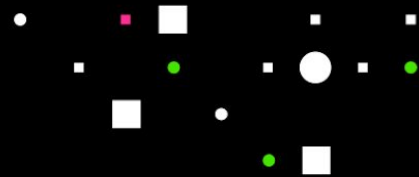


treinadev

Desafio

Etapa 2 – Turma 3

CAMPUS
CODE



Desafio

O Censo Demográfico é a mais complexa operação estatística realizada por um país, sobretudo quando ele tem dimensões continentais como o Brasil, com 8.515.692,27 km², distribuídos em um território heterogêneo, muitas vezes de difícil acesso, composto por 5.565 municípios e uma população de 190.755.799 habitantes.

Compilamos a informação do último censo disponível num [arquivo CSV](#) listando as populações de cada município das 27 unidades federativas. E nosso desafio vai usar essas informações para criar uma app, só que em uma versão *old school*. Então prepare seus conhecimentos em Ruby puro e vamos lá!

A ideia aqui é criar uma aplicação totalmente Ruby com auxílio de algumas gems para testes e conexão com banco de dados, entre outras. Usando um CSV vamos carregar os dados populacionais dos municípios do Brasil e fazer algumas consultas a essa base para obtermos dados como: municípios mais populosos de uma Unidade Federativa, população média de uma Unidade Federativa e busca de municípios.

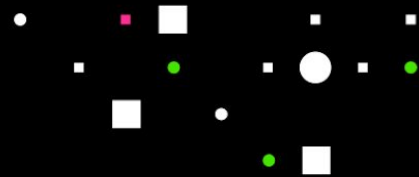
Levando isso em conta a aplicação poder ser executada de duas formas, mas fique à vontade caso encontrar algo intermediário.

1. Como uma aplicação que fica em execução de forma constante e conta com um menu para o usuário escolher a consulta a ser realizada ou sair da aplicação.
2. Como um CLI, similar a comandos como ``git`` e ``rails`` que você executa em seu Terminal. Nesse formato você deve ter uma opção ``--help`` que traz todas as opções possíveis para execução (todas as consultas e como o usuário deve passar os parâmetros necessários em cada consulta).

Dividimos as tarefas em alguns passos, mas você pode fazer o seu próprio planejamento usando as ferramentas que quiser, aqui temos uma sugestão.

Fase 1

O primeiro passo é criar a fundação da nossa aplicação. Todos os dados que precisamos para fazer as consultas estão disponíveis em um arquivo CSV (link no início deste documento). Como a leitura de arquivos é uma transação lenta e os dados que estamos manipulando não vão sofrer alterações, vamos começar criando um banco de dados a partir do arquivo. Essa operação deve ser feita ao iniciar a aplicação pela primeira vez. Daí em diante, esperamos que o código verifique se o banco de dados já está populado e pule esta etapa.



Resumindo, para começar você deve criar uma aplicação Terminal que, ao ser inicializada:

1. Crie uma estrutura de conexão a um banco de dados SQLite
2. Crie e popule tabelas para armazenar todas as UFs, todos os municípios e suas populações.

Lembre-se de armazenar somente os dados essenciais para o uso dentro da sua aplicação. A organização das tabelas fica a seu critério.

O arquivo CSV deve ser armazenado junto com sua aplicação em um diretório a seu critério

Fase 2

Agora que temos os dados carregados numa base de dados podemos fazer algumas consultas para começar a esquentar os motores.

1. Ao escolher uma UF, listar os 10 municípios com maior população daquela UF
2. Buscar um município por nome e listar a população daquele município
3. Incluir na busca a possibilidade de buscar por código (disponível no CSV) e listar a população daquele município

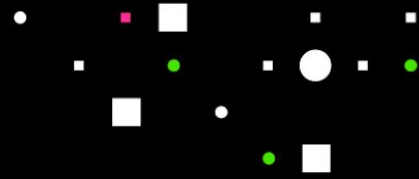
Fase 3

Agora que estamos executando queries no nosso banco, podemos começar a utilizar outros recursos como ORDER, AVERAGE, SUM, entre outros.

1. Permitir que a busca seja feita também por nome e código da UF, mostrando a soma da população total dos municípios
2. Ao escolher uma UF, listar um ranking com as 10 cidades mais populosas
3. Ao escolher uma UF, listar a média populacional dos municípios

Utilizando o projeto

Foque na qualidade e organização do seu código, assim como na facilidade de uso das interfaces e na qualidade da exibição dos resultados de cada consulta. Então, dedique algum tempo para formatar minimamente as tabelas solicitadas.



Dicas

Esse projeto possui desafios em temas variados, partindo de lógica e organização de código até banco de dados e leitura de arquivos. Por isso, veja algumas dicas que podem te ajudar:

1. Não existe uma única ordem ou forma correta para resolver todos os problemas propostos, como foi dito no começo deste texto. O que recomendamos é:
 - Leia toda a proposta mais de uma vez, fazendo anotações ao seu jeito;
 - Abra o arquivo CSV da população para conhecer sua estrutura e seus dados.

Ao final de tudo isso, faça um planejamento e comece a escrever seu código.

2. Não esqueça de criar um repositório Git em alguma plataforma online e compartilhar conosco assim que possível.
3. Documente, no arquivo README.md, na raiz do seu projeto, como funciona sua aplicação e decisões tomadas durante a implementação. Não deixe de citar `gems` e outros recursos utilizados para resolver os problemas que encontrou ao longo do projeto.
4. Foque em evoluir seus conhecimentos de Ruby puro, Orientação a Objetos e consultas a banco de dados. Mas você pode utilizar gems quando necessário, como a gem sqlite3 para conexão com banco de dados.
5. Busque 100% de cobertura e extraia suas classes de forma que facilite a criação de testes e cenários com RSpec.

treinadev

é um programa gratuito de
formação de devs da Campus Code

Patrocínio:

VINDI

R E
B A
S E

PORTAL
solar

smartfit

CAMPUS
CODE

Al. Santos, 1293, conj. 73
Cerqueira César, São Paulo, SP
Telefone/Whatsapp: [\[11\] 2369-3476](tel:(11)2369-3476)

