Write your own SQL

While the Prisma Client API aims to make all your database queries intuitive, type-safe, and convenient, there may still be situations where raw SQL is the best tool for the job.

This can happen for various reasons, such as the need to optimize the performance of a specific query or because your data requirements can't be fully expressed by Prisma Client's query API.

In most cases, [TypedSQL](#) allows you to express your query in SQL while still benefiting from Prisma Client's excellent user experience. However, since TypedSQL is statically typed, it may not handle certain scenarios, such as dynamically generated `WHERE` clauses. In these cases, you will need to use [$queryRaw](#) or [$executeRaw](#), or their unsafe counterparts.

Writing type-safe queries with Prisma Client and TypedSQL
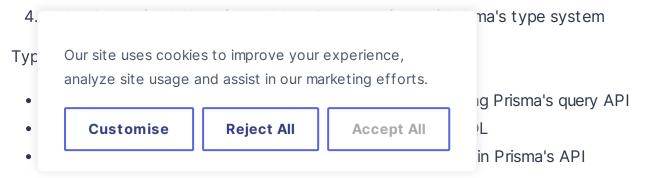
> ⓘ **INFO**
>
> TypedSQL is available in Prisma ORM 5.19.0 and later. For raw database access in previous versions, see [our raw queries documentation](#).

What is TypedSQL?

TypedSQL is a new feature of Prisma ORM that allows you to write your queries in `.sql` files while still enjoying the great developer experience of Prisma Client. You can write the code you're comfortable with and benefit from fully-typed inputs and outputs.

With TypedSQL, you can:

1. Write complex SQL queries using familiar syntax
2. Benefit from full IDE support and syntax highlighting for SQL
3. Import your SQL queries as fully typed functions in your TypeScript code
4. ██ ma's type system

Typ██

• ███████████████████████████████ ng Prisma's query API
• ███████████████████████████████ QL
• ███████████████████████████████ in Prisma's API

Ask AI

By using TypedSQL, you can write efficient, type-safe database queries without sacrificing the power and flexibility of raw SQL. This feature allows you to seamlessly integrate custom SQL queries into your Prisma-powered applications, ensuring type safety and improving developer productivity.

For a detailed guide on how to get started with TypedSQL, including setup instructions and usage examples, please refer to our [TypedSQL documentation](#).

Raw queries

Prior to version 5.19.0, Prisma Client only supported raw SQL queries that were not type-safe and required manual mapping of the query result to the desired type.

While not as ergonomic as [TypedSQL](#), these queries are still supported and are useful when TypedSQL queries are not possible either due to features not yet supported in TypedSQL or when the query is dynamically generated.

Alternative approaches to raw SQL queries in relational databases

Prisma ORM supports four methods to execute raw SQL queries in relational databases:

- [$queryRaw](#)
- [$executeRaw](#)
- [$queryRawUnsafe](#)
- [$executeRawUnsafe](#)

These commands are similar to using TypedSQL, but they are not type-safe and are written as strings in your code rather than in dedicated `.sql` files.

Alternative approaches to raw queries in document databases

For MongoDB, Prisma ORM supports three methods to execute raw queries:

- [$runCommandRaw](#)
- [<model>.findRaw](#)
- [<model>.aggregateRaw](#)

These methods allow you to execute raw MongoDB commands and queries, providing flexibility when you need to use MongoDB-specific features or optimizations.

$ru   del>.findRaw is used to find documents
tha   regation operations. All three methods
are

Sim   re not type-safe and require manual
han