Olá! como saber se a chave API do Mistral 7B que eu gerei no OpenRouter é pago ou free?

**O ChatGPT disse:**

No OpenRouter, a chave API que você gera pode ser usada tanto para modelos gratuitos quanto pagos. No entanto, mesmo na versão **FREE**, algumas restrições podem se aplicar, como limite diário de uso ou necessidade de configurar um método de pagamento para evitar abuso.

**Como verificar se sua chave API é gratuita ou paga?**

1. **Acesse sua conta no OpenRouter**
   * Vá para 👉 <https://openrouter.ai>
   * Faça login e clique na aba de **Billing** (Faturamento).
   * Se não houver cobranças ou saldo, então você está no plano gratuito.
2. **Cheque a disponibilidade do modelo "mistral-7b-instruct"**
   * No OpenRouter, alguns modelos são **100% gratuitos**, enquanto outros exigem créditos.
   * Acesse a página de modelos no OpenRouter e verifique se o **Mistral 7B Instruct** está marcado como gratuito.
3. **Tente fazer uma requisição via terminal**
   * Execute o seguinte comando para testar sua chave:

sh

CopiarEditar

curl https://openrouter.ai/api/v1/models \

-H "Authorization: Bearer SUA\_CHAVE\_API"

* + Se a resposta mostrar "mistral-7b-instruct" na lista de modelos disponíveis, significa que você pode usá-lo.
  + Se houver um erro indicando **"insufficient credits"** ou **"payment required"**, sua conta pode precisar de créditos para acessar o modelo.

1. **Limitação do Plano Free**
   * O plano gratuito pode ter um **limite diário de requisições**. Se você ultrapassou esse limite, pode ser necessário esperar ou adicionar créditos.

Se precisar de mais ajuda, me avise! 🚀

4o

**Você disse:**

PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> -H "Authorization: Bearer SUA\_CHAVE\_API" -H : O termo '-H' não é reconhecido como nome de cmdlet, função, arquivo de script ou programa operável. Verifique a grafia do nome ou, se um caminho tiver sido incluído, veja se o caminho está correto e tente novamente. No linha:1 caractere:3 + -H "Authorization: Bearer SUA\_CHAVE\_API" + ~~ + CategoryInfo : ObjectNotFound: (-H:String) [], CommandNotFoundException + FullyQualifiedErrorId : CommandNotFoundException PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8>

**O ChatGPT disse:**

O erro acontece porque o comando curl que passei foi formatado para **Linux/macOS** e você está rodando no **Windows PowerShell**. No PowerShell, a sintaxe do curl é um pouco diferente.

**Teste correto no PowerShell**

No **PowerShell**, use o seguinte comando:

powershell

CopiarEditar

$headers = @{

"Authorization" = "Bearer SUA\_CHAVE\_API"

}

Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/models" -Headers $headers -Method Get

🔹 **Se a resposta incluir "mistral-7b-instruct"**, sua chave API está funcionando.  
🔹 **Se aparecer um erro de pagamento ou falta de créditos**, pode ser que sua conta esteja limitada.

Me diga o que aparece na resposta para te ajudar melhor! 🚀

4o

**Você disse:**

PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $headers = @{ >> "Authorization" = "Bearer SUA\_CHAVE\_API" >> } PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/models" -Headers $headers -Method Get data ---- {@{id=all-hands/openhands-lm-32b-v0.1; name=OpenHands LM 32B V0.1; created=1743613013; description=OpenHands LM v0.1... PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8>

**O ChatGPT disse:**

Ótimo! 🎉 Isso significa que sua chave API **está funcionando** e você tem acesso a alguns modelos no OpenRouter.

Agora, precisamos verificar se **o modelo "mistral-7b-instruct" está disponível** para você.

**O que fazer agora?**

Tente este comando no PowerShell para listar **todos os modelos disponíveis** na sua conta:

powershell

CopiarEditar

(Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/models" -Headers $headers -Method Get).data | Select-Object id

🔹 Se **"mistral-7b-instruct" aparecer na lista**, então você pode usá-lo.  
🔹 Se **ele não aparecer**, significa que **o modelo pode não estar disponível gratuitamente** na sua conta, ou você já atingiu o limite de uso.

Me diga o que aparece no resultado! 🚀

4o

**Você disse:**

PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> (Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/models" -Headers $headers -Method Get).data | Select-Object id id -- all-hands/openhands-lm-32b-v0.1 mistral/ministral-8b deepseek/deepseek-v3-base:free scb10x/llama3.1-typhoon2-8b-instruct scb10x/llama3.1-typhoon2-70b-instruct allenai/molmo-7b-d:free bytedance-research/ui-tars-72b:free qwen/qwen2.5-vl-3b-instruct:free google/gemini-2.5-pro-exp-03-25:free qwen/qwen2.5-vl-32b-instruct:free deepseek/deepseek-chat-v3-0324:free deepseek/deepseek-chat-v3-0324 featherless/qwerky-72b:free openai/o1-pro mistralai/mistral-small-3.1-24b-instruct:free mistralai/mistral-small-3.1-24b-instruct open-r1/olympiccoder-7b:free open-r1/olympiccoder-32b:free steelskull/l3.3-electra-r1-70b allenai/olmo-2-0325-32b-instruct google/gemma-3-1b-it:free google/gemma-3-4b-it:free google/gemma-3-4b-it ai21/jamba-1.6-large ai21/jamba-1.6-mini google/gemma-3-12b-it:free google/gemma-3-12b-it cohere/command-a openai/gpt-4o-mini-search-preview openai/gpt-4o-search-preview tokyotech-llm/llama-3.1-swallow-70b-instruct-v0.3 rekaai/reka-flash-3:free google/gemma-3-27b-it:free google/gemma-3-27b-it thedrummer/anubis-pro-105b-v1 latitudegames/wayfarer-large-70b-llama-3.3 thedrummer/skyfall-36b-v2 microsoft/phi-4-multimodal-instruct perplexity/sonar-reasoning-pro perplexity/sonar-pro perplexity/sonar-deep-research deepseek/deepseek-r1-zero:free qwen/qwq-32b:free qwen/qwq-32b qwen/qwen2.5-32b-instruct moonshotai/moonlight-16b-a3b-instruct:free nousresearch/deephermes-3-llama-3-8b-preview:free openai/gpt-4.5-preview google/gemini-2.0-flash-lite-001 anthropic/claude-3.7-sonnet:beta anthropic/claude-3.7-sonnet anthropic/claude-3.7-sonnet:thinking perplexity/r1-1776 mistralai/mistral-saba cognitivecomputations/dolphin3.0-r1-mistral-24b:free cognitivecomputations/dolphin3.0-mistral-24b:free meta-llama/llama-guard-3-8b openai/o3-mini-high deepseek/deepseek-r1-distill-llama-8b google/gemini-2.0-flash-001 google/gemini-2.0-pro-exp-02-05:free qwen/qwen-vl-plus aion-labs/aion-1.0 aion-labs/aion-1.0-mini aion-labs/aion-rp-llama-3.1-8b qwen/qwen-vl-max qwen/qwen-turbo qwen/qwen2.5-vl-72b-instruct:free qwen/qwen2.5-vl-72b-instruct qwen/qwen-plus qwen/qwen-max openai/o3-mini deepseek/deepseek-r1-distill-qwen-1.5b mistralai/mistral-small-24b-instruct-2501:free mistralai/mistral-small-24b-instruct-2501 deepseek/deepseek-r1-distill-qwen-32b:free deepseek/deepseek-r1-distill-qwen-32b deepseek/deepseek-r1-distill-qwen-14b:free deepseek/deepseek-r1-distill-qwen-14b perplexity/sonar-reasoning perplexity/sonar liquid/lfm-7b liquid/lfm-3b deepseek/deepseek-r1-distill-llama-70b:free deepseek/deepseek-r1-distill-llama-70b google/gemini-2.0-flash-thinking-exp:free deepseek/deepseek-r1:free deepseek/deepseek-r1 sophosympatheia/rogue-rose-103b-v0.2:free minimax/minimax-01 mistralai/codestral-2501 microsoft/phi-4 sao10k/l3.1-70b-hanami-x1 deepseek/deepseek-chat:free deepseek/deepseek-chat google/gemini-2.0-flash-thinking-exp-1219:free sao10k/l3.3-euryale-70b openai/o1 eva-unit-01/eva-llama-3.33-70b x-ai/grok-2-vision-1212 x-ai/grok-2-1212 cohere/command-r7b-12-2024 google/gemini-2.0-flash-exp:free meta-llama/llama-3.3-70b-instruct:free meta-llama/llama-3.3-70b-instruct amazon/nova-lite-v1 amazon/nova-micro-v1 amazon/nova-pro-v1 qwen/qwq-32b-preview:free qwen/qwq-32b-preview google/learnlm-1.5-pro-experimental:free eva-unit-01/eva-qwen-2.5-72b openai/gpt-4o-2024-11-20 mistralai/mistral-large-2411 mistralai/mistral-large-2407 mistralai/pixtral-large-2411 x-ai/grok-vision-beta infermatic/mn-inferor-12b qwen/qwen-2.5-coder-32b-instruct:free qwen/qwen-2.5-coder-32b-instruct raifle/sorcererlm-8x22b eva-unit-01/eva-qwen-2.5-32b thedrummer/unslopnemo-12b anthropic/claude-3.5-haiku:beta anthropic/claude-3.5-haiku anthropic/claude-3.5-haiku-20241022:beta anthropic/claude-3.5-haiku-20241022 neversleep/llama-3.1-lumimaid-70b anthracite-org/magnum-v4-72b anthropic/claude-3.5-sonnet:beta anthropic/claude-3.5-sonnet x-ai/grok-beta mistralai/ministral-8b mistralai/ministral-3b qwen/qwen-2.5-7b-instruct nvidia/llama-3.1-nemotron-70b-instruct:free nvidia/llama-3.1-nemotron-70b-instruct inflection/inflection-3-productivity inflection/inflection-3-pi google/gemini-flash-1.5-8b thedrummer/rocinante-12b anthracite-org/magnum-v2-72b liquid/lfm-40b meta-llama/llama-3.2-3b-instruct:free meta-llama/llama-3.2-3b-instruct meta-llama/llama-3.2-1b-instruct:free meta-llama/llama-3.2-1b-instruct meta-llama/llama-3.2-90b-vision-instruct meta-llama/llama-3.2-11b-vision-instruct:free meta-llama/llama-3.2-11b-vision-instruct qwen/qwen-2.5-72b-instruct:free qwen/qwen-2.5-72b-instruct qwen/qwen-2.5-vl-72b-instruct neversleep/llama-3.1-lumimaid-8b openai/o1-preview openai/o1-preview-2024-09-12 openai/o1-mini openai/o1-mini-2024-09-12 mistralai/pixtral-12b cohere/command-r-plus-08-2024 cohere/command-r-08-2024 qwen/qwen-2.5-vl-7b-instruct:free qwen/qwen-2.5-vl-7b-instruct sao10k/l3.1-euryale-70b google/gemini-flash-1.5-8b-exp ai21/jamba-1-5-mini ai21/jamba-1-5-large microsoft/phi-3.5-mini-128k-instruct nousresearch/hermes-3-llama-3.1-70b nousresearch/hermes-3-llama-3.1-405b openai/chatgpt-4o-latest sao10k/l3-lunaris-8b aetherwiing/mn-starcannon-12b openai/gpt-4o-2024-08-06 meta-llama/llama-3.1-405b nothingiisreal/mn-celeste-12b perplexity/llama-3.1-sonar-small-128k-online perplexity/llama-3.1-sonar-large-128k-online meta-llama/llama-3.1-8b-instruct:free meta-llama/llama-3.1-8b-instruct meta-llama/llama-3.1-405b-instruct meta-llama/llama-3.1-70b-instruct mistralai/codestral-mamba mistralai/mistral-nemo:free mistralai/mistral-nemo openai/gpt-4o-mini openai/gpt-4o-mini-2024-07-18 qwen/qwen-2-7b-instruct:free qwen/qwen-2-7b-instruct google/gemma-2-27b-it alpindale/magnum-72b google/gemma-2-9b-it:free google/gemma-2-9b-it 01-ai/yi-large ai21/jamba-instruct anthropic/claude-3.5-sonnet-20240620:beta anthropic/claude-3.5-sonnet-20240620 sao10k/l3-euryale-70b cognitivecomputations/dolphin-mixtral-8x22b qwen/qwen-2-72b-instruct mistralai/mistral-7b-instruct:free mistralai/mistral-7b-instruct mistralai/mistral-7b-instruct-v0.3 nousresearch/hermes-2-pro-llama-3-8b microsoft/phi-3-mini-128k-instruct:free microsoft/phi-3-mini-128k-instruct microsoft/phi-3-medium-128k-instruct:free microsoft/phi-3-medium-128k-instruct neversleep/llama-3-lumimaid-70b google/gemini-flash-1.5 openai/gpt-4o openai/gpt-4o:extended openai/gpt-4o-2024-05-13 meta-llama/llama-guard-2-8b neversleep/llama-3-lumimaid-8b:extended neversleep/llama-3-lumimaid-8b sao10k/fimbulvetr-11b-v2 meta-llama/llama-3-8b-instruct:free meta-llama/llama-3-8b-instruct meta-llama/llama-3-70b-instruct mistralai/mixtral-8x22b-instruct microsoft/wizardlm-2-8x22b microsoft/wizardlm-2-7b google/gemini-pro-1.5 openai/gpt-4-turbo cohere/command-r-plus cohere/command-r-plus-04-2024 sophosympatheia/midnight-rose-70b cohere/command cohere/command-r anthropic/claude-3-haiku:beta anthropic/claude-3-haiku anthropic/claude-3-opus:beta anthropic/claude-3-opus anthropic/claude-3-sonnet:beta anthropic/claude-3-sonnet cohere/command-r-03-2024 mistralai/mistral-large openai/gpt-3.5-turbo-0613 openai/gpt-4-turbo-preview nousresearch/nous-hermes-2-mixtral-8x7b-dpo mistralai/mistral-medium mistralai/mistral-small mistralai/mistral-tiny mistralai/mistral-7b-instruct-v0.2 cognitivecomputations/dolphin-mixtral-8x7b google/gemini-pro-vision google/gemini-pro mistralai/mixtral-8x7b mistralai/mixtral-8x7b-instruct openchat/openchat-7b:free openchat/openchat-7b neversleep/noromaid-20b anthropic/claude-2.1:beta anthropic/claude-2.1 anthropic/claude-2:beta anthropic/claude-2 teknium/openhermes-2.5-mistral-7b undi95/toppy-m-7b:free undi95/toppy-m-7b alpindale/goliath-120b openrouter/auto openai/gpt-3.5-turbo-1106 openai/gpt-4-1106-preview google/palm-2-chat-bison-32k google/palm-2-codechat-bison-32k jondurbin/airoboros-l2-70b xwin-lm/xwin-lm-70b openai/gpt-3.5-turbo-instruct mistralai/mistral-7b-instruct-v0.1 pygmalionai/mythalion-13b openai/gpt-3.5-turbo-16k openai/gpt-4-32k openai/gpt-4-32k-0314 nousresearch/nous-hermes-llama2-13b mancer/weaver huggingfaceh4/zephyr-7b-beta:free anthropic/claude-2.0:beta anthropic/claude-2.0 undi95/remm-slerp-l2-13b google/palm-2-chat-bison google/palm-2-codechat-bison gryphe/mythomax-l2-13b:free gryphe/mythomax-l2-13b meta-llama/llama-2-13b-chat meta-llama/llama-2-70b-chat openai/gpt-3.5-turbo openai/gpt-3.5-turbo-0125 openai/gpt-4 openai/gpt-4-0314 PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8>

**O ChatGPT disse:**

Ótima notícia! 🎉 O modelo **"mistralai/mistral-7b-instruct"** **está disponível** na sua conta. E melhor ainda, ele tem uma versão gratuita:

✅ **mistralai/mistral-7b-instruct:free**

Agora, você pode chamá-lo na API sem precisar pagar.

**Próximo passo: Fazer uma requisição para o modelo**

Agora, tente enviar uma mensagem para ele usando este comando no PowerShell:

powershell

CopiarEditar

$headers = @{

"Authorization" = "Bearer SUA\_CHAVE\_API"

"Content-Type" = "application/json"

}

$body = @{

"model" = "mistralai/mistral-7b-instruct:free"

"messages" = @(@{"role" = "system"; "content" = "Você é um assistente útil." }, @{"role" = "user"; "content" = "Qual é a capital da França?"})

"max\_tokens" = 100

} | ConvertTo-Json -Depth 10

Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions" -Headers $headers -Method Post -Body $body

Se tudo der certo, o modelo deve responder **"A capital da França é Paris."** 🇫🇷

Me avise se precisar de ajustes! 🚀

4o

**Você disse:**

PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $headers = @{ >> "Authorization" = "Bearer SUA\_CHAVE\_API" >> "Content-Type" = "application/json" >> } PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $body = @{ >> "model" = "mistralai/mistral-7b-instruct:free" >> "messages" = @(@{"role" = "system"; "content" = "Você é um assistente útil." }, @{"role" = "user"; "content" = "Qual é a capital da França?"}) >> "max\_tokens" = 100 >> } | ConvertTo-Json -Depth 10 PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions" -Headers $headers -Method Post -Body $body Invoke-RestMethod : {"error":{"message":"No auth credentials found","code":401}} No linha:1 caractere:1 + Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions ... + ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ + CategoryInfo : InvalidOperation: (System.Net.HttpWebRequest:HttpWebRequest) [Invoke-RestMethod], WebException + FullyQualifiedErrorId : WebCmdletWebResponseException,Microsoft.PowerShell.Commands.InvokeRestMethodCommand PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8

**O ChatGPT disse:**

O erro {"error":{"message":"No auth credentials found","code":401}} indica que a requisição não está autenticada corretamente. Aqui estão algumas coisas que você pode verificar e corrigir:

**1. Verifique a Chave da API**

* Certifique-se de que você substituiu "SUA\_CHAVE\_API" por uma chave válida da OpenRouter.
* Exemplo correto:

powershell

CopiarEditar

$headers = @{

"Authorization" = "Bearer sk-xxxxxx" # Use sua chave real aqui

"Content-Type" = "application/json"

}

**2. Remova Caracteres Ocultos ou Espaços Extras**

* Às vezes, copiar e colar pode adicionar caracteres invisíveis. Tente digitar a chave manualmente.

**3. Verifique o Formato do JSON**

* O PowerShell às vezes gera JSONs incompatíveis. Tente forçar a formatação correta:

powershell

CopiarEditar

$body = @"

{

"model": "mistralai/mistral-7b-instruct:free",

"messages": [

{ "role": "system", "content": "Você é um assistente útil." },

{ "role": "user", "content": "Qual é a capital da França?" }

],

"max\_tokens": 100

}

"@

Depois, envie a requisição:

powershell

CopiarEditar

Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions" -Headers $headers -Method Post -Body $body

**4. Verifique a URL**

* A URL parece correta, mas confirme no site da OpenRouter se há atualizações.

**5. Teste a Chave API com um Curl Simples**

Se ainda estiver com erro, teste com curl no PowerShell:

powershell

CopiarEditar

curl -X POST "https://openrouter.ai/api/v1/chat/completions" `

-H "Authorization: Bearer SUA\_CHAVE\_API" `

-H "Content-Type: application/json" `

-d '{ "model": "mistralai/mistral-7b-instruct:free", "messages": [{ "role": "system", "content": "Você é um assistente útil." }, { "role": "user", "content": "Qual é a capital da França?" }], "max\_tokens": 100 }'

Se funcionar no curl, o problema pode estar na conversão JSON do PowerShell.

Caso o problema continue, me avise com os detalhes! 🚀

4o

**Você disse:**

PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $headers = @{ >> "Authorization" = "Bearer sk-xxxxxx" # Use sua chave real aqui >> "Content-Type" = "application/json" >> } PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $headers = @{ >> "Authorization" = "Bearer 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 >> " # Use sua chave real aqui >> "Content-Type" = "application/json" >> } PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $body = @" >> { >> "model": "mistralai/mistral-7b-instruct:free", >> "messages": [ >> { "role": "system", "content": "Você é um assistente útil." }, >> { "role": "user", "content": "Qual é a capital da França?" } >> ], >> "max\_tokens": 100 >> } >> "@ PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions" -Headers $headers -Method Post -Body $body id : gen-1743620040-Fkt5kXvmqREcrqqQ4MRt provider : HuggingFace model : mistralai/mistral-7b-instruct object : chat.completion created : 1743620040 choices : {@{logprobs=; index=0; message=}} usage : @{prompt\_tokens=25; completion\_tokens=7; total\_tokens=32} PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8>

**O ChatGPT disse:**

Parece que agora a requisição foi bem-sucedida! 🚀 No entanto, a resposta parece incompleta, pois o campo choices[0].message está vazio. Aqui estão algumas sugestões para corrigir isso:

**🔍 1. Verifique a Estrutura da Resposta**

Tente exibir os detalhes do retorno para ver exatamente como a resposta está chegando:

powershell

CopiarEditar

$response = Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions" -Headers $headers -Method Post -Body $body

# Exibir a resposta completa

$response | ConvertTo-Json -Depth 10

Isso ajudará a entender se o problema é com a API ou apenas com a exibição dos dados.

**🔧 2. Exiba a Resposta Corretamente**

Se a resposta estiver chegando corretamente mas você não estiver conseguindo visualizá-la, tente exibir especificamente o conteúdo da mensagem:

powershell

CopiarEditar

$response.choices[0].message.content

Se message estiver vazio, o problema pode estar na API ou no formato do body.

**✏ 3. Ajuste o Corpo da Requisição**

Se o problema persistir, tente adicionar "stream": false no corpo da requisição, pois algumas APIs retornam dados em partes quando o streaming está ativado:

powershell

CopiarEditar

$body = @"

{

"model": "mistralai/mistral-7b-instruct:free",

"messages": [

{ "role": "system", "content": "Você é um assistente útil." },

{ "role": "user", "content": "Qual é a capital da França?" }

],

"max\_tokens": 100,

"stream": false

}

"@

Depois, rode o comando novamente.

Se ainda houver problemas, me avise o que aparece no $response para que possamos investigar mais a fundo! 🔎💡

4o

**Você disse:**

PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $response = Invoke-RestMethod -Uri "https://openrouter.ai/api/v1/chat/completions" -Headers $headers -Method Post -Body $body PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> # Exibir a resposta completa PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8> $response | ConvertTo-Json -Depth 10 { "id": "gen-1743620128-Ln3iYSOYUZ81yul0DMrd", "provider": "HuggingFace", "model": "mistralai/mistral-7b-instruct", "object": "chat.completion", "created": 1743620128, "choices": [ { "logprobs": null, "index": 0, "message": { "role": "assistant", "content": " A capital da FranÃ§a Paris.", "refusal": null } } ], "usage": { "prompt\_tokens": 25, "completion\_tokens": 7, "total\_tokens": 32 } } PS C:\Users\profi\Downloads\project-bolt-sb1-bc1pual8>

**O ChatGPT disse:**

Agora está funcionando! 🚀 O modelo respondeu corretamente que "A capital da França é Paris." 🎉

**⚠️ Pequeno Problema de Codificação**

A resposta veio com um pequeno erro de codificação: "content": " A capital da FranÃ§a Paris.". Isso sugere que a API está retornando a resposta em um formato que não está sendo interpretado corretamente pelo PowerShell.

**✅ Como Corrigir**

Tente forçar o PowerShell a interpretar a saída corretamente usando -Encoding utf8:

powershell

CopiarEditar

$response.choices[0].message.content | Out-File -FilePath output.txt -Encoding utf8

Get-Content output.txt

Isso pode garantir que caracteres acentuados sejam exibidos corretamente.

Se precisar de mais ajustes, me avise! 🔥💬