



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!

A RELATIONAL DATABASE IS ONE THAT STORES DATA IN TABLES. THE RELATIONSHIP BETWEEN EACH DATA POINT IS CLEAR AND SEARCHING THROUGH THOSE RELATIONSHIPS IS RELATIVELY EASY. THE RELATIONSHIP BETWEEN TABLES AND FIELD TYPES IS CALLED A SCHEMA. FOR RELATIONAL DATABASES, THE SCHEMA MUST BE CLEARLY DEFINED. LET'S LOOK AT AN EXAMPLE

The diagram illustrates a relational database schema with three tables:

- Purchase table:** Contains columns for Transaction ID, Customer ID, Product ID, and Purchase date.
- Customer table:** Contains columns for Customer ID, Customer name, and Address.
- Product table:** Contains columns for Product ID, Name, and Price.

Relationships are established through foreign keys:

- The Customer ID column in the Purchase table is linked to the Customer ID column in the Customer table.
- The Product ID column in the Purchase table is linked to the Product ID column in the Product table.

| Transaction ID | Customer ID | Product ID | Purchase date |
|----------------|-------------|------------|---------------|
| 1112 | 24221 | 8977 | 03-22-2010 |
| 1113 | 24222 | 8978 | 03-22-2010 |
| 1114 | 24223 | 8979 | 03-22-2010 |

| Customer ID | Customer | Address |
|-------------|----------|------------------|
| 24221 | Bob | 123 East street |
| 24222 | Alice | 223 Main street |
| 24223 | Martha | 465 North street |

| Product ID | Name | Price |
|------------|--------|-------|
| 8977 | Banana | .79 |
| 8978 | TV | 400 |
| 8979 | Watch | 50 |



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!

Non-relational databases are also known as NoSQL databases which stands for “Not Only SQL.” Where relational databases only use SQL, non-relational databases can use other types of query language.

There are four different types of NoSQL databases.

- Document-oriented databases – Also known as a document store, this database is designed for storing, retrieving and managing document-oriented information. Document databases usually pair each key with a complex data structure (called a document).
- Key-Value Stores – This is a database that uses different keys where each key is associated with only one value in a collection. Think of it as a dictionary. This is one of the simplest database types among NoSQL databases.
- Wide-Column Stores – this database uses tables, rows, and columns, but unlike a relational database, the names and format of the columns can vary from row to row in the same table.
- Graph Stores – A graph database uses graph structures for semantic queries with nodes, edges, and properties to represent and store data.

•

Non-relational databases are becoming more popular as more and more businesses begin to leverage big data for analysis and reporting. Since critical data doesn't always fit well into a pre-defined schema, NoSQL databases allow more flexibility.

| Key | Document |
|------|---|
| 1001 | { "CustomerID": 99, "OrderItems": [{ "ProductID": 2010, "Quantity": 2, "Cost": 520 }, { "ProductID": 4365, "Quantity": 1, "Cost": 18 }], "OrderDate": "04/01/2017" } |
| 1002 | { "CustomerID": 220, "OrderItems": [{ "ProductID": 1285, "Quantity": 1, "Cost": 120 }], "OrderDate": "05/08/2017" } |



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!

STRUCTURED QUERY LANGUAGE

- can execute queries against a database
- can retrieve data from a database
- can insert records in a database
- can update records in a database
- can delete records from a database
- can create new databases
- can create new tables in a database
- can create stored procedures in a database
- can create views in a database
- can set permissions on tables, procedures, and views

```
SELECT CustomerName, City  
FROM Customers;
```



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!

OBJECT-RELATIONAL-MAPPER

`customers.select()`

Object-relational-mapping is the idea of being able to write queries, using the object-oriented paradigm of your preferred programming language - interact with our database using our language of choice instead of SQL.

Why?

- abstracts away the database system so that switching from MySQL to PostgreSQL is easy.
- Depending on the ORM you get a lot of advanced features out of the box, such as support for transactions, connection pooling, migrations, seeds, streams
- Many of the queries you write will perform better if you are not SQL master



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!

MODELS

A model is the single, definitive source of information about your data. It contains the essential fields and behaviors of the data you're storing. Generally, each model maps to a single database table.

```
customers = sqlalchemy.Table(  
    "customers",  
    metadata,  
    sqlalchemy.Column("id", sqlalchemy.Integer, primary_key=True),  
    sqlalchemy.Column("customer", sqlalchemy.String),  
    sqlalchemy.Column("address", sqlalchemy.String),  
)
```



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!

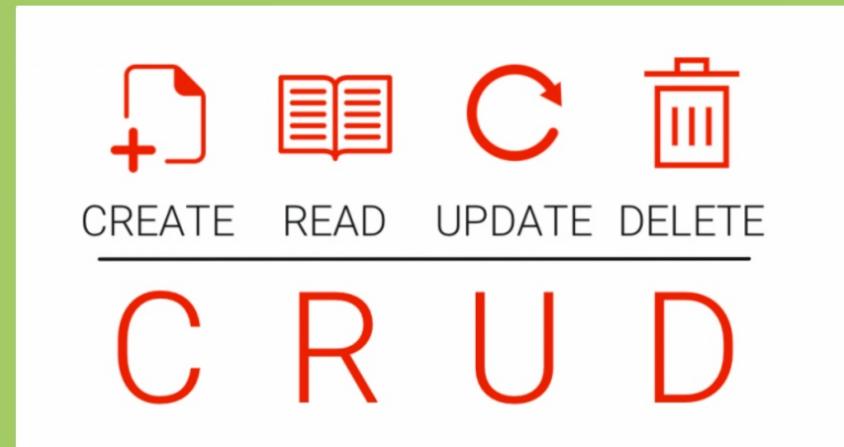
TASK 6

- In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage
- Alternate words are sometimes used when defining the basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of



TASK 6

- In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage
- Alternate words are sometimes used when defining the basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of



customers.insert().values(customer="Jane", address="known street 2")

TASK 6

- In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage
- Alternate words are sometimes used when defining the basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of



customers.insert().values(customer="Jane", address="known street 2")

customers.select()

TASK 6

- In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage
- Alternate words are sometimes used when defining the basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of



`customers.insert().values(customer="Jane", address="known street 2")`

`customers.select()`

`customers.update().values(name="Sarah")`

TASK 6

- In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage
- Alternate words are sometimes used when defining the basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of

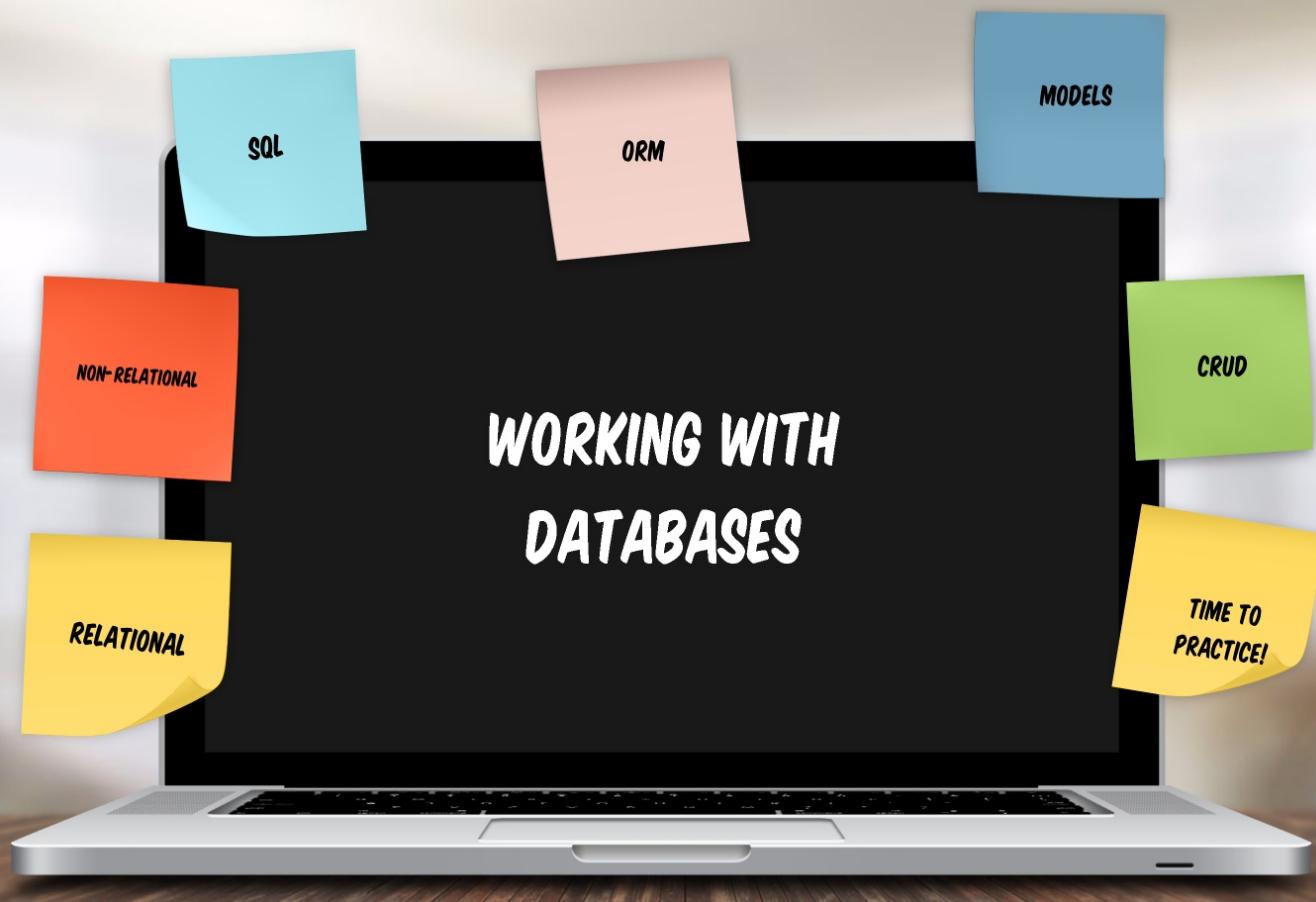


customers.insert().values(customer="Jane", address="known street 2")

customers.select()

customers.update().values(name="Sarah")

customers.delete()



BOOKS STORE

- Books
- Readers
- Alembic migrations
- Different type of relationships



WORKING WITH DATABASES

SQL

ORM

MODELS

NON-RELATIONAL

RELATIONAL

CRUD

TIME TO
PRACTICE!