

Zainstaluj cztery wtyczki. Komendy są takie same na Linuksie i Windowsie (PowerShell).

Wtyczka	Pakiet	Do czego
omnisync	pi-omniroute-sync	Podłącza OMP do OmniRoute i synchronizuje listę modeli. Wymagana.
bigpowers	npm:bigpowers	Zestaw skilli z cyklem pracy: specyfikacja, plan, TDD, review, release. Alternatywa: superpowers .
i-have-adhd	i-have-adhd@i-have-adhd	Krótsze odpowiedzi agenta: najpierw konkretna akcja, potem numerowane kroki.
ponytail	npm:@dietrichgebert/ponytail	Agent pisze minimum kodu: najpierw sprawdza, czy rozwiązanie już istnieje w kodzie, bibliotece standardowej albo zainstalowanych zależnościach.

Instalacja

1. omnisync:

```
omp plugin install pi-omniroute-sync
```

2. bigpowers:

```
omp plugin install npm:bigpowers
```

3. i-have-adhd (z marketplace na GitHubie, więc najpierw dodajesz źródło):

```
omp plugin marketplace add ayghri/i-have-adhd  
omp plugin install i-have-adhd@i-have-adhd
```

4. ponytail:

```
omp plugin install npm:@dietrichgebert/ponytail
```

5. Sprawdź listę:

```
omp plugin list
```

Oczekiwany wynik: `pi-omniroute-sync`, `bigpowers i@dietrichgebert/ponytail` w sekcji `npm Plugins` z kropką (włączone), a `i-have-adhd@i-have-adhd` w sekcji `Marketplace Plugins`.

6. Uruchom ponownie `omp`, żeby wtyczki się załadowały.

Konfiguracja omnisync

W OMP wpisz `/omni setup`, podaj `https://omni.suncode.dev` i klucz API. Szczegóły: [OMP: instalacja](#).

Przydatne komendy:

Komenda	Co robi
<code>/omni setup</code>	Ustawia URL i klucz, potem synchronizuje modele. Uruchom ponownie po zmianie klucza.
<code>/omni sync</code>	Ręcznie odświeża listę modeli.
<code>/omni models <szukaj></code>	Wyświetla dostępne modele.
<code>/omni test <model></code>	Wysyła testowe zapytanie do modelu.
<code>/omni autosync <sekundy></code>	Zmienia interwał automatycznej synchronizacji (domyślnie 300 s, off wyłącza).

Pierwsze kroki z bigpowers

W nowym projekcie napisz do agenta: `ujj skilla using-bigpowers`. Skill opisz cykl pracy i podpowie, od którego skilla zacząć.

superpowers: alternatywa dla bigpowers

`superpowers` to drugi zestaw skilli z cyklem pracy: `brainstorming` → `plan` → `TDD` → `debugowanie` → `code review` → `zakończenie gałęzi`. Jest mniejszy od `bigpowers` i działa też w Claude Code i Codexie.

	bigpowers	superpowers
Zakres	Pełny cykl projektu: discover, plan, build, verify, release; stan w specs/ (YAML)	Cykl jednego zadania: projekt, plan, TDD, review
Liczba skilli	ok. 80	ok. 15
Pierwszy krok	uyj skilla using-bigpowers	uyj skilla using-superpowers (ładuje się sam na starcie sesji)

Instaluj jedno z dwóch, nie oba. Oba zestawy podpowiadają agentowi podobne kroki i będą się wzajemnie zagłuszać.

Zamiast bigpowers (krok 2 instalacji):

```
omp plugin install git:github.com/obra/superpowers
```

Nie używaj npm: superpowers: pakiet o tej nazwie w npm nie pochodzi od autora superpowers.

Przejdźcie z bigpowers na superpowers:

```
omp plugin disable bigpowers
omp plugin install git:github.com/obra/superpowers
```

Claude Code: /plugin install superpowers@claude-plugins-official. Codex: /plugins, wyszukaj superpowers, wybierz Install Plugin.

i-have-adhd: włączanie i wyłączanie

- Włączenie i wyłączenie w bieżącej sesji: wpisz /i-have-adhd. Każde wywołanie przełącza tryb.
- Wyłączenie na stałe: omp plugin disable i-have-adhd@i-have-adhd.

ponytail: poziomy i wyłączanie

- /ponytail lite, /ponytail full (domyślny), /ponytail ultra: siła ograniczania kodu.
- Wyłączenie w bieżącej sesji: /ponytail off albo napisz stop ponytail.
- Wyłączenie na stałe: omp plugin disable @dietrichgebert/ponytail.

Aktualizacja wtyczek

omp plugin upgrade