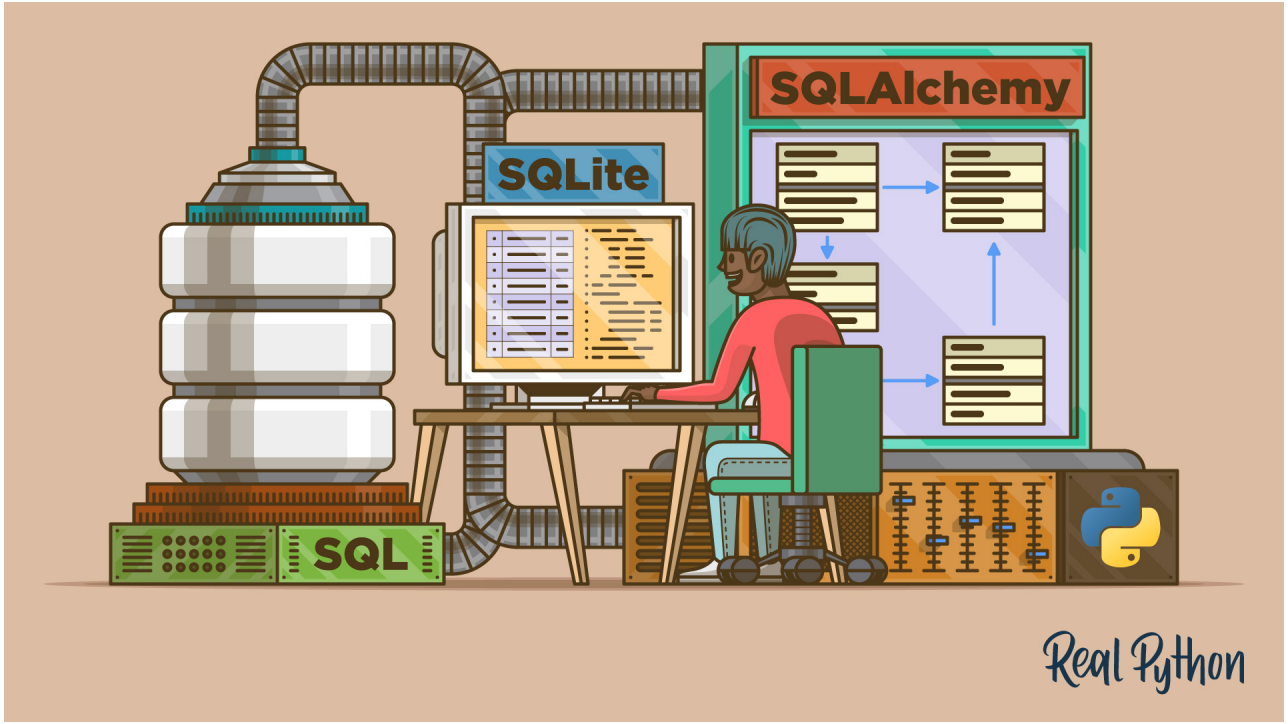
DATA MANAGEMENT WITH PYTHON AND SQLAlchemy



YOU WILL LEARN ABOUT:

1. SQL language for database access
2. SQLAlchemy:
    i. Using raw SQL
    ii. Using statements
    iii. ORM

VERSIONS

Note:

- Sample code tested using:
  - Python 3.10
  - SQLAlchemy 1.4.36
  - SQLite 3.37.0

OVERVIEW

- Software is all about manipulating data
- Data frequently needs to persist between programs
- Flat files provide a simple mechanism for storing data but are often limited
- Databases give you more power to express items and their relationships
- Most relational databases support SQL for managing data
- SQLite is a popular single file database engine that comes with a command line control program
- SQLAlchemy is a third-party Python library that abstracts SQL through raw methods, statements, and Object Relational Mapping (ORM)

NEXT UP...

Data storage with CSV

TABLE OF CONTENTS

1. **Overview**
2. **Flat Files**
3. Relational Databases
4. SQLAlchemy: Core Text
5. SQLAlchemy: Core Statements
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

DATA STORAGE

- Memory is temporary
- Data storage allows for the use and adaptation of data between multiple runs of a program
- Enables manipulation of more data than can be held in memory
- Flat file storage:
  - Text based
  - (Usually) Human readable
  - Common formats:
    - CSV
    - JSON
    - XML

CSV LIMITATIONS

- Repeated Data
  - Stephen King shows up in 4 separate lines
- Everything is from a single perspective
  - All about the book
  - How do you add an author attribute?
- No relationships
  - Richard Bachman is Stephen King
  - Detecting "*It*" has **one** author but **two** publishers is matter of code

NEXT UP...

Relational Databases

TABLE OF CONTENTS

1. **Overview**
2. **Flat Files**
3. **Relational Databases**
4. SQLAlchemy: Core Text
5. SQLAlchemy: Core Statements
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

RELATIONAL DATABASE

- Data is a series of tuples (fields)
- Relationships can be defined between the tuples
- **Primary Key**: unique identifier for a tuple
    - Often an automatically incrementing integer
- **Foreign Key**: indicates a relationship between this tuple and another through the storage of the other's Primary Key

RELATIONAL DATABASE

Authors

PK	First Name	Last Name	Aliases
1	Stephen	King	2
2	Richard	Bachman	1

Books

PK	Title	Author
1	It	1
2	Dead Zone	1
3	Runningman	2

SQL

- **Structure Query Language**
- Declarative language: describes what is to be created rather than how
- Supported by most relational databases
  - Common base, but some variations
- Database responsible for storage mechanism

SQLite

- Small, self-contained SQL database engine
- Most used database engine in the world
- Command-line tool
- Installation:

    https://sqlite.org/download.html

NEXT UP...

SQLAlchemy Core

TABLE OF CONTENTS

1. **Overview**
2. **Flat Files**
3. **Relational Databases**
4. **SQLAlchemy: Core Text**
5. SQLAlchemy: Core Statements
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

SQLAlchemy

- Popular third-party library for accessing and managing databases
- Two parts: **Core** and **ORM**
- Supports many databases:
  - SQLite
  - Postgresql
  - MySQL
  - Oracle
  - MS-SQL
  - ... and more

ALWAYS USE PARAMETERS

- When using `text()` do not build strings dynamically

```
text(f"INSERT INTO person (last_name, ) VALUES ({ln}, )")
```

```
text("INSERT INTO person (last_name, ) VALUES (:ln, )")
```

- SQL Injection is #3 in the OWASP Top 10, 2021

NEXT UP...

SQLAlchemy Statements

TABLE OF CONTENTS

1. **Overview**
2. **Flat Files**
3. **Relational Databases**
4. **SQLAlchemy: Core Text**
5. **SQLAlchemy: Core Statements**
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

SQLAlchemy STATEMENTS

- Statements abstract some of the SQL
- Python error feedback instead of SQL error feedback
- Tied tightly to SQL

NEXT UP...

SQLAlchemy ORM

TABLE OF CONTENTS

1. Overview
2. Flat Files
3. Relational Databases
4. SQLAlchemy: Core Text
5. SQLAlchemy: Core Statements
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

SQLAlchemy ORM

- **Object Relational Mapping**
- Python objects map to tables in database
- Queries result in lists of objects
- Object properties represent columns in the database
- Foreign keys become references to other objects
- Built on top of the SQLAlchemy Core library

NEXT UP...

Many to many relationships

TABLE OF CONTENTS

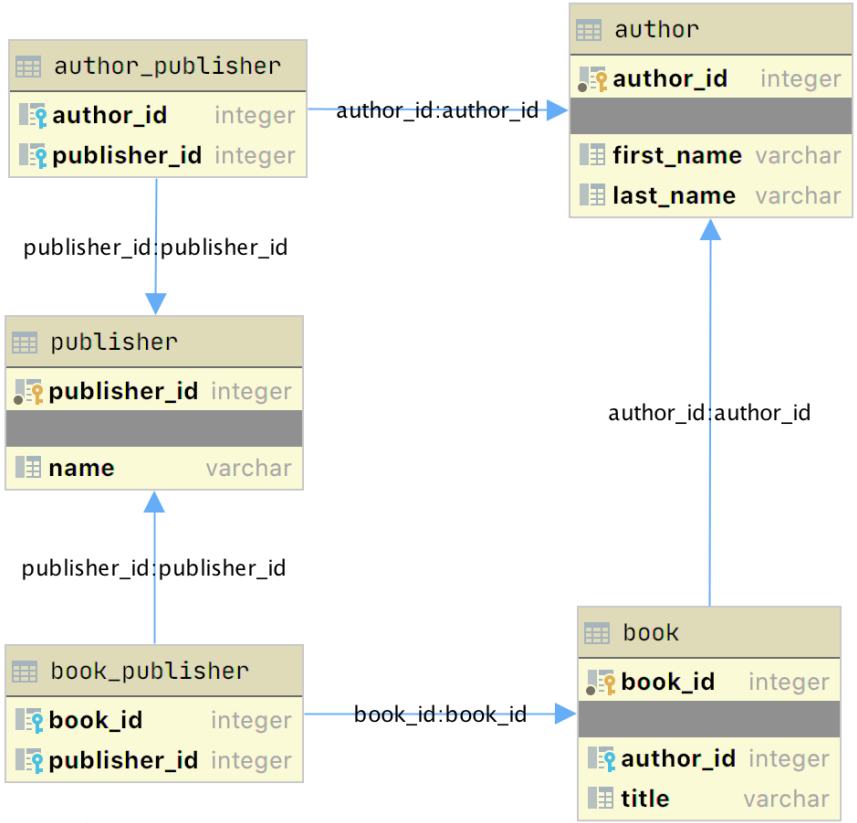
1. Overview
2. Flat Files
3. Relational Databases
4. SQLAlchemy: Core Text
5. SQLAlchemy: Core Statements
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

MANY TO MANY RELATIONSHIPS

- Foreign keys only provide one-to-many relationships:
  - Author has many books: book has single FK to Author
- Many to Many relationships use a secondary table
  - Contains just mappings between objects
  - Publisher has a relationship with books and another relationship with authors

ENTITY RELATIONSHIP DIAGRAMS

ENTITY RELATIONSHIP DIAGRAMS



NEXT UP...

Summary

TABLE OF CONTENTS

1. Overview
2. Flat Files
3. Relational Databases
4. SQLAlchemy: Core Text
5. SQLAlchemy: Core Statements
6. SQLAlchemy: ORM
7. Many To Many Relationships
8. Summary

SUMMARY

- Databases provide a powerful way of storing data and the relationships between datum
- SQL is implemented by most relational databases and provides the ability to:
  - Create & manage data structures
  - Insert data
  - Query data
  - Remove data
- SQLite is a popular self-contained database engine with a command line based query tool

SUMMARY

- SQLAlchemy is a third-party library that abstracts SQL concepts
- Provides a single interface to a number of different databases
- Two components: Core and ORM
- Core:
  - Supports raw SQL with parameters, and
  - Function based interfaces
  - Provides results in lists and dictionaries

SUMMARY

- ORM:
  - Use Python classes and objects to abstract database tables
  - Each object maps to a row in a table
  - Query mechanisms return lists of objects
  - Relationships between tables are managed as object references

SUMMARY

- Primary key is a unique identifier for a row in the database
- Foreign key indicates a relationship by pointing to the PK of another row, possibly in a different table
- Many-to-many relationships can be expressed through tables containing only PKs and FKs
- Entity Relationship Diagrams (ERDs) express the objects in a database and their interrelationships

FURTHER INVESTIGATION

- SQLite:
  https://sqlite.org/
- SQLAlchemy:
  https://www.sqlalchemy.org/
- SQL:
  https://www.khanacademy.org/computing/computer-programming/sql/
- Introduction to Python SQL Libraries:
  https://realpython.com/python-sql-libraries/
- Build a Contact Book with Python, PyQt, and SQLite:
  https://realpython.com/python-contact-book/

FURTHER INVESTIGATION

- Python REST APIs with Flask, Connexion, and SQLAlchemy:
  https://realpython.com/flask-connexion-rest-api/
- Django for Web Development:
  https://realpython.com/learning-paths/django-web-development/

Thanks!

Dankie ju faleminderit faleminderit شکرا Grazias Շնորհակալություն Sağ ol eskerrik asko Дзякуй তোমাকে ধন্যবাদ hvala trugéré
благодаря Akeva Chezu ba gràcies Salamat zikomo 谢谢 hvala děkuji Tak dank u Dankon aitäh takk fyri salamat kiitos Merci
Grazas დიდი მადლობა Danke σας ευχαριστώ આભાર Mèsi poutèt ou Na gode Mahalo תודה Dhanyawaad köszönöm þakka þér
Daalụ terima kasih Go raibh maith agat grazie ありがとう matur nuwun ಧನ್ಯವಾದಗಳು ស្មមអរគុណអ្នក Kamsahamnida ຂໍຂອບໃຈທ່ານ
Ngiyabonga paldies ačiū ви благодариме m aotr Te m Gra zi kie ru ru Dhanyawaadh Welálin баярлалаа barka
Ahéhee' Dhanyabaad miigwetch manana شکریہ شما dziękuję obrigado ਤੁਹਾਡਾ ਧੰਨਵਾਦ mulțumesc спасибо tapadh leibh хвала
ďakujem hvala Waad ku mahadsan tahay Gracias Asante Tack Salamat rahmat நன்றி ధన్యవాదాలు ขอบคุณ tualumba teşekkür
ederim Спасибі آپ کا شکریہ rahmat cảm ơn bạn Diolch yn fawr אַ דאַנק Balika o ṣeun