# Starting Seneca.js

1. Seneca lets you build message based microservice systems.
2. You don’t need to know where the other services are located. Everything external to your business logic - such as databases, caches and third-party integrations
3. Seneca.js is used to organize the back-end API.
4. Seneca is a tool that separates an backend apis into small actions.
5. An action is a function that is identified by a JSON object.
6. Actions lie at the core of Seneca, and in order to use Seneca, a developer must be able to think in terms of small functions that can be called from anywhere by their JSON identifiers.
7. Actions are created using the seneca.add.