Find User by ID Specialization

Jaime | March 10, 2022

Participants: Engine, Selection (Operation), Postgres (Driver), StatementBuilder, Select (Clause), From (Clause), Where (Clause), Postgres (DB)

- Engine → Postgres (Driver): driver.run(selection)
- Postgres (Driver) → Postgres (Driver): self.selection(selection)
- Postgres (Driver) → StatementBuilder: new(selection, [Select, From, Where])
- StatementBuilder ⇠ Postgres (Driver): builder
- Postgres (Driver) → StatementBuilder: builder.build
- StatementBuilder → Select (Clause): new(selection, {})
- Select (Clause) ⇠ StatementBuilder: clause
- StatementBuilder → Select (Clause): clause.build
- Select (Clause) ⇠ StatementBuilder: ("SELECT *", {})
- StatementBuilder → From (Clause): new(selection, {})
- From (Clause) ⇠ StatementBuilder: clause
- StatementBuilder → From (Clause): clause.build
- From (Clause) ⇠ StatementBuilder: ("FROM users", {})
- StatementBuilder → Where (Clause): new(selection, {})
- Where (Clause) ⇠ StatementBuilder: clause
- StatementBuilder → Where (Clause): clause.build
- Where (Clause) ⇠ StatementBuilder: ("WHERE id = :where_a_id", { where_a_id: 1 })
- StatementBuilder ⇠ Postgres (Driver): ("SELECT * FROM users WHERE id = :where_a_id", { where_a_id: 1 })
- Postgres (Driver) → Postgres (DB): db.fetch("SELECT * FROM users WHERE id = :where_a_id", { where_a_id: 1 })
- Postgres (DB) ⇠ Postgres (Driver): [{ id: 1 }]
- Postgres (Driver) → Selection (Operation): selection.after([{ id: 1 }])
- Selection (Operation) ⇠ Postgres (Driver): { id: 1 }
- Postgres (Driver) ⇠ Engine: { id: 1 }