Vitor Amadeu Souza

Introdução a

SQLite

Com exemplos práticos em Python

Volume I

© 2024 by Cerne Tecnologia e Treinamento Ltda.

© 2024 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

Julho de 2024

Direitos reservados por:
Cerne Tecnologia e Treinamento Ltda
Produção: Cerne Tecnologia e Treinamento
E-mail da Empresa: cerne@cerne-tec.com.br
Home Page: www.cerne-tec.com.br.com.br
Atendimento ao Consumidor: sac@cerne-tec.com.br
Contato com o Autor: vitor@cerne-tec.com.br



FEITO NO BRASIL

“Recompensou-me o Senhor conforme a minha justiça,

retribuiu-me conforme a pureza das minhas mãos.”

Sl 18:20

Kits Didáticos e Gravadores da Cerne Tecnologia

A Cerne tecnologia têm uma linha completa de aprendizado para os microcontroladores da família PIC, 8051, Holtek, dsPIC, ARM, STM etc. Veja os detalhes de um kit na figura abaixo:



Kit Cerne PIC12F

- Microcontrolador PIC12F675
- Botões
- Leds
- Gravação ICSP
- E muito mais!

Uma linha completa de componentes para o desenvolvimento de seus projetos eletrônicos como displays, PICs, botões, leds, cristais dentre outros. Visite a nossa página na Internet, no endereço www.cerne-tec.com.br e conheça melhor nossos serviços e produtos.

Cerne
Conhecimento para o Desenvolvimento

Sumário

Introdução.....6

Capítulo I – SQLite.....7

1. Banco de dados.....7
2. SQLite.....11
3. Replit.....13
4. Tipos de dados.....15
5. SELECT.....19
6. LIMIT.....22
7. WHERE.....23

Introdução

A proposta desta literatura é apresentar o acesso ao banco de dados SQLite usando para isso a plataforma on-line Replit. Em cada parte desta série de livros uma parte do SQLite será apresentada, no qual o tema abordado está no sumário de cada obra.

Em todos os casos, exemplos práticos escritos em Python procuram envolver a base de dados com programação, facilitando desta forma a compreensão deste assunto.

Capítulo I

SQLite

1. Banco de dados

Um banco de dados é uma coleção organizada de informações ou dados, que são armazenados e gerenciados de forma a permitir fácil acesso, gerenciamento e atualização. Bancos de dados são essenciais para armazenar grandes volumes de dados de forma estruturada, de modo que possam ser consultados e manipulados de maneira eficiente. Acompanhe as principais características de um banco de dados:

1. Organização e Estrutura:

- Tabelas: Dados são armazenados em tabelas que contêm linhas (registros) e colunas (campos). Cada tabela é projetada para armazenar informações sobre um tipo específico de entidade (por exemplo, clientes, produtos, transações).

- Schemas: Define a estrutura de uma base de dados, incluindo tabelas, campos, tipos de dados, e as relações entre os diferentes dados.

2. Acesso e Manipulação de Dados:

- Linguagem SQL: A linguagem de consulta estruturada (SQL) é usada para inserir, atualizar, excluir e consultar dados.

- CRUD: Operações básicas suportadas por bancos de dados são Create (criar), Read (ler), Update (atualizar) e Delete (excluir).

3. Integridade dos Dados:

- Integridade Referencial: Garante que as relações entre tabelas permaneçam consistentes.

- Integridade de Domínio: Assegura que os dados nos campos estejam dentro de um conjunto específico de valores permitidos.

- Integridade de Entidade: Garante que cada tabela tenha uma chave primária única.

4. Segurança:

- Controle de Acesso: Implementa permissões e autenticação para proteger dados sensíveis.

- Auditoria e Logs: Registra operações realizadas no banco de dados para monitoramento e auditoria.

5. Confiabilidade e Recuperação:

- Transações ACID: Garantem Atomicidade, Consistência, Isolamento e Durabilidade, essenciais para a confiabilidade das operações de banco de dados.

- Backups e Restauração: Métodos para fazer backup dos dados e restaurá-los em caso de falha.

6. Performance e Eficiência:

- Índices: Estruturas de dados que melhoram a velocidade de consulta de tabelas.

- Normalização: Processo de organização de dados para reduzir a redundância e melhorar a integridade.

- Desnormalização: Às vezes utilizada para melhorar a performance de leitura, armazenando dados redundantes.

7. Escalabilidade:

- Escalabilidade Vertical: Aumentar a capacidade do hardware do servidor para melhorar o desempenho.

- Escalabilidade Horizontal: Distribuir a carga de trabalho entre múltiplos servidores.

Tipos de Bancos de Dados

1. Relacionais (RDBMS):

- Usam tabelas para organizar dados.
- Suportam SQL.
- Exemplos: MySQL, PostgreSQL, Oracle Database.

2. Não Relacionais (NoSQL):

- Incluem várias categorias como chave-valor, documentos, colunas e grafos.
- Usados para grandes volumes de dados não estruturados ou semi-estruturados.
- Exemplos: MongoDB, Cassandra, Redis.

3. Banco de Dados em Memória:

- Armazenam dados na memória principal para acesso rápido.
- Exemplos: Redis, Memcached.

4. Banco de Dados Orientados a Objetos:

- Armazenam dados na forma de objetos, como na programação orientada a objetos.
- Exemplos: db4o, ObjectDB.

5. Bancos de Dados Distribuídos: