

Projeto de Bloco: Fundamentos de Dados

Professor: Alcione Santos Dolavale



Kaike Torres da silva

30.11.2024

Análise e Desenvolvimento de Sistemas (ADS)

INTRODUÇÃO

Este projeto, TP2 - Projeto de Bloco de Fundamento de Dados, tem como foco a criação e o relacionamento de três tabelas: Funcionários, Cargos e Departamentos. Cada uma delas armazena informações essenciais como nome, cargo, salário e o departamento dos funcionários. Para enriquecer o conjunto de dados, foi adicionado um campo extra em cada tabela.

As tabelas foram construídas e relacionadas no MySQL, utilizando o MySQL Workbench para sua execução e consulta. Foram inseridos registros suficientes para viabilizar consultas que mostram informações como a descrição do cargo e o nome do departamento, evidenciando o uso de chaves estrangeiras.

RESULTADOS

1. Criação e Estrutura do Banco de Dados

A primeira parte do código é responsável por criar um banco de dados SQLite chamado `dados.db`, contendo três tabelas:

- **TB_EVENTOS:** Armazena informações básicas sobre os eventos, como nome e faixa etária.

```
CREATE TABLE TB_EVENTOS (  
    id_evento INT PRIMARY KEY,  
    nome_evento VARCHAR(255) NOT NULL,  
    faixa_etaria VARCHAR(50)  
);
```

Exemplo de dados:

id_evento	nome_evento	faixa_etaria
1	Festival de Verão	18-35
2	Conferência Tech 2024	25-40

- **TB_DADOS_EVENTOS:** Contém detalhes sobre as datas e localização dos eventos.

```
CREATE TABLE TB_DADOS_EVENTOS (
    id_evento INT,
    data_evento DATE,
    local_evento VARCHAR(255),
    FOREIGN KEY (id_evento) REFERENCES TB_EVENTOS(id_evento)
);
```

Exemplo de dados:

id_evento	data_evento	local_evento
1	2024-06-15	Arena São Paulo
2	2024-08-20	Centro de Convenções

- **TB_METADADOS_EVENTOS:** Armazena informações adicionais (metadados) sobre os eventos.

```
CREATE TABLE TB_METADADOS_EVENTOS (
    id_evento INT,
    descricao_evento TEXT,
    tipo_evento VARCHAR(100),
    FOREIGN KEY (id_evento) REFERENCES TB_EVENTOS(id_evento)
);
```

Exemplo de dados:

id_evento	descricao_evento	tipo_evento
1	Festival de música com artistas locais	Música
2	Conferência inovações te	Tecnologia

O script garante que as tabelas sejam criadas no banco de dados sempre que executado, removendo versões anteriores das tabelas se já existirem.

2. Classes de Entidade

O script utiliza três classes para representar e manipular os dados de eventos:

- **EventoEntity**: Representa um evento e permite salvar informações como nome e faixa etária na tabela **TB_EVENTOS**.
- **DadoEventoEntity**: Representa os detalhes de um evento, como data de início e término, além da localização. Esses dados são armazenados na tabela **TB_DADOS_EVENTOS**.
- **MetadadoEventoEntity**: Representa metadados do evento (informações adicionais extraídas do conteúdo HTML) e é armazenado na tabela **TB_METADADOS_EVENTOS**.

3. Extração de Dados do Site

O script faz uma requisição HTTP para o site **sampaingressos.com.br** e utiliza o **BeautifulSoup** para fazer o parsing do HTML e extrair os dados de cada evento.

Os dados extraídos incluem:

- Nome do Evento: Localizado em uma tag **** com a classe **titulo**.
- Localização: Extraída de uma tag **** com a classe **local**.
- Temporada (datas de início e término): Extraída de uma tag **** com a classe **temporada**, e convertida para o formato **date**.
- Faixa Etária: Determinada a partir do atributo **src** de uma tag **** dentro de uma tag **** com a classe **rec_etaria_card**.
- Metadados: Informações adicionais extraídas do conteúdo do evento.

Esses dados são então salvos nas tabelas do banco de dados, utilizando as classes previamente descritas.

4. Consulta de Dados

Uma consulta SQL foi implementada para retornar detalhes sobre os eventos, combinando informações das tabelas **TB_EVENTOS** e **TB_DADOS_EVENTOS** por meio de um **JOIN**.

O código realiza a seguinte consulta:

```
SELECT
    eve.Id,
    eve.Nome,
    eve.Faixa_Etaria,
    dad.DataDe,
    dad.DataAte,
    dad.Localizacao
FROM TB_EVENTOS eve
JOIN TB_DADOS_EVENTOS dad ON eve.Id = dad.Id_Evento;
```

A consulta retorna os seguintes dados para cada evento:

- Id do Evento
- Nome do Evento
- Faixa Etária
- Data de Início e Término
- Localização

Esses dados são apresentados ao usuário no formato:

```
Id Evento: <id>
Nome Evento: <nome>
Faixa Etária: <faixa_etaria>
Data de Início: <data_inicio>
Data de Término: <data_termino>
Localização: <localizacao>
```

5. Considerações Finais

Este script oferece uma solução automatizada para coletar, processar e armazenar informações sobre eventos em um banco de dados SQLite. Ele integra tecnologias de web scraping (usando `requests` e `BeautifulSoup`), manipulação de banco de dados SQLite (com `sqlite3`), e formatação de datas (usando `datetime`). A consulta gerada permite visualizar de maneira clara as informações dos eventos, facilitando futuras análises.

Possíveis melhorias:

- Implementação de tratamento de erros mais robusto para casos em que o site esteja temporariamente fora do ar ou o formato do HTML seja alterado.
- Adição de funcionalidades para atualizar eventos existentes no banco de dados, evitando duplicação de dados.