات المختلفة تلقائياً
- 03 مخرجات تفاعلية**
توليد تطبيقات صغيرة، جداول تفاعلية، مخططات بيانية في نافذة جانبية
- 04 رؤية الصور والملفات**
ارفع صورة، PowerPoint، Excel، PDF – يقرأه ويحلله ويجيب عن أسئلتك
- 05 مشاريع منظمة**
تنظيم المحادثات مع تعليمات ثابتة وملفات مرجعية خاصة بكل مشروع
- 06 المحادثة الصوتية**
تحدث مباشرة مع Claude عبر تطبيق الجوال للحصول على إجابات سريعة



Cowork – المساعد الذي يُنفذ لا فقط يُجيب



⚠ تحذير مهم: Cowork يعدّل ملفاتك الحقيقية. خذ نسخة احتياطية وراجع خطته قبل التنفيذ.

قراءة وكتابة الملفات المحلية →

جدولة المهام المتكررة تلقائياً →

تصفح وتنظيم المجلدات →

بناء التقارير والتحليلات →



Cowork – المساعد الذي يُنفَّذ لا فقط يُجيب

تطبيقات عملية



Code – لماذا يهتمك حتى لو لم تكن مطوراً؟

2

لماذا في دليل للمبتدئين؟

لأن Cowork مبني على نفس التقنية – فهمك ل Code يجعلك تدرك إمكانيات Cowork الحقيقية

1

لمن هو (المطورون)

أداة سطر أوامر تتيح ل Claude J قراءة وكتابة وتعديل الكود مباشرة لبناء مشاريع كاملة

4

ملف CLAUDE.md

ملف إرشادي يُقرأ تلقائياً في كل جلسة – يحتوي أوامر مشروعك وقواعده وحدوده

3

الأوامر الأساسية

اكتب claude للبدء – /init لتهيئة المشروع – # للإشارة للملفات المحددة

\$ claude

\$ /init

Initialized Claude Code.

CLAUDE.md created.

Ready! Type your task:





نماذج Claude 2



نماذج Claude ما ستكتسبه

كيف تختار النموذج الصحيح لكل مهمة – وما الفرق الحقيقي في الأداء والتكلفة

فهم عائلة Claude 4.6: Opus, Sonnet, Haiku – متى تستخدم كل واحد

إطار قرار عملي يحدد النموذج المناسب لكل سيناريو من عمك الحقيقي

أوضاع التفكير المعمق (Extended Thinking) وكيف تؤثر على الجودة



عائلة نماذج Claude 4.6 – الصورة الكاملة

Opus 4.6

للتفكير العميق والعمل المعقد

الأعلى تكلفة

الأبطأ

الأعمق تفكيراً

Sonnet 4.6

الخيار الأمثل لمعظم المهام

متوسط

متوازن

متوازن الأداء

Haiku 4.5

السرعة والكفاءة

الأقل تكلفة

الأسرع

خفيف وسريع

العائلة 4.6 هي الأحدث حالياً. الأرقام ترمز للإصدار وليس للجودة المطلقة – كل نموذج مُحسَّن لسيناريوهات مختلفة.



Opus 4.6

للتفكير العميق والعمل المعقد

⚠️ الأبطأ والأعلى تكلفة – لا تستخدمه لأسئلة بسيطة
أو محادثات سريعة



نقاط القوة

- أعمق نموذج في التفكير والاستدلال المعقد
- مثالي للتحليل الاستراتيجي متعدد الأبعاد
- الأفضل في كتابة الكود المعقد وإعادة الهيكلة
- يتعامل مع السياقات الطويلة جداً بكفاءة

حالات الاستخدام المثالية

- تحليل عقود أو وثائق قانونية ومالية معقدة
- بناء استراتيجية شاملة أو خطة عمل متعددة السنوات
- كتابة كود متقدم أو تصحيح أخطاء معمقة



SONNET 4.6

الخيار الأمثل لمعظم المهام

نقاط القوة

- أفضل توازن بين الأداء والسرعة والتكلفة
- يؤدي 90% من مهام Opus بسرعة أعلى وتكلفة أقل
- مثالي لكتابة المحتوى والتحليل والبرمجة اليومية
- يعمل بسلاسة مع Cowork ومهام الأتمتة

حالات الاستخدام المثالية

- كتابة تقارير ومحتوى احترافي
- تحليل بيانات وملفات Excel و CSV
- البرمجة والتصحيح اليومي
- بناء Workflows في Cowork



HAIKU 4.5

السرعة والكفاءة عندما تحتاجهما

⚠️ **تجنبه** في مهام التفكير الاستراتيجي والتحليل المعقد

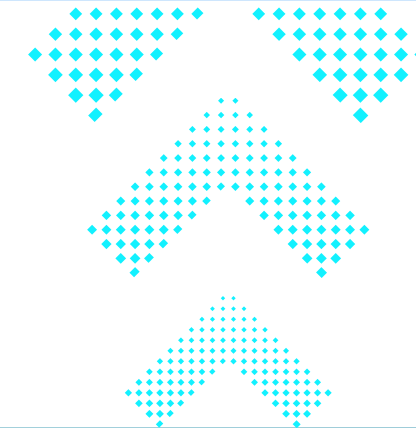
 **Haiku**
4.5

نقاط القوة

- الأسرع استجابة بين جميع النماذج
- الأقل تكلفة – مثالي للمهام التكرارية
- كافٍ تماماً للمهام البسيطة والمحادثات السريعة
- مناسب لبناء تطبيقات تحتاج ردوداً فورية

حالات الاستخدام المثالية

- ردود سريعة في محادثات العملاء
- تصنيف وفلتر كميات كبيرة من البيانات
- تطبيقات تحتاج استجابة فورية (low latency)



إطار اختيار النموذج المناسب

نوع المهمة	التعقيد	النموذج الموصى به	السبب
تحليل استراتيجي عميق	عالٍ جداً	Opus 4.6	يحتاج أعمق استدلال
كتابة تقرير تنفيذي	متوسط-عالٍ	Sonnet 4.6	جودة + سرعة معقولة
برمجة ودبغنة معقدة	عالٍ	Opus / Sonnet	حسب تعقيد الكود
تلخيص مقالات ووثائق	منخفض-متوسط	Sonnet 4.6	كافٍ وأسرع
أسئلة سريعة وبسيطة	منخفض	Haiku 4.5	لا داعي لنموذج ثقيل
تصنيف بيانات بكميات كبيرة	منخفض	Haiku 4.5	السرعة والتكلفة أولاً
بحث وتحليل تنافسي	متوسط-عالٍ	Sonnet 4.6	بحث + تحليل متوازن



سيناريوهات عمل حقيقية – أي نموذج تختار؟

Sonnet كتابة تقرير أداء

تحليل أرقام الشهر وإعداد ملخص تنفيذي واضح ومهني للفريق أو الإدارة

Sonnet عصف ذهني لحملة تسويقية

توليد أفكار إبداعية متعددة للحملة مع تقييم كل فكرة واقتراح الأنسب

Opus تحليل خطة توسع

تحليل المخاطر والفرص وصياغة توصيات استراتيجية شاملة لمجلس الإدارة

Haiku ردود سريعة على العملاء

تصنيف الاستفسارات وكتابة ردود احترافية قياسية بسرعة وكفاءة

Sonnet بحث تنافسي

تحليل معلومات المنافسين من الويب واستخلاص الرؤى والفرص السوقية



أوضاع التفكير وعمق الاستدلال

Standard Thinking VS Extended Thinking

Extended Thinking وضع التفكير المعمّق

- يُفَعّل يدويًا عند الحاجة لاستدلال أعمق
- يقوم النموذج بالتفكير خطوة بخطوة قبل الإجابة
- أبطأ وأكثر استهلاكًا للموارد
- مناسب للمهام التي تتطلب تحليلًا معمقًا
- مثالي لـ:

- التحليلات الاستراتيجية
- المسائل متعددة الأبعاد
- القرارات المعقدة
- تقييم السيناريوهات
- حل المشكلات المركبة

Standard Thinking

الوضع الافتراضي

- مناسب لـ 90% من المهام اليومية
- استجابة سريعة وكافية للكتابة والتحليل المعتاد
- كفاءة تكلفة عالية (لا يستهلك موارد إضافية)
- مثالي لـ:

- الكتابة
- البحث السريع
- البرمجة العادية
- الردود اليومية
- التلخيص وإعادة الصياغة

القاعدة: ابدأ دائماً بـ Standard فعل Extended فقط عندما يخيب توقعاتك النموذج في مهمة تحليلية معقدة



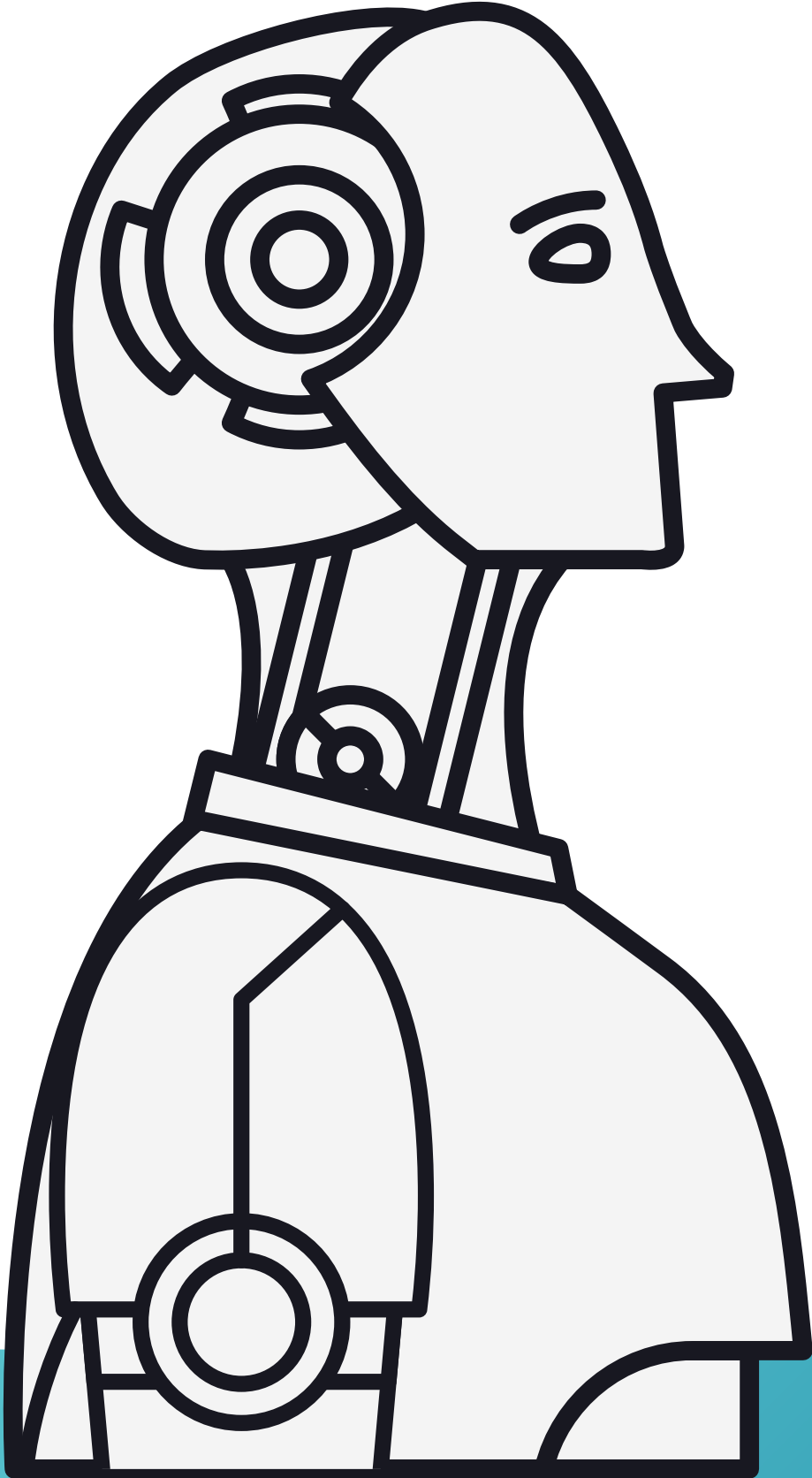


Prompt Engineering g Context Engineering



Prompt Engineering و Context Engineering

ما ستكتسبه

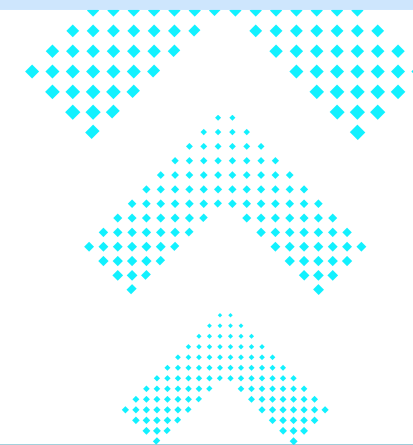


كيف تكتب أوامر تُنتج نتائج قوية من أول مرة – بلا تجربة وخطأ مستمر

إطار CRAFT العملي – 5 عناصر تضمن وضوح كل طلب

Context Engineering – الفرق الجوهرى عن Prompting وسبب أهميته

10 قواعد ذهبية جاهزة للطباعة تلخص كل هذا الجزء



الأساس الذي يغيّر كل شيء - Prompt Engineering

وضوح طلبك = جودة إجابتك

المثال أقوى من الوصف

أرفق مثالاً واحداً على الأسلوب
المطلوب بدلاً من وصفه بعشر جمل

السياق يسبق الطلب

أخبر Claude من أنت وما هدفك قبل أن
تطلب - يضاعف الدقة

الوضوح أقوى من الذكاء

واضح لنموذج متوسط يتفوق Prompt
غامض لأقوى نموذج Prompt على

ضعيف

"اكتب لي محتوى عن التسويق"

قوي

"أنت محرر محتوى متخصص في B2B. اكتب مقدمة لينكدان (150 كلمة)
تستهدف مديري التسويق في الشركات المتوسطة، تبرز خطر إهمال
SEO، بنبرة مباشرة بدون مديح"



إطار CRAFT – هيكل الـ [PROMPT] الذي يعمل فعلاً

5 عناصر تضمن وضوح كل طلب وتنتج مخرجات قوية من أول مرة

C – Context

من أنت؟ ما الموقف؟ ما الهدف؟
السياق يمنع سوء الفهم من البداية.



R – Role

ماذا تريد Claude أن يكون؟ خبير
تسويق؟ محلل بيانات؟ الدور يضبط
النبرة والعمق.



A – Action

ماذا تريده أن يفعل تحديداً؟ اكتب – حلّ
– قارن – لخص. فعل واحد واضح.

F – Format

كيف تريد المخرج؟ نقاط – جدول – فقرات – 200 – JSON كلمة.
حدد الشكل صراحةً.



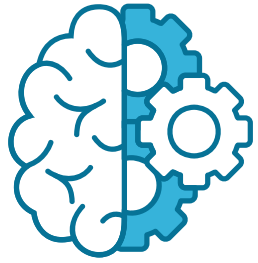
T – Tone

بأي أسلوب؟ رسمي – مباشر – ودود – تقني – مبسط. النبرة
تحدد الجمهور المستهدف.



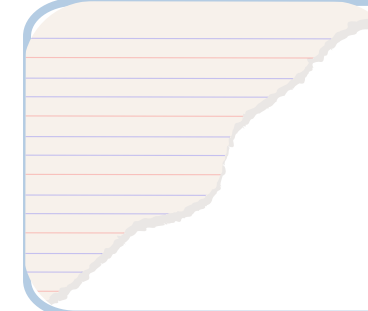
أنماط Prompting للمهام الحقيقية

6 أنماط جاهزة للاستخدام – كل نمط بمثال تطبيقي مباشر



التحليل والاستدلال

"حلّل [الموضوع] من منظور [الزاوية]. أبرز نقاط القوة والضعف وقدم توصية واحدة"



الكتابة والمحتوى

"اكتب [نوع المحتوى] لـ [الجمهور] بنبرة [أسلوب] – الهدف: [الفعل المطلوب]"



استخلاص البيانات

"من النص التالي، استخرج: [الحقل 1]، [الحقل 2]، [الحقل 3]. أخرجها في جدول"



الاستراتيجية والقرار

"قارن بين [خيار أ] و [خيار ب] باستخدام معايير: [م1]، [م2]، [م3]. أوص بخيار مُعلّل"



المراجعة والتحسين

"راجع هذا [النص/الكود]. ابحث عن: [م1]، [م2]. اقترح بدائل للمشاكل التي تجدها"



التلخيص الذكي

"لخص [المصدر] في [عدد] نقاط. الجمهور: [من]. أبرز ما يتعلق بـ [موضوع محدد]"



أخطاء ال Prompting الشائعة – تجنبها من البداية

أخطاء يقع فيها الجميع – مع البديل الصحيح لكل منها

✗ افتراض معرفة السياق

...بناءً على ما تعرفه عن شركتنا



البديل

قدّم سياقك كاملاً أو تّبته في User Preferences

✗ طلب عدة مهام في طلب واحد

اكتب خطة استراتيجية + محتوى + تصميم



البديل

قسّم المهمة: الخطة ثم المحتوى ثم توجيهات التصميم – كل طلب مستقل

✗ أوامر قصيرة بلا سياق

اكتب عن القيادة



البديل

اكتب مقدمة (3 جمل) لمنشور عن القيادة في الأزمات، جمهور مديرون عرب، نبرة مباشرة

القاعدة الذهبية: عندما لا يعجبك الجواب، لا تكرر السؤال – عدّل الطلب.



مضاعف الأداء الحقيقي – Context Engineering

تحسين صياغة طلب محدد داخل محادثة واحدة: Prompt Engineering

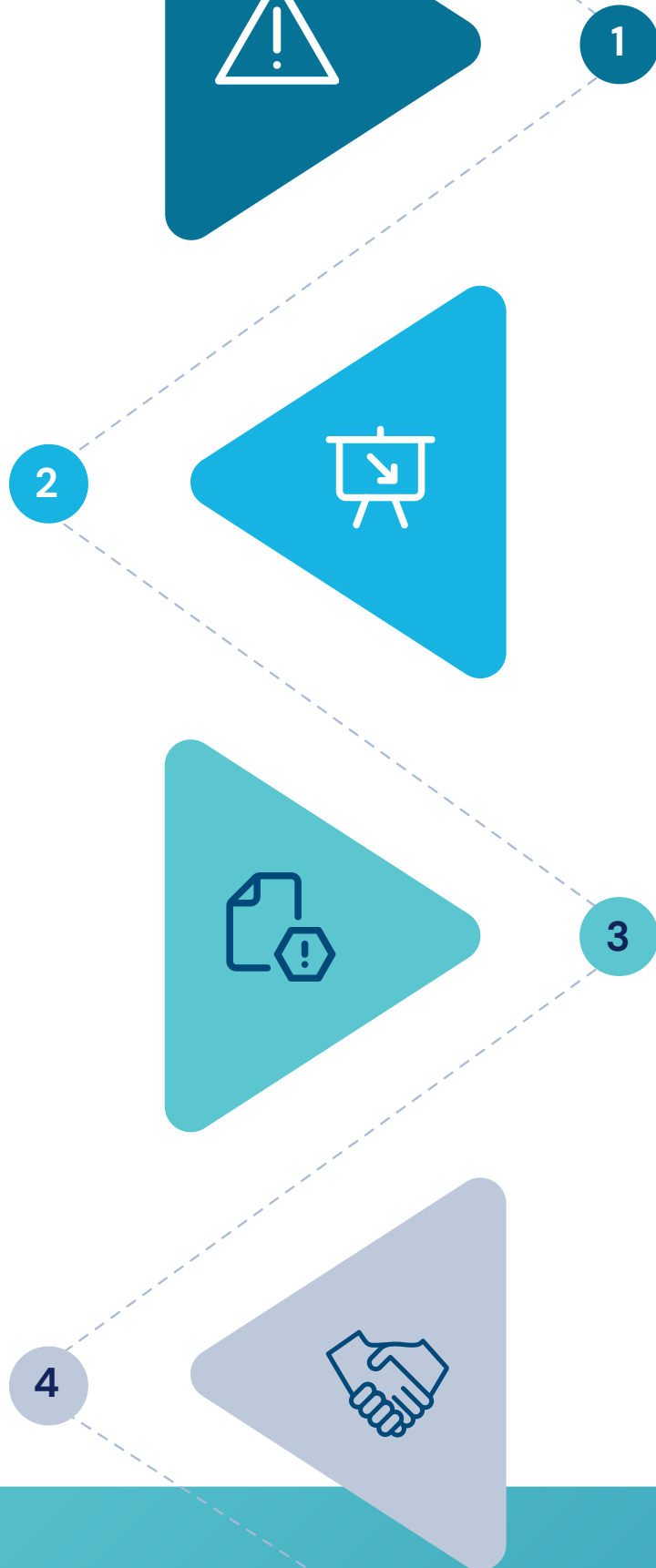
بناء إطار دائم حول كل محادثتك: Context Engineering

- User Preferences – هويتك الدائمة
- Project Instructions – قواعد كل مشروع
- ملفات مرجعية – معرفة ثابتة
- Memory – ذاكرة تتراكم

السر: يحوّل Claude إلى مساعد يعرفك ويوفر وقتاً هائلاً.



كيف تبني سياقاً متكاملًا لـ Claude



User Preferences

Project Instructions

Knowledge Base

Memory



التعامل مع السياقات الكبيرة والمدخلات الطويلة

ما هي نافذة السياق؟

Window هي الكمية الكلية من النص التي يستطيع Claude معالجتها في جلسة واحدة. كلما اقتربت من الحد قل دقة الإجابات على المحتوى القديم.



رفع الملفات واستخراج المعلومات

أفضل الممارسات

- كن محدداً: استخرج أرقام الإيرادات في الفصل الثالث فقط
- حدد المخرج: جدول بأعمدة البند والقيمة والصفحة
- اطلب التحقق: قل إن لم توجد المعلومة



أفضل استخدام	الصيغ	نوع الملف
تحليل العقود، التلخيص، الاستخراج تحليل الأرقام وبناء تقارير	PDF, DOCX, TXT, MD	وثائق نصية
مراجعة الشرائح واستخراج النقاط	PPTX	عروض
قراءة نصوص ورسوم بيانية	PNG, JPG, WEBP	صور
مراجعة وتصحيح وشرح	Python, JS, TXT	كود



التحكم في شكل المخرجات - Structured Outputs

اطلب ما تريده بالضبط - Claude ينفذ بدقة عندما تحدد الشكل

النقاط والقوائم

طلب محدد:

"ما فوائد التسويق بالمحتوى؟ أعطني قائمة من 5 نقاط، كل نقطة في جملة واحدة فقط."

طلب مبهم:

"ما فوائد التسويق بالمحتوى؟"

الجداول المنظمة

طلب محدد:

"قارن بين الأدوات الثلاثة في جدول يحتوي على الأعمدة التالية: الاسم، المزايا، العيوب، السعر، الأنسب لـ."

طلب مبهم:

"قارن بين الأدوات الثلاثة."

JSON وبيانات منظمة

طلب محدد:

"استخرج بيانات المنتجات بصيغة JSON وتتضمن: name, price, category, availability."

طلب مبهم:

"استخرج بيانات المنتجات."

التنسيق الواضح يوفر عليك جولة تعديل كاملة - حدد الشكل في الطلب ولا تنتظر أن يخمنه Claude



10 قواعد ذهبية للتعامل مع Claude

حدد التنسيق والطول مسبقاً

فعل واحد في كل طلب

ابدأ بالسياق قبل الطلب

استخدم Projects للسياق المتكرر

عدّل الطلب عند الخطأ

أرفق مثالاً للأسلوب

فعل Extended Thinking عند الحاجة

اطلب المراجعة الذاتية

قسّم المهام الكبيرة

ابن Context Engineering مرة





سير عمل Claude App

ما ستكتسبه

كيف تحوّل Claude من أداة محادثة إلى بيئة عمل احترافية متكاملة - خطوة بخطوة

إعداد Styles Memoryg User Preferences بشكل صحيح من أول يوم

Projects - مساحة عمل مستقلة لكل عميل أو مشروع بذاكرة ومعرفة خاصة

Artifacts والميزات المتقدمة: Scheduled Tasksg Skillsg Connectors



البداية الصحيحة داخل Claude App



لا تتخطَّ ضبط User Preferences – يوفر عليك إعادة الشرح.



هويتك الدائمة – User Preferences

اكتب تعليمات عملية لا سيرة ذاتية

- السياق: اللقب، الجمهور، الهدف
- الإرشادات: اللغة، الطول، النبرة
- القيود: ما تتجنبه

اطلب من Claude اقتراح Preferences مثالية لك.



User Memory
توجيهات يدوية دائمة

Session Memory
تستخلص تلقائياً تفضيلاتك

Project Memory
سياق معزول لكل مشروع

الطبقات – MEMORIES
الثلاث للذاكرة الذكية

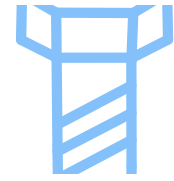


Styles – تحكم بنبرة وأسلوب الردود



Formal

رسمي وموضوعي للتقارير



Concise

قصير ومباشر للقرارات



Natural

حوار طبيعي للاستخدام اليومي



Explanatory

شرح تفصيلي للتعلم

ابن Style مخصص برفع 5-10 أمثلة لأسلوبك.



مخرجات تفاعلية قابلة للاستخدام Artifacts

لا يكتفي بالنص - يبني أدوات ومخرجات تفاعلية حية Claude

الكود والتطبيقات

JavaScript و Python
يكتب ويُنفذ كودًا مباشرة
يمكنك تعديله وتشغيله في نافذة
الجانبية

الجدول التفاعلية

بيانات منظمة قابلة للتصفية والتحرير
والتصدير مباشرة
جاهزة للتنزيل بصيغ: CSV و Excel

المخططات البيانية

رسوم بيانية تفاعلية لبياناتك:
بار، خطي، دائري
مرئية وقابلة للتعديل فورًا

المستندات الجاهزة

تقارير وعروض تقديمية قابلة للتنزيل
مستندات PDF و PPTX جاهزة
للاستخدام مباشرة

التطبيقات التفاعلية

حاسبات، لوحات تحكم، وأدوات تفاعلية
مصغرة
تعمل مباشرة داخل نافذة المحادثة

صفحات الويب

HTML كاملة مع CSS و JavaScript
جاهزة للنشر أو للمعاينة الفورية
موقع بسيط في دقائق

استخدم Artifacts عندما تريد مخرجاً قابلاً للتنزيل أو المشاركة أو البناء عليه.



Claude للبحث وجمع المعلومات

5 خطوات لبحث فعّال مع Claude – من صياغة الاستعلام إلى استخلاص الرؤى وتوثيق المصادر



استخلاص الرؤى

ملخص تنفيذي مع قائمة المصادر والاقتراسات الرئيسية – جاهز للمراجعة

التلخيص والمقارنة

اطلب مقارنة المصادر المختلفة وتحديد نقاط الاتفاق والاختلاف بين الآراء

تحليل المصادر

ارفع PDF أو روابط – يقرأ Claude ويستخلص المعلومات ذات الصلة مباشرة

Web Search المدمج

Claude يبحث تلقائياً في الويب ويستطيع تحليل نتائج متعددة في وقت واحد

صياغة الاستعلام

حدد موضوعك بدقة واطلب من Claude صياغة استعلامات بحث متعددة للوصول لأفضل النتائج

اجمع البحث الويب مع رفع الملفات في طلب واحد: 'ابحث في الويب + حلّل هذه الدراسة وقارن النتائج'



Claude للتقارير والتحليل

من البيانات الخام إلى تقرير جاهز – سير عمل تحليلي متكامل

المدخلات

- ملفات CSV/Excel
- PDF ووثائق نصية
- بيانات من نسخ لصق
- روابط مباشرة للمصادر

المعالجة

- تنظيف البيانات وتصنيفها
- الكشف عن الأنماط والاتجاهات
- مقارنة الفترات الزمنية
- استخلاص الإحصاءات الرئيسية

المخرجات

- تقرير تنفيذي منسق
- جداول ومخططات بيانية
- ملخص للقرار (1 صفحة)
- توصيات قابلة للتطبيق

مثال تطبيقي – تقرير أداء شهري:

1

الخطوة 1

ارفع ملف Excel المبيعات + تقرير الشهر الماضي PDF

2

الخطوة 2

حلل: نمو الإيرادات – المنتجات الأعلى أداءً – الانحرافات عن الهدف

3

الخطوة 3

أخرج تقريرًا تنفيذيًا بصفحة واحدة مع 3 توصيات فورية



Claude للاستراتيجية ودعم القرار

نماذج استخدام استراتيجي جاهزة للتطبيق الفوري

خطة تنفيذية 90 يومًا

الطلب: 'بناءً على [الهدف]، ابن خطة لـ 90 يومًا:
الأسبوع 1-4 / الشهر 2 / الشهر 3 مع مؤشرات
النجاح'

المخرج: خطة مراحل + مؤشرات + نقاط مراجعة

مقارنة الخيارات

الطلب: 'قارن [خيار أ] و [خيار ب] و [خيار ج]
باستخدام: [م1]، [م2]، [م3]. أوص بخيار واحد مع
تبرير واضح'

المخرج: جدول مقارنة + درجات + توصية معللة

تحليل SWOT

الطلب: 'أنت مستشار استراتيجي. حلل [شركتي]/
منتجي] باستخدام SWOT مع ترتيب العوامل
بالأولوية وتوصية واحدة فورية'

المخرج: جدول SWOT + خريطة الأولويات + خطوة
تنفيذية

⚠️ Claude ليس قرار – هو دعم قرار. قدّم له الحقائق والقيود وسياقك الكامل، واحتفظ بالقرار النهائي لنفسك.



Scheduled Tasks – المهام المجدولة

اطلب من Claude تنفيذ مهام تلقائيًا في مواعيد محددة – بدون إشراف

أمثلة عملية

- **تقرير أسبوعي كل إثنين:** كل إثنين صباحًا، اقرأ ملف أداء الأسبوع وأنشئ تقريرًا تنفيذيًا في مجلد التقارير'
- **تنظيف الملفات شهريًا:** في أول كل شهر، رتب مجلد Downloads واحذف الملفات المؤقتة وأنشئ ملخصًا بما تم'
- **تلخيص البريد يوميًا:** كل يوم الساعة 8 صباحًا، لخص أهم 5 رسائل في Gmail وضعها في ملف اليومية'

ما هي Scheduled Tasks؟

ميزة تسمح لك بتحديد مهمة وتوقيت تنفيذها – Claude ينفذها تلقائيًا في الوقت المحدد حتى لو لم تكن أمام الشاشة.

كيف تجدول مهمة – 3 خطوات:

3

أكد وراقب

يؤكد الجدولة Claude

2

أضف التوقيت

'نفذ هذا تلقائيًا كل [يوم/أسبوع] في [الوقت]'

1

افتح Claude App

اكتب مهمتك كالمعتاد

تعمل فقط مع Desktop App + اشتراك Pro/Max/Team



Follow a Plan – التنفيذ متعدد الخطوات

الفرق بين طلب عادي و خطة تنفيذية واضحة

طلب عادي

'نظم ملفات مشروعى وأنشئ تقريراً'

المشكلة:

- Claude يفتر بطريقته الخاصة
- يتخطى خطوات قد تراها مهمة
- النتيجة غير متوقعة
- تحتاج لإعادة التوضيح مرات عدة

خطة تنفيذية واضحة

'اتبع هذه الخطة بالترتيب تمامًا:'

1. ادخل مجلد /المشروع واقراء كل الملفات
2. صنفها: تقارير / صور / مسودات / نهائي
3. أنشئ مجلدًا فرعيًا لكل تصنيف
4. انقل الملفات للمجلدات الصحيحة
5. أنشئ ملف README.md يصف هيكل المشروع

القاعدة: كلما كانت الخطة أكثر تفصيلاً وتسلسلاً، كلما كانت النتيجة أكثر توقعًا وأقل حاجة لتصحيح.



Connectors – ربط Claude بعالمك الرقمي

Claude يتكامل مع أدواتك المفضلة – يقرأ ويكتب وينفذ مباشرة

التخزين والبيانات

Google Drive · OneDrive · Salesforce



البحث الذكي في آلاف الملفات – استخراج بيانات العملاء – مقارنة التقارير آلياً

التقويم والاجتماعات

Google Calendar · Zoom · Teams



إنشاء وتحرير المواعيد – تلخيص اجتماعات – تحديد أوقات الفراغ المشتركة

الإنتاجية والتواصل

Gmail · Slack · Notion · Asana



قراءة وتلخيص الرسائل – استخلاص المهام من سلاسل المحادثات الطويلة – تحديث المهام تلقائياً

التصميم والتطوير

Figma · GitHub · Jira · Confluence



مراجعة التصميم – إدارة المهام البرمجية – قراءة التوثيق التقني وتلخيصه



للإضافة: Settings → Connectors → اختر الأداة → امنح الصلاحيات. يمكن تفعيل حتى 5 connectors في المحادثة الواحدة.



Skills – وصفات مخصصة للمهام المتكررة

علّمه كيف يعمل مرة واحدة – يطبق تلقائيًا في كل مرة

User Preferences	Projects	Skills	
	كل المحادثات دائمًا	مشروع واحد محدد	تُفَعَّل عند الحاجة لمهمة محددة
	هويتك وأسلوبك العام	سياق ومعرفة مشروع	إجراءات وقواعد لمهمة متكررة
	تفعيل تلقائي	عند فتح المشروع	يدوي

مثال – بناء Custom Skill لمراجعة منشورات LinkedIn:

```
{
  "name": "linkedin_reviewer",
  "description": "مراجع منشورات LinkedIn",
  "instructions": "عند مراجعة منشور":
  1. ابدأ بقصة قصيرة جذابة
  2. فكرة رئيسية واحدة
  3. أضف 3-4 نقاط ذات قيمة
  4. اختتم بسؤال تفاعلي
  5. الحد الأقصى 1300 حرف
  6. "يمنع الهاشتاقات"
}
```

القاعدة: أي إجراء تشرحه ل Claude أكثر من 3 مرات يستحق أن يصبح Skill مخصصة.






الأدوات
والتكاملات
والتنفيذ



Ai

الأدوات والتكاملات والتنفيذ

ما ستكتسبه

Code Execution و Web Search القدرات التنفيذية الحقيقية ل Claude J

Tool Use و MCP البنية التقنية التي تجعل Claude يتصل بالعالم الخارجي

Privacy كيف تتحكم ببياناتك وتحمي معلوماتك الحساسة

Computer Use التحكم في الواجهة وتنفيذ المهام المعقدة



كيف يستخدم Claude الأدوات؟ – Tool Use

ما معنى أن Claude 'يستخدم أداة' – شرح تقني مبسط



أنواع الأدوات المتاحة لـ Claude:

تنفيذ الكود كتابة وتشغيل Python, JS وأكثر	بحث الويب استرجاع معلومات حديثة من الإنترنت
قراءة الملفات صور وأكثر, PDF, Excel, DOCX	Connectors وغيرها Notion, Drive, Gmail

Claude يختار الأداة المناسبة تلقائيًا بناءً على طلبك – لكن يمكنك طلب أداة محددة صراحةً.



MCP – بروتوكول ربط الأدوات الخارجية

Model Context Protocol – المعيار الذي يجعل Claude يتصل بأي أداة

ما هو MCP؟

بروتوكول مفتوح المصدر من Anthropic يسمح ل Claude بالتواصل مع أي أداة خارجية بطريقة موحّدة وآمنة.

أمثلة على MCP Servers:

- **Gmail MCP**: قراءة وإرسال وتنظيم البريد الإلكتروني مباشرة
- **Google Drive MCP**: البحث وقراءة وتعديل الملفات على Drive
- **Notion MCP**: إنشاء وتحديث صفحات Notion من المحادثة
- **GitHub MCP**: قراءة المستودعات وفتح وإغلاق المشكلات
- **Canva/Figma MCP**: تعديل التصميم وإنشاء محتوى مرئي

MCP vs التقليدي API

معياري واحد يعمل مع أي أداة تدعمه

يحتاج تكاملاً مخصصاً لكل أداة

MCP مفتوح المصدر – أي شركة أو مطور يمكنه بناء Server MCP لأداته.



Web Search المدمج – البحث واسترجاع المعلومات

Claude يبحث تلقائيًا عندما يحتاج معلومات حديثة تتجاوز معرفته.
يختار توقيت البحث ذاتيًا أو يمكنك طلبه صراحةً.

أفضل الممارسات



حدود البحث – يجب معرفتها

- لا يصل للمحتوى خلف كلمات مرور
- محتوى بعض المواقع محجوب
- لا يتصفح الصفحات كاملًا – يرى النصوص

للبحث المتعمق: استخدم ميزة Deep Research المتاحة في Claude Pro/Max.



Code Execution – تنفيذ الكود داخل Claude

Claude يكتب ويُشغّل الكود مباشرة – ليس فقط يقترحه

وتحليل البيانات Python

يكتب ويشغّل Python مباشرة – تحليل CSV،
رسم مخططات، معالجة بيانات، خوارزميات
حسابية

HTML وJavaScript

إنشاء صفحات تفاعلية، مكوّنات واجهة، نصوص
تشغيل، تحويل البيانات بين الصيغ

أوامر Bash وNLP

تحليل نصوص، معالجة ملفات، تحويلات الصيغ
(PDF→Text, Excel→JSON)

حدوده:

- لا يتصل بالإنترنت أثناء التنفيذ
- لا يصل لملفاتك المحلية مباشرة
- محدودية الوقت والذاكرة

كيف تطلبه:

- 'اكتب كود Python يحل هذا الملف'
- 'اشغّل هذا الكود وأرني النتيجة'
- 'عدّل الكود حتى ينجح بدون أخطاء'

Code Execution متاح في Claude Chat مباشرة – لا تحتاج Code أو Cowork.



Computer Use – التنفيذ على مستوى الواجهة

Claude يرى شاشتك ويتحكم فيها – القدرة الأكثر تقدمًا

⚠ تحذيرات جوهرية:

ميزة تجريبية – قد تحدث أخطاء غير متوقعة

لا تمنحه صلاحية الوصول لحسابات مالية أو بيانات حساسة

راقب ما يفعله – لا تتركه يعمل دون مراقبة

اختبر أولًا في بيئة آمنة قبل الاستخدام الإنتاجي

ما هو Computer Use؟

قدرة تتيح ل Claude رؤية الشاشة والضغط على الأزرار وملء النماذج وتصفح المواقع – كأن شخصًا يجلس أمام الحاسوب وينفذ نيابة عنك.

حالات الاستخدام:

- ملء نماذج متكررة على مواقع متعددة
- تصفح وجمع بيانات من مواقع لا تدعم API
- تنفيذ سير عمل يدوي معقد على أي تطبيق

Computer Use متاح حاليًا عبر Claude API للمطورين، وبشكل تدريجي في Claude Desktop App.



Privacy والأمان – تحكّم بياناتك بوعي

معرفة هذه الإعدادات واجب – خاصّة إن كنت تتعامل مع بيانات حساسة

Improve Claude (استخدام بياناتك للتدريب)

افتراضي مفعّل – أوقفه: Settings → Privacy → 'Improve Claude' → OFF

للخطط Free/Pro – أوقفه إذا تعاملت مع بيانات عملاء أو أسرار تجارية

المحادثات المؤقتة (Temporary Chats)

فعله للمعلومات العابرة: New Conversation → Temporary

المحادثات المؤقتة لا تُحفظ ولا تُستخدم في التدريب

خطط Team/Enterprise

عدم استخدام البيانات للتدريب مضمون افتراضياً

بيانات Cowork تبقى محلياً

سجل محادثات Cowork يُحفظ على حاسوبك فقط – لا يُرسل ل Anthropic

القاعدة: لا تشارك كلمات المرور أو أرقام البطاقات في أي محادثة. استخدم المحادثات المؤقتة للمعلومات العابرة.



التكلفة والأداء – منطق الاستخدام العملي

ما هي ال Tokens؟ ال Token هي وحدة معالجة النص – كلمة واحدة ≈ 1.5 token. طلبك + إجابة Claude = مجموع ال tokens المستهلكة.

مقارنة التكلفة النسبية:

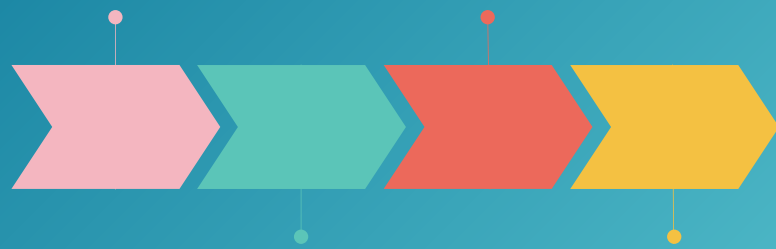
النموذج	الاستخدام الأمثل	التكلفة
Claude Opus 4.6	التحليل الاستراتيجي العميق فقط	●●●●●
Claude Sonnet 4.6	الاستخدام اليومي الاحترافي	○●●●●
Claude Haiku 4.5	المهام البسيطة والتكرارية	○●●●●

نصائح لتقليل الاستهلاك:

- استخدم Haiku للمهام البسيطة والسريعة
- لا ترفع ملفات أكبر مما يحتاجه الطلب
- أنشئ Projects بدل تكرار السياق في كل محادثة
- فَعّل Extended Thinking فقط عند الحاجة الفعلية

الأدوات موجودة

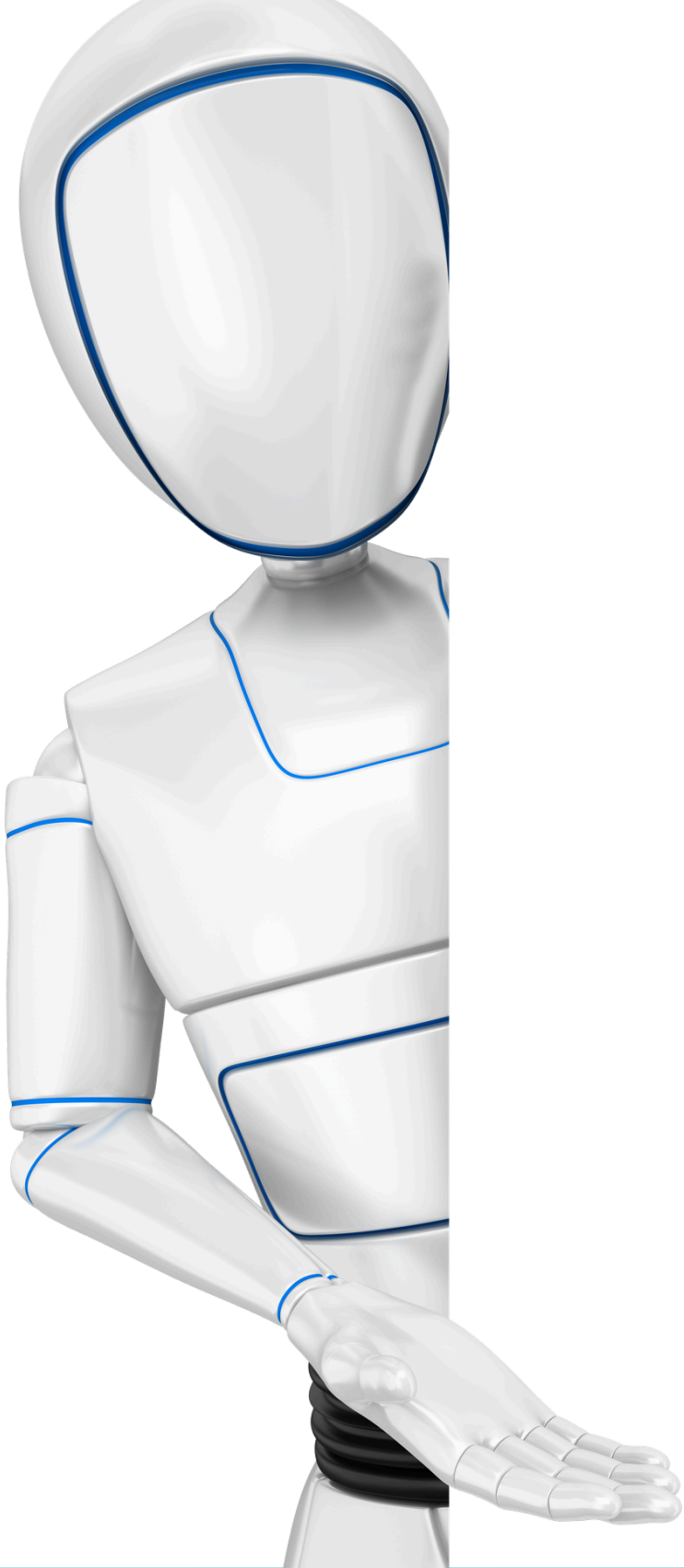
التحدي هو تصميم Workflow يناسب عملك.





Claude Code بالتفصيل الكامل





بالتفصيل الكامل Claude Code

ما ستكتسبه

خطوة بخطوة - من الصفر للتشغيل الفعلي

كتابة Tests مع Claude باحترافية

سر إعطاء Claude ذاكرة مشروعك الدائمة

Headless Mode للمحترفين والأتمتة الكاملة



ما هو Claude Code؟

تعريف دقيق: Claude Code أداة سطر أوامر (CLI) تعمل من Terminal. تتيح ل Claude ل قراءة وكتابة وتعديل الكود مباشرة داخل مشروعك الحقيقي – لا نسخ لصق.



يكتب ويعدّل الكود مباشرة

يفتح الملفات ويكتب فيها – لا copy/paste، نتيجة فورية



يعمل بالتعاون معك

يقترح وينفذ بإذنك – أنت تراجع وتوافق قبل كل تغيير



يقرأ كامل Codebase

يفهم بنية المشروع والعلاقات بين الملفات من أول تشغيل



يتذكر سياق المشروع

عبر ملف CLAUDE.md يتعلم قواعذك وأسلوبك في كل جلسة

Claude Code ليس مجرد 'Copilot' – إنه مطوّر يعمل معك داخل مشروعك الحقيقي بصلاحيات كاملة تمنحها أنت.

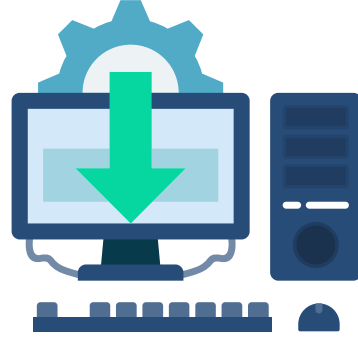


متى تستخدم Claude Code بدلاً من Claude App؟

المعيار	Claude Code	Claude App
البيئة	حصراً CLI/Terminal	متصفح أو تطبيق
الوصول للكود	يقرأ ويعدّل الملفات مباشرة	نسخ لصق يدوي
حجم المشروع	مشاريع كاملة متعددة الملفات	أسئلة ومقتطفات
التعديلات المتسلسلة	مثالي - يتتبع كل التغييرات	غير عملي لتعديلات كثيرة
التكامل مع Git	مدمج - يقرأ diff ويقترح commits	يدوي بالكامل
الاستخدام الأمثل	بناء مشاريع، refactoring، debugging	شرح، توليد كود قصير
المتطلبات	Node.js 18+ + npm install -g claude	أي Claude حساب + متصفح

القاعدة: إذا كنت تشرح أو تتعلم → Claude App. إذا كنت تبني أو تصحّح أو تعيد هيكلة → Claude Code.





تثبيت وإعداد Claude Code

من الصفر لأول تشغيل ناجح – 5 خطوات فقط

تسجيل الدخول

شغلّ claude وسيطلب تسجيل الدخول بحسابك على claude.ai (يفتح المتصفح للتفويض)



تثبيت Claude Code

```
npm install -g @anthropic-ai/claude-code
```



تثبيت Node.js 18+

node --version: وثبته. تحقق LTS حمل الإصدار ← nodejs.org



التحقق من العمل

اشرح هيكل هذا المشروع ' → claude



تهيئة المشروع

```
cd my-project → claude /init
```



Claude Code يعمل على macOS و Linux و Windows (WSL). النسخة المجانية محدودة – يُوصى بـ Pro/Max للاستخدام الجدي.



فهم بيئة عمل Claude Code

نموذج الصلاحيات:

يحتاج موافقتك	أي تعديل على ملف – Claude يقترح ويطلب الموافقة قبل التنفيذ
تلقائي	قراءة الملفات وتحليل البنية – لا يحتاج إذن للقراءة
بإذن صريح	تشغيل أوامر shell – تحتاج كتابة yes صراحة
ممنوع	الوصول لملفات خارج المشروع أو شبكة الإنترنت بدون أداة

ما يراه Claude Code:

- **مجلد المشروع:** يقرأ كل الملفات في المجلد الحالي وما بداخله
- **CLAUDE.md:** يقرأه أولاً في كل جلسة – التعليمات والقواعد الثابتة
- **gitignore:** يحترمه – لا يلمس ما هو مستبعد من Git
- **ما خارج المجلد:** لا يرى ولا يلمس ملفات خارج نطاق المشروع أبداً



Claude لا يُغيّر ملفاً بدون أن تراه – كل تغيير يُعرض عليك أولاً. هذا تصميم مقصود للسلامة.



العمل مع Codebases موجودة – أمر/init

كيف تجعل Claude Code يفهم مشروعك القائم من أول جلسة

2

شغّل/init

claude /init – يقرأ الهيكل ويُنشئ – claude /init

4

اطلب فهم المشروع

'اشرح هيكل هذا المشروع وأهم التبعية'

1

انتقل للمجلد

cd /path/to/your/project

3

راجع CLAUDE.md

افتحه وأضف: قواعدك, conventions, ما تتجنبه

محتوى CLAUDE.md المثالي:

```
# Project: Marketing Dashboard
## Stack: React 18, TypeScript, Tailwind
## Rules:
• Use TypeScript strict mode always
• Components: functional only, no classes
• Tests: Jest + React Testing Library
• Commits: conventional commits format
## Avoid:
• Never use any type
• No inline styles
```

– CLAUDE.md هو استثمار لمرة واحدة –
كلما كان أدق كلما كانت جلساتك اللاحقة أسرع وأكثر دقة.



Claude Code مع Debugging

كيف تصف المشكلة – وكيف يحلها Claude ويقترح الحل

وصف فعّال

'دالة calculateTotal () في cart.ts تُرجع NaN عندما يكون discount صفرًا. المتوقع: الإجمالي الكامل. الخطأ يظهر فقط في حالة الخصم = 0'

وصف ضعيف ❌

'عندي bug في الكود'

لا يعرف: أين؟ ماذا حدث؟ ماذا توقعت؟ Claude

سير عمل Debugging:

1

صف المشكلة بدقة

الملف + الدالة + السلوك المتوقع + السلوك الفعلي

3

اطلب تحليلًا خطوة بخطوة

'حلّ الكود وأخبرني سبب الخطأ قبل أن تصلحه'

2

أرفق رسالة الخطأ كاملة

أعطه stack trace كامل – لا تلّخصه بنفسك

4

راجع قبل التطبيق

اقرأ الحل المقترح – تأكد من فهمك قبل yes

\ اطلب من Claude 'شرح سبب الخطأ' أولاً قبل 'الإصلاح' – هذا يضمن فهمك وتجنّب تكرار نفس المشكلة.



Refactoring وكتابة Tests مع Claude Code

إعادة الهيكلة – Refactoring

- **تحديد الهدف:** 'اعد هيكلة UserService.ts لتطبيق مبدأ Single Responsibility – لا تغير السلوك'
- **تصغير الدوال:** 'كل دالة تجاوزت 40 سطرًا في هذا الملف – قسمها مع الحفاظ على الاسم الأصلي'
- **تحسين الأداء:** 'ابحث عن queries N+1 في هذا الكود واقترح حلًا'

كتابة Tests – الممارسات الصحيحة

اطلب Tests احترافية:

- 'اكتب unit tests ل cart.ts بتغطية +90%'
- 'أضف edge cases: null, empty, boundary values'
- 'اكتب integration test يختبر flow الشراء كاملاً'

قواعد مهمة للطلب:

- حدد framework المستخدم: Jest, Vitest, Pytest...
- اذكر اصطلاحات التسمية في مشروعك
- اطلب mocking لل dependencies الخارجية

أفضل ممارسة: 'أضف test يثبت هذا ال bug أولاً – ثم أصلحه'



CLAUDE.md – ذاكرة مشروعك الدائمة

الملف الذي يحوّل Claude Code من مساعد عام إلى مطوّر يعرف مشروعك

مثال CLAUDE.md حقيقي:

```
# E-Commerce Platform
Node.js + Express + PostgreSQL + React

## Code Rules:
ESLint + Prettier
No raw SQL – use Prisma ORM only
Error handling: always use AppError class

## Structure:
/src/api → routes & controllers
/src/services → business logic
/src/models → Prisma schema only

## Never:
No console.log in production code
```

ماذا تضع في CLAUDE.md؟

1. وصف المشروع: الاسم، الهدف، Stack التقني، مستوى الجمهور
2. قواعد الكود: Style guide, naming conventions, linting rules
3. بنية المجلدات: أين كل شيء – /src, /tests, /docs, ولماذا
4. ما يجب تجنبه: Patterns محظورة، dependencies مرفوضة، أسباب التقييد
5. سير العمل المعتاد: كيف تشغّل التطوير، Build, Test, Deploy

ابدأ بـ init/ ثم أضف ما يخصك يدويًا. راجع CLAUDE.md دوريًا وحديثه مع تطور المشروع.



Claude Code في Skills, Hooks, و Slash Commands

أدوات التخصيص المتقدمة في Claude Code

مهارات مخصصة – Skills

أضف مهارات JSON تُفَعَّل تلقائيًا عند الحاجة.
مثال 'code-reviewer' تُطبَّق معاييرك في كل مراجعة.

خطافات التنفيذ – Hooks

أوامر تُشغَّل تلقائيًا قبل أو بعد تنفيذ Claude:

- **PreToolUse hook:** npm run lint --fix (قبل كل تعديل)
- **PostToolUse hook:** npm run test --changed (بعد كل تعديل)

اختصارات – Slash Commands

أوامر مختصرة تكتبها في المحادثة لتنفيذ مهام متكررة:

- /review → مراجعة شاملة للكود
- /test → للملف tests كتابة
- /doc → توليد توثيق
- /commit → اقتراح commit message
- /pr → إعداد PR description

Hooks هي القوة الخفية – اضبط pre-commit hook لتشغيل linting تلقائيًا قبل كل تعديل Claude.



GitHub Workflows و PRs J Claude Code

تكامُل متقدم مع Git و GitHub Actions

GitHub Actions + Claude Code:

```
# .github/workflows/review.yml
on: [pull_request]
jobs:
  claude-review:
    runs-on: ubuntu-latest
    steps:
      - uses: anthropics/claude-code-action
```

ما يقدر Claude على فعله في CI/CD:

- تعليق مراجعة تلقائية على كل PR
- فحص جودة الكود وفق معاييرك
- اقتراح إصلاحات على المشاكل المكتشفة

سير عمل Pull Requests:

- **مراجعة PR موجود:** 'راجع هذا PR واطرح التغييرات وأي مخاوف تراها'
- **كتابة PR description:** 'اكتب PR description واضحًا من التغييرات في هذا diff'
- **تنفيذ review feedback:** 'طبّق هذه التعليقات من المراجع على الكود'
- **اقتراح commit messages:** 'اقتراح commit message وفق conventional commits'



تكامُل GitHub Actions يعني: كل PR يُراجع تلقائيًا بمعاييرك – بدون عناء يدوي.



Remote, Headless, CI/CD – Claude Code عن بُعد

استخدام Claude Code في بيئات غير تفاعلية

SSH / Remote Server

شغل Claude Code مباشرة على server بعيد عبر SSH – نفس تجربة المحلي.

```
$ ssh user@server
$ claude
> refactor this API endpoint
```

Headless Mode

تشغيل بدون تدخل بشري – يُمرّر الموافقة تلقائيًا. مناسب للأتمتة.

```
$ claude --headless --yes-
always \
  -p 'fix all lint errors'
```

CI/CD Integration

أدمج Claude Code في pipeline التطوير – اجعله يُشغل على كل commit تلقائيًا.

```
# في pipeline script:
claude --headless --yes-always
\
  -p 'write tests for new code'
```

Headless Mode قوي لكن خطير – لا تستخدم --yes-always على بيئة production. دائمًا على staging أوّلًا. ⚠️





أفضل ممارسات Claude Code

قائمة مراجعة قبل كل جلسة عمل جديدة

موصى به

- حدّث CLAUDE.md قبل بدء مهمة جديدة نوعًا
- قسّم المهام الكبيرة لخطوات صغيرة قابلة للتحقق
- اطلب تشغيل Tests بعد كل تعديل مهم
- استخدم git diff لمراجعة التغييرات قبل commit

إلزامي

- خذ نسخة احتياطية (git commit) قبل أي جلسة
- راجع كل تغيير قبل الموافقة – لا تضغط yes عميًّا
- لا تمنح صلاحية الوصول لملفات إنتاج حساسة
- راجع .gitignore – تأكد أن secrets خارج النطاق

القاعدة الذهبية: كلما كانت المهمة أصغر وأوضح، كانت نتيجة Claude Code أكثر دقة وأقل مفاجآت.



حدود Claude Code وقواعد المراجعة البشرية

متى لا تثق بالتنفيذ التلقائي – Human-in-the-Loop

متى يجب التدخل البشري؟

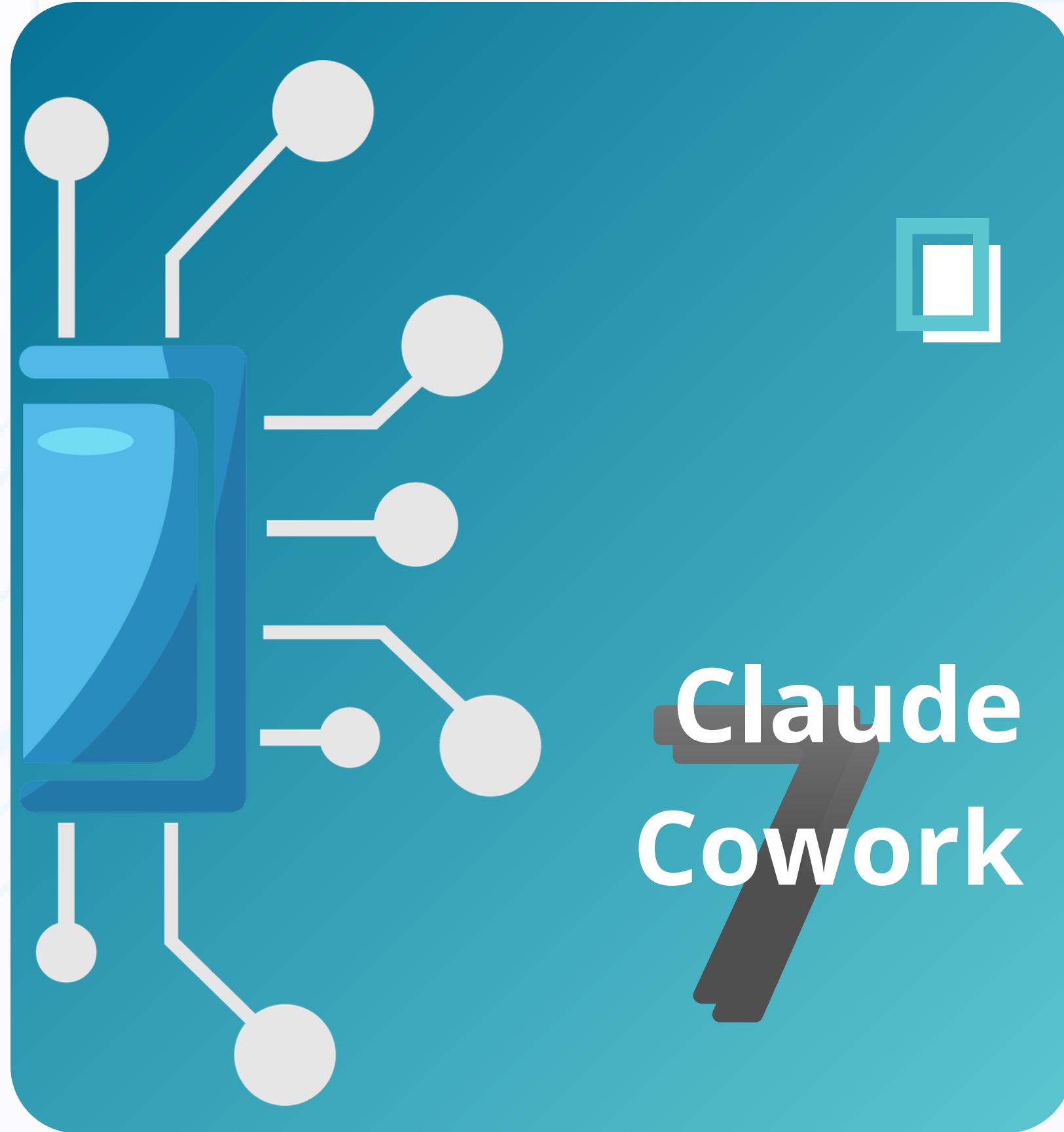
- أي تغيير يمس منطق الدفع أو البيانات المالية
- تعديلات على قاعدة البيانات الإنتاجية أو migrations
- تغييرات في Authentication و Authorization
- Deploy لبيئة production دون اختبار staging
- حذف ملفات أو بيانات بشكل دائم
- تغييرات تؤثر على contract API مع عملاء خارجيين

⚠️ حدود مهمة يجب معرفتها:

- **لا يفهم متطلبات العمل:** يكتب كودًا صحيحًا تقنيًا لكن قد يُخطئ في business logic
- **لا يرى الأثر الكامل:** تعديل في مكان قد يكسر منطقيًا في مكان آخر
- **Rate Limits موجودة:** جلسات مكثفة قد تصطدم بحدود الاستخدام
- **لا يحفظ حالة بين الجلسات:** كل جلسة تبدأ من صفر إلا ما هو في CLAUDE.md

تكامّل GitHub Actions يعني: كل PR يُراجع تلقائيًا بمعاييرك – بدون عناء يدوي.





Claude Cowork

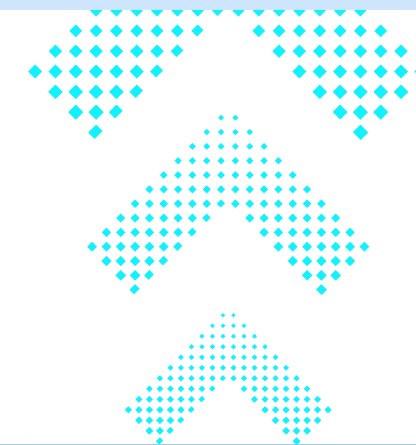
ما ستكتسبه

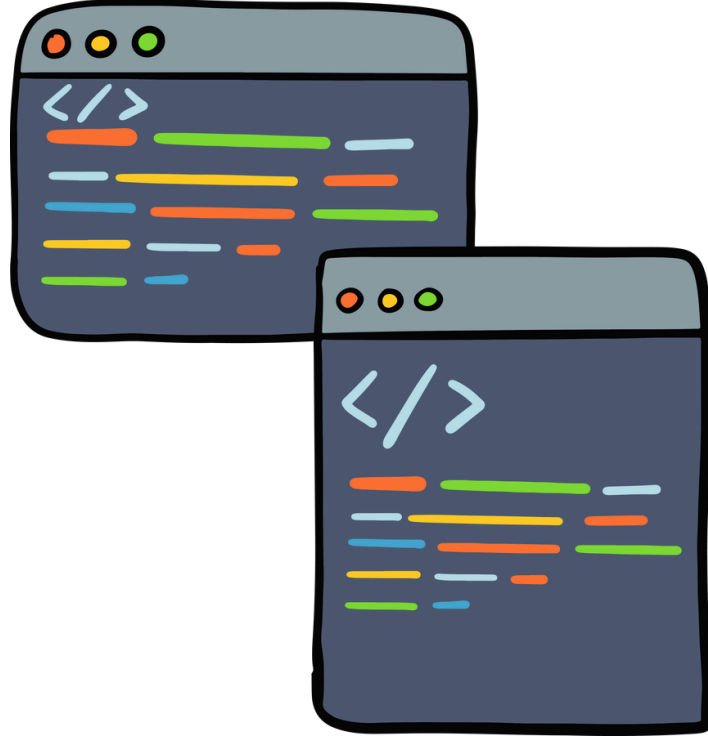
بدون كود – Claude ينفذ مهامك المعقدة بملفاتك المحلية

متى تختار كل منها – Claude Code و Claude App و Cowork الفرق بين

خطوة بخطوة – من منح الوصول للتقرير النهائي

سيناريوهات حقيقية ودمج في سير عمل واحد





ما هو Claude Cowork ؟

تعريف دقيق: Cowork مبني على نفس تقنية Claude Code – لكن بواجهة حوارية بسيطة. أعطه إذنًا للوصول لمجلد محدد على حاسوبك، فيحلل ويخطط وينفذ المهام المعقدة نيابة عنك.

4 قدرات أساسية:

1. **قراءة وكتابة الملفات:** يفتح ويقرأ ويعدّل أي ملف في المجلد الممنوح إذنًا
2. **تصفح وتنظيم المجلدات:** يبحث ويُصنّف ويعيد تسمية ويرتب الملفات
3. **جدولة المهام التلقائية:** ينفذ مهام في توقيت محدد دوريًا بدون تدخل
4. **بناء التقارير والتحليلات:** يجمع البيانات ويُحللها ويُخرج تقارير منسّقة

الفرق الجوهرى عن Claude Chat:

- Claude Chat يُجيب – Cowork يُنفذ
- يقرأ ملفاتك الحقيقية ويكتب فيها مباشرة
- يعمل في الخلفية دون إشراف مستمر

Cowork يتطلب Desktop App + اشتراك Pro/Max/Team. لا يعمل في المتصفح.



Cowork مقابل Claude App مقابل Claude Code

المعيار	Claude Code	Claude Cowork	Claude App
المستخدم المستهدف	مطورون وتقنيون	مديرون ومحللون	الجميع
البيئة	CLI/Terminal	Desktop App حصراً	متصفح أو تطبيق
الوصول للملفات	وصول كامل بإذنك	وصول كامل بإذنك	رفع يدوي فقط
نوع المهام	كود + debugging + بناء	ملفات + تنظيم + تقارير	محادثة + تحليل + كتابة
العمل في الخلفية	نعم (headless)	نعم (/schedule)	لا
الجدولة التلقائية	نعم (CI/CD)	نعم	لا
المتطلبات	Node.js + npm	Desktop + Pro/Max	Pro مناسب

القاعدة: إذا لا تكتب كودًا وتريد أتمتة ملفاتك → Cowork. إذا تكتب كودًا → Claude Code. للمحادثة والكتابة → App.



سير عمل Cowork الأساسية – من الصفر للتنفيذ

4 خطوات تحوّل مهمة معقدة إلى تنفيذ تلقائي

جدولة المهام المتكررة

'نُفذ هذا كل إثنين صباحًا' أو
استخدم أمر /schedule/
للتحكم الكامل.

راقب وتدخل عند الحاجة

يعرض خطته أولًا – Cowork
راجعها قبل الموافقة. يمكنك
تعديل أي خطوة قبل التنفيذ

وصف المهمة بوضوح

اكتب المهمة بالتفصيل –
الهدف، الملفات المستهدفة،
المخرج المتوقع، أي قيود

منح الوصول للمجلد

Cowork → Desktop App →
اختر المجلد المستهدف →
يرى Claude. امنح الإذن
محتواه فقط.

التنفيذ بدفعات صغيرة أفضل من مهمة ضخمة واحدة – يمنحك تحكمًا أكبر ويقلل الأخطاء.



Cowork للمحللين والمديرين وصناع القرار

4 سيناريوهات حقيقية جاهزة للتطبيق اليوم

تقرير حالة أسبوعي

قبل: تجمع البيانات يدويًا من عدة ملفات وتكتب التقرير
بعد: كل إثنين صباحًا: يقرأ ملفات الأسبوع → يكتب التقرير → يحفظه في المجلد

تحليل فواتير شهرية

قبل: تفتح 30 فاتورة يدويًا وتدخل الأرقام في Excel
بعد: Cowork يقرأ كل الفواتير → يُنشئ جدول Excel منظم تلقائيًا في دقائق

بحث تنافسي موثّق

قبل: تجمع البيانات يدويًا من الويب وتنسق التقرير
بعد: Chrome يجمع البيانات → Cowork يحللها ويكتب تقرير المقارنة محليًا

ترتيب مجلد المشاريع

قبل: ساعات في ترتيب Downloads وإعادة تسمية الملفات المبعثرة
بعد: 'رتب مجلد Downloads: صنّف حسب النوع وأعد تسمية بالتاريخ' – دقائق

ابدأ بأبسط مهمة تكررية في يومك – هذه هي نقطة الدخول الأفضل لـ Cowork.



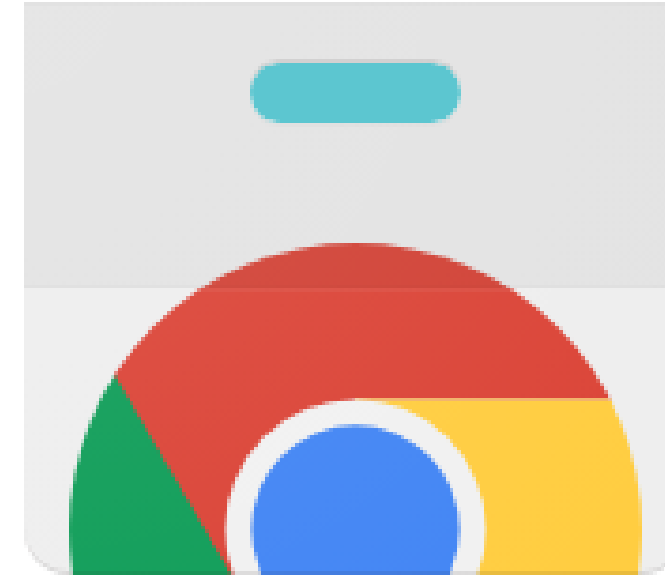
سير عمل المعرفة – Cowork + Chrome معًا

السيناريو الأقوى: اجمع البيانات من الويب وحلها محليًا تلقائيًا



مجلد محلي

النتائج تُحفظ في مجلد محدد
على حاسوبك – منطمة وجاهزة



Chrome Extension

الإضافة تتصفح المواقع
وتستخرج البيانات – منافسون،
أسعار، أخبار



المخرج النهائي

تقرير Word أو Excel احترافي في
مجلدك – جاهز للعرض



Claude Cowork

الإضافة تتصفح المواقع
وتستخرج البيانات – منافسون،
أسعار، أخبار



سير عمل المعرفة – Cowork + Chrome معًا

السيناريو الأقوى: اجمع البيانات من الويب وحلها محليًا تلقائيًا

مثال تطبيقي – بحث تنافسي أسبوعي:

Chrome: اجمع آخر 5 منشورات لكل منافس من LinkedIn وسعر منتجاتهم من موقعهم

→ احفظ في /competitive-intel/

Cowork: حلّل كل الملفات واكتب تقرير مقارنة أسبوعي في /reports/

هذا السيناريو يحوّل ساعتين من عمل يدوي أسبوعي إلى 15 دقيقة – مع نتيجة أكثر تنظيمًا واتساقًا.



أفضل ممارسات Cowork

إلزامي – مراجعة الخطة

راجع الخطة قبل الموافقة – Cowork يعرض خطة تنفيذه. اقرأها بدقة وعدّل أي خطوة لا توافق عليها.

إلزامي – النسخة الاحتياطية

خذ نسخة احتياطية قبل أي جلسة – git commit أو نسخ يدوي للمجلد. لا يمكن التراجع عن تعديلات Cowork بسهولة.

موصى به

- نَقِّذ بدفعات صغيرة لا بمهمة ضخمة
- ابدأ بمهمة صغيرة لاختبار الفهم أول مرة
- وثِّق ما نجح في ملف README لإعادة الاستخدام

إلزامي – تحديد الوصول

لا تعطه وصولاً أوسع مما يحتاج – امنحه مجلد المشروع فقط.

Cowork يعدّل ملفاتك الحقيقية – لا يوجد Undo تلقائي. النسخة الاحتياطية قبل كل جلسة ليست اختيارية.





التبني
المفني
الحقيقي



Ai

التبني المهني الحقيقي

ما ستكتسبه

مؤسسياً Claude من مستخدم عادي إلى باني أنظمة - كيف تتبنى

محلل / مستقل / مطور - سيناريوهات مخصصة / CEO / CMO

متى تغيّر النموذج؟ الأداة؟ أم سير العمل؟

من أول تثبيت إلى workflow حقيقي في عملك



Claude لكل دور – ماذا يستخدم كل منهم؟

5 أدوار مختلفة – سيناريوهات مخصصة لكل منهم

مدير تنفيذي / CEO

CLAUDE CHAT + COWORK

- تحليل استراتيجي وتقارير تنفيذية
- متابعة أداء الفرق ومراجعة KPIs
- إعداد العروض التقديمية للمجلس
- مراجعة العقود والوثائق المهمة

مدير تسويق / CMO

CLAUDE CHAT + CHROME + COWORK

- كتابة استراتيجيات تسويقية
- تحليل المنافسين ومتابعة السوق
- إنتاج محتوى متعدد القنوات
- مراجعة وتحسين الحملات الإعلانية

محلل بيانات / مالي

CLAUDE CHAT + COWORK + CODE

- تحليل ملفات Excel وال CSV الضخمة
- بناء تقارير شهرية وربعية آلياً
- كتابة Python لتحليل بيانات متكرر

مستقل / صاحب وكالة

CLAUDE CHAT + PROJECTS + CHROME

- مستقل لكل عميل Project
- إنتاج محتوى بأسلوب العميل تلقائياً
- بحث تنافسي وتقارير الأداء

مطوّر / تقني

CLAUDE CODE + CLAUDE CHAT

- يوميًا Debugging و Refactoring
- كتابة Tests وتوثيق الكود
- مراجعة PRs واقتراح تحسينات



سير العمل الموصى بها حسب نوع المهمة

ملاحظة	AI Workflow الموصى	نوع المهمة
استخدم Custom Style محفوظ	Chat + Sonnet	كتابة محتوى LinkedIn
ملف Cowork → Excel → تقرير PDF	Cowork + Sonnet	تقرير بيانات شهري
محدّث قبل البدء CLAUDE.md	Code + Sonnet/Opus	برمجة ميزة جديدة
يحلل Chat → يجمع Chrome	Chrome + Chat + Sonnet	بحث تنافسي
مفعّل Extended Thinking	Chat + Opus	استراتيجية دخول سوق
مجلد مستقل + نسخة احتياطية	Cowork + Sonnet	تنظيم ملفات مشروع
كامل في الطلب stack trace	Code + Opus	خطأ معقد DEBUGGING
مباشرة PPTX → Artifacts	Chat + Sonnet	إعداد عرض تقديمي
قسّم الوثيقة + Extended Thinking	Chat + Opus	تلخيص وثائق قانونية
Gmail Connector + Scheduled Task	Connectors + Haiku	متابعة بريد يومية



Claude متى لا تعتمد كليًا على – Human-in-the-Loop

6 مواقف يجب فيها المراجعة البشرية – بلا استثناء

- | | |
|---|--|
| 1 | بيانات مالية وقرارات الميزانية
أي رقم يدخل تقريرًا ماليًا أو يُبنى عليه قرار استثماري – تحقق من المصدر يدويًا |
| 2 | قرارات قانونية وتعاقدية
Claude لا يُقدّم استشارة قانونية – مراجعة محامٍ إلزامية قبل توقيع أي عقد |
| 3 | نشر علني واسع النطاق
منشور يُرسل لآلاف أو إعلان يُطلق – مراجعة بشرية قبل النشر دائمًا |
| 4 | قرارات تمس الأشخاص
أي قرار يؤثر على موظف أو عميل أو شريك – الإنسان هو من يقرر لا ال AI |
| 5 | بيانات شخصية وحساسة
لا تعالج بيانات عملاء سرية ب Claude دون سياسة privacy واضحة ومعتمدة |
| 6 | Deploy لبيئة الإنتاج مباشرة
لا تنشر كودًا كتبه Claude ل production دون مراجعة بشرية ومرور ب staging |



خارطة طريق 30 يومًا لتبني Claude

من أول تسجيل إلى Workflow حقيقي في عملك

الأسبوع 3 – Workflows حقيقية

- حدّد 3 مهام تكررية أسبوعيًا
- ابنى Workflow لكل منها بـ Cowork/Code
- ربط أداة خارجية واحدة (Gmail/Drive)
- قس الوقت الموفّر وسجّله

الأسبوع 1 – الأساسيات

- سجّل في claude.ai وأنشئ حسابك
- اضبط User Preferences بدقة
- فعّل الذاكرة (Session Memory)
- جرّب 10 محادثات متنوعة يوميًا

الأسبوع 4 – التحسين والتخصص

- راجع ما نجح وما لم ينجح
- حدّث User Preferences و CLAUDE.md
- عمّق الـ Workflow الأنجح
- قرر: هل تحتاج تدريبًا متقدمًا؟

الأسبوع 2 – Projects والذاكرة

- أنشئ Project لأهم مشروع عندك
- أضف ملفات مرجعية لـ Knowledge Base
- جرّب Custom Style لمحتواك
- طبّق إطار CRAFT على كل طلب



من مستخدم إلى باني أنظمة

التحول الذهني الذي يفصل بين النتائج العادية والاستثنائية

مستخدم عادي

يسأل أسئلة فردية – يعيد شرح نفسه كل مرة – لا يبني على ما سبق

نتائج متذبذبة

1

مستخدم محترف

يبني User Preferences و Projects – يستخدم إطار CRAFT – يُعيد استخدام Prompts ناجحة

نتائج ثابتة وإنتاجية أعلى

2

باني أنظمة

يُصمّم Workflows متكاملة – يُدرّب فريقه على Claude – يُقيس الأثر ويُحسن باستمرار – يبني بنية تشغيلية

تحول مؤسسي حقيقي وقابل للقياس

3

أين أنت الآن؟ المرحلة التالية لا تحتاج تقنية جديدة – تحتاج تفكيرًا مختلفًا.



10 مبادئ لاستخدام Claude باحترافية

كل مهمة تكريرية تستحق Skill أو Workflow

مثال واحد أقوى من وصف طويل

السياق يأتي قبل الطلب دائمًا

النسخة الاحتياطية قبل Cowork ليست اختيارية

CLAUDE.md هو استثمار يُسدّد مرات عديدة

النموذج الأبسط هو الأذكى اقتصاديًا

ابن Context Engineering مرة وانتفع منه دائمًا

Claude مستشار لا قرار – القرار لك أنت

راجع كل مخرج مهم قبل الاعتماد عليه

من مستخدم إلى باني أنظمة – هذا هو الهدف الحقيقي





تمامًا مثلك

www.al-bosat.com



HamadaBosat



د.م: حمادة بساط
استشاري التحول الرقمي

