



Comece aqui

Roteiros ▾

Tutor de IA

Login

Inscrição

[← Voltar para Projetos](#)[≡ Detalhe do projeto](#)[🔍 Soluções da comunidade](#)

Linguagem De Programação

CLI

Sistema De Arquivos

Iniciante

Rastreador de tarefas

Crie um aplicativo CLI para acompanhar suas tarefas e gerenciar sua lista de tarefas.

Comece a criar, envie a solução e obtenha feedback da comunidade.

[Comece a trabalhar](#)

-

2

[Enviar solução](#)

-

[👍 5 votos positivos](#)

-

[👍 10 votos positivos](#)

O rastreador de tarefas é um projeto usado para rastrear e gerenciar suas tarefas. Nesta tarefa, você criará uma interface de linha de comando (CLI) simples para rastrear o que precisa fazer, o que fez e no que está trabalhando no momento. Este projeto ajudará você a praticar suas habilidades de programação, incluindo trabalhar com o sistema de arquivos, lidar com entradas do usuário e criar um aplicativo CLI simples.

Requisitos

O aplicativo deve ser executado a partir da linha de comando, aceitar ações e entradas do usuário como argumentos e armazenar as tarefas em um arquivo JSON. O usuário deve ser capaz de:

- Adicionar, atualizar e excluir tarefas
- Marcar uma tarefa como em andamento ou concluída
- Listar todas as tarefas
- Listar todas as tarefas que são feitas
- Listar todas as tarefas que não foram concluídas
- Listar todas as tarefas que estão em andamento

Aqui estão algumas restrições para orientar a implementação:

- Você pode usar qualquer linguagem de programação para construir este projeto.
- Use argumentos posicionais na linha de comando para aceitar entradas do usuário.
- Use um arquivo JSON para armazenar as tarefas no diretório atual.
- O arquivo JSON deve ser criado se não existir.
- Use o módulo de sistema de arquivos nativo de sua linguagem de programação para interagir com o arquivo JSON.
- Não use bibliotecas ou estruturas externas para criar este projeto.
- Certifique-se de lidar com erros e casos extremos normalmente.

Exemplo

A lista de comandos e seu uso é fornecida abaixo:

```
# Adding a new task
task-cli add "Buy groceries"
# Output: Task added successfully (ID: 1)

# Updating and deleting tasks
task-cli update 1 "Buy groceries and cook dinner"
task-cli delete 1

# Marking a task as in progress or done
task-cli mark-in-progress 1
task-cli mark-done 1

# Listing all tasks
task-cli list

# Listing tasks by status
task-cli list done
task-cli list todo
task-cli list in-progress
```

Propriedades da tarefa

Cada tarefa deve ter as seguintes propriedades:

- id: Um identificador exclusivo para a tarefa
- description: Uma breve descrição da tarefa
- status: O status da tarefa (, , todo in-progress done)
- createdAt: A data e a hora em que a tarefa foi criada

- `updatedAt`: a data e a hora em que a tarefa foi atualizada pela última vez

Certifique-se de adicionar essas propriedades ao arquivo JSON ao adicionar uma nova tarefa e atualizá-las ao atualizar uma tarefa.

Começando

Aqui estão algumas etapas para ajudá-lo a começar a usar o projeto da CLI do Rastreador de Tarefas:

Configurar seu ambiente de desenvolvimento

- Escolha uma linguagem de programação com a qual você se sinta confortável (por exemplo, Python, JavaScript, etc.).
- Certifique-se de ter um editor de código ou IDE instalado (por exemplo, VSCode, PyCharm).

Inicialização do projeto

- Crie um novo diretório de projeto para a CLI do Rastreador de Tarefas.
- Inicialize um sistema de controle de versão (por exemplo, Git) para gerenciar seu projeto.

Implementando recursos

- Comece criando uma estrutura CLI básica para lidar com as entradas do usuário.
- Implemente cada recurso um por um, garantindo um teste completo antes de passar para o próximo, por exemplo, implemente a adição da funcionalidade da tarefa primeiro, listando em seguida, atualizando, marcando como em andamento, etc.

Teste e depuração

- Teste cada recurso individualmente para garantir que funcionem conforme o esperado. Examine o arquivo JSON para verificar se as tarefas estão sendo armazenadas corretamente.
- Depure todos os problemas que surgirem durante o desenvolvimento.


Finalizando o Projeto

- Certifique-se de que todos os recursos sejam implementados e testados.

- Limpe seu código e adicione comentários quando necessário.
- Escreva um bom arquivo leia-me sobre como usar a CLI do Rastreador de Tarefas.

Ao final deste projeto, você terá desenvolvido uma ferramenta prática que pode ajudar você ou outras pessoas a gerenciar tarefas com eficiência. Este projeto estabelece uma base sólida para projetos de programação mais avançados e aplicações do mundo real.

Boa codificação!

 Encontrou um erro? [Ajude-nos a melhorar.](#)

Join the Community

roadmap.sh is the 7th most starred project on GitHub and is visited by hundreds of thousands of developers every month.

Rank 7th out of 28M!

313K

GitHub Stars

★ **Star us on GitHub**
Help us reach #1

+90k every month

+1.5M

Registered Users

👤 **Register yourself**
Commit to your growth

+2k every month

35K

Discord Members

🗨️ **Join on Discord**
Join the community

[Roadmaps](#) [Best Practices](#) [Guides](#) [Videos](#) [FAQs](#) [YouTube](#)

Community created roadmaps, best practices, projects, articles, resources and journeys to help you choose your path and grow in your career.

© roadmap.sh · [Terms](#) · [Privacy](#) · [Advertise](#) ·   

THE NEW STACK

The top DevOps resource for Kubernetes, cloud-native computing, and large-scale development and deployment.

[DevOps](#) · [Kubernetes](#) · [Cloud-Native](#)