HugSQL in Detail

HugSQL encourages SQL, DDL, and DML statements to be stored in SQL files such that you are not concatenating large strings or needing to use leaky-abstraction DSLs in your Clojure projects.

In order to generate the Clojure functions from your SQL statements, HugSQL requires a simple set of conventions in your SQL files. These conventions allow HugSQL to:

- define functions by name
- add docstrings to defined functions
- determine how to execute (the command type):
  - SQL select
  - DDL create table/index/view, drop ...
  - DML insert, update, delete
  - any other statements (e.g. vacuum analyze)
- determine the result type:
  - one row (hash-map)
  - many rows (vector of hash-maps)
  - affected rows
  - any other result you implement
- replace parameters for:
  - Values: `where id = :id`
  - Value Lists: `where id in (:v*:ids)`
  - Tuple Lists (for multi-insert): `values :tuple*:my-records`
  - SQL Identifiers: `from :i:table-name`
  - SQL Identifier Lists: `select :i*:column-names`

    - Raw SQL: `:sql:my-query`

Edit this page