

Getting Started

A high level view and getting set up.

[Overview](#) | [Installation Guide](#) | [Frequently Asked Questions](#) | [Migration from 1.0](#) | [Glossary](#) | [Changelog catalog](#)

SQLAlchemy ORM

Here, the Object Relational Mapper is introduced and fully described. If you want to work with higher-level SQL which is constructed automatically for you, as well as automated persistence of Python objects, proceed first to the tutorial.

- **Read this first:** [Object Relational Tutorial](#)
- **ORM Configuration:** [Mapper Configuration](#) | [Relationship Configuration](#)
- **Configuration Extensions:** [Declarative Extension](#) | [Association Proxy](#) | [Hybrid Attributes](#) | [Automap](#) | [Mutable Scalars](#)
- **ORM Usage:** [Session Usage and Guidelines](#) | [Loading Objects](#) | [Cached Query Extension](#)
- **Extending the ORM:** [ORM Events and Internals](#)
- **Other:** [Introduction to Examples](#)

SQLAlchemy Core

The breadth of SQLAlchemy's SQL rendering engine, DBAPI integration, transaction integration, and schema description services are documented here. In contrast to the ORM's domain-centric mode of usage, the SQL Expression Language provides a schema-centric usage paradigm.

- **Read this first:** [SQL Expression Language Tutorial](#)
- **All the Built In SQL:** [SQL Expression API](#)
- **Engines, Connections, Pools:** [Engine Configuration](#) | [Connections, Transactions](#) | [Connection Pooling](#)
- **Schema Definition:** [Overview](#) | [Tables and Columns](#) | [Database Introspection \(Reflection\)](#) | [Insert/Update Defaults](#) | [Constraints and Indexes](#) | [Using Data Definition Language \(DDL\)](#)
- **Datatypes:** [Overview](#) | [Building Custom Types](#) | [API](#)
- **Core Basics:** [Overview](#) | [Runtime Inspection API](#) | [Event System](#) | [Core Event Interfaces](#) | [Creating Custom SQL Constructs](#) |

Dialect Documentation

The **dialect** is the system SQLAlchemy uses to communicate with various types of DBAPIs and databases. This section describes notes, options, and usage patterns regarding individual dialects.

[Index of all Dialects](#)

