

# Claude Console

platform.claude.com

Przewodnik krok po kroku dla laika

---

Claude Console to panel deweloperski Anthropic. Nie trzeba umieć programować, żeby z niego korzystać — ten przewodnik pokazuje co każda sekcja robi i od czego zacząć.

## 1 Dashboard — punkt startowy [\[zaczynij tutaj\]](#)

### \* Otwierz Dashboard

Po zalogowaniu na platform.claude.com trafiasz na Dashboard. Widac tu saldo kredytów (np. 5,00 USD), skróty do najważniejszych sekcji oraz powiadomienia systemowe.

### \$ Sprawdź kredyty

Kredyty to Twój budżet na zapytania do API. Każde wysłane zapytanie zużycuje pewną ilość kredytów w zależności od modelu i długości wiadomości. Monitoruj je regularnie.

**Wskazówka:** *Dashboard to też dobre miejsce do sprawdzania aktualności — Anthropic informuje tu o zmianach w API i nowych modelach.*

## 2 Workbench — plac zabaw z AI [\[najważniejsze narzędzie\]](#)

### ~ Otwierz Workbench

To interaktywny edytor do testowania zapytań. Tutaj możesz rozmawiać z Claude bez pisania kodu — idealny do eksperymentowania i tworzenia promptów.

### S Wybierz model i parametry

Po prawej stronie znajdziesz panel z ustawieniami: wybór modelu (Sonnet, Opus, Haiku), suwak temperatury (kreatywność odpowiedzi), maksymalna długość odpowiedzi. Zaczynij od domyślnych wartości.

### > Ustaw System Prompt

Pole 'System' na gorze to instrukcja dla Claude obowiązująca przez całą rozmowę — np. 'Jesteś pomocnym asystentem piszącym tylko po polsku'. To potężne narzędzie do personalizacji zachowania modelu.

### <> Kliknij 'Get Code'

Gdy skończyłeś tworzyć prompt w Workbench, kliknij przycisk 'Get code'. Zobaczysz gotowy kod API w Pythonie, JavaScript lub innym języku — możesz go skopiować i użyć w swojej aplikacji.

---

**Wskazo wka:** *Workbench zapisuje historie sesji. Wroce do poprzednich prob, zeby porownac jak zmiana parametrow wpływa na jakosc odpowiedzi.*

### 3 Files, Skills i Batches — zasoby i masowe przetwarzanie

#### **F** **Files**

Przesylaj pliki (PDF, CSV, obrazy), ktore Claude moze analizowac w zapytaniach API. Unikasz wielokrotnego wysylania tych samych plikow.

#### **T** **Skills**

Definiuj niestandardowe 'umiejetnosci' — gotowe bloki instrukcji wielokrotnego uzytku dla Twoich agentow i aplikacji.

#### **B** **Batches**

Wysylaj tysiace zapytan jednoczesnie z 50% rabatem na cenie. Idealne do analizy duzych zbiorow danych, gdy czas odpowiedzi nie jest krytyczny.

**Wskazo wka:** *Jesli przetwarzasz duza liste dokumentow (np. 500 faktur), uzyj Batches zamiast wysylac zapytania jedno po drugim — oszczedzasz pieniadze i czas.*

### 4 Managed Agents — autonomiczne asystenty **[nowoc]**

#### **R** Co to sa agenci?

Agenci to instancje Claude, ktore moga wykonywac wielokrokowe zadania samodzielnie — przeszukiwac internet, wykonywac kod, uzywac narzedzi zewnetrznych. Dzialaja jak 'miniaturowe autopiloty'.

#### **Q** Quickstart — zacznij od szablonu

Sekcja Quickstart zawiera gotowe szablony agentow. Wybierz szablon odpowiedni dla Twojego przypadku (np. analizator danych, asystent emailowy), zmodyfikuj instrukcje i uruchom.

#### **M** Memory Stores i Credential Vaults

Memory Stores to pamiec agenta miedzy sesjami (np. zapamietuje preferencje uzytkownika). Credential Vaults bezpiecznie przechowuje klucze API do zewnetrznych uslug, ktorych agent uzywa.

#### **L** Sessions — sledz dzialania agenta

W zakladce Sessions widzisz historie kazdej sesji agenta: jakie kroki podjal, ile kredytow zuzylo, czy zadanie zostalo ukonczone. Pomocne do debugowania i optymalizacji.

### 5 Analytics — kontrola kosztow i uzycia **[monitoruj regularnie]**

#### **#** Sledz zuzycie tokenow

Analytics pokazuje wykresy zuzycia w czasie: ile tokenow 'wejscowych' (Twoje zapytania) i 'wyjscowych' (odpowiedzi Claude) zuzytu. Tokeny to podstawowa jednostka rozliczeniowa — 1 token to ok. 0,75 slowa.

## \$ Analizuj koszty per model

Filtruj dane według modelu i okresu czasu. Zobaczysz ile kosztuje Cię korzystanie z Opus vs Sonnet vs Haiku — co pozwala świadomie wybrać tańszy model do prostszych zadań.

**Wskazówka:** *Haiku jest najtańszy i najszybszy — świetny do prostych zadań (klasyfikacja, streszczenie). Sonnet to złoty środek. Opus zostawiaj do najtrudniejszych problemów wymagających głębokiego rozumowania.*

## 6 Claude Code — AI w Twoim terminalu [dla programistów]

### > Co to jest Claude Code?

Claude Code to narzędzie wiersza poleceń (CLI), które działa bezpośrednio w Twoim projekcie programistycznym. Może czytać pliki, pisać kod, uruchamiać testy i naprawiać błędy — wszystko przez rozmowę.

### % Monitoruj użycie w konsoli

Zakładka Usage w sekcji Claude Code pokazuje ile tokenów i kredytów zużyła Twoja praca z Claude Code. W Settings możesz ustawić limity wydatków i preferencje modelu.

## 7 Pierwsze kroki — plan działania [zaczynaj tu]

### 1 Idź do Workbench i napisz pierwszy prompt

Wpisz system prompt i kilka wiadomości testowych. Pobaw się suwakiem temperatury. Sprawdź jak różne modele odpowiadają na to samo pytanie.

### 2 Skopiuj wygenerowany kod API

Kliknij 'Get code', wybierz swój język programowania i wklej kod do projektu. Masz działającą integrację z Claude w kilka minut.

### 3 Sprawdź Analytics po kilku dniach

Wróć do Analytics i sprawdź ile kredytów zużyło. Ocen czy używasz odpowiedniego modelu i czy Twoje prompty są efektywne (krótkie i precyzyjne = tańsze).

### 4 Eksploruj Managed Agents

Gdy oswoisz się z podstawami, zajrzyj do Quickstart w sekcji Agents. Gotowe szablony pomogą Ci zbudować pierwszego autonomicznego asystenta bez pisania kodu od zera.

**Wskazówka:** *Dokumentacja: Kliknij link 'Documentation' widoczny w górnym menu Console — [docs.anthropic.com](https://docs.anthropic.com) to oficjalne, szczegółowe źródło wiedzy (po angielsku).*