

O Futuro Brilhante do COBOL

Daqui para o longo prazo

18 passos 4 horas

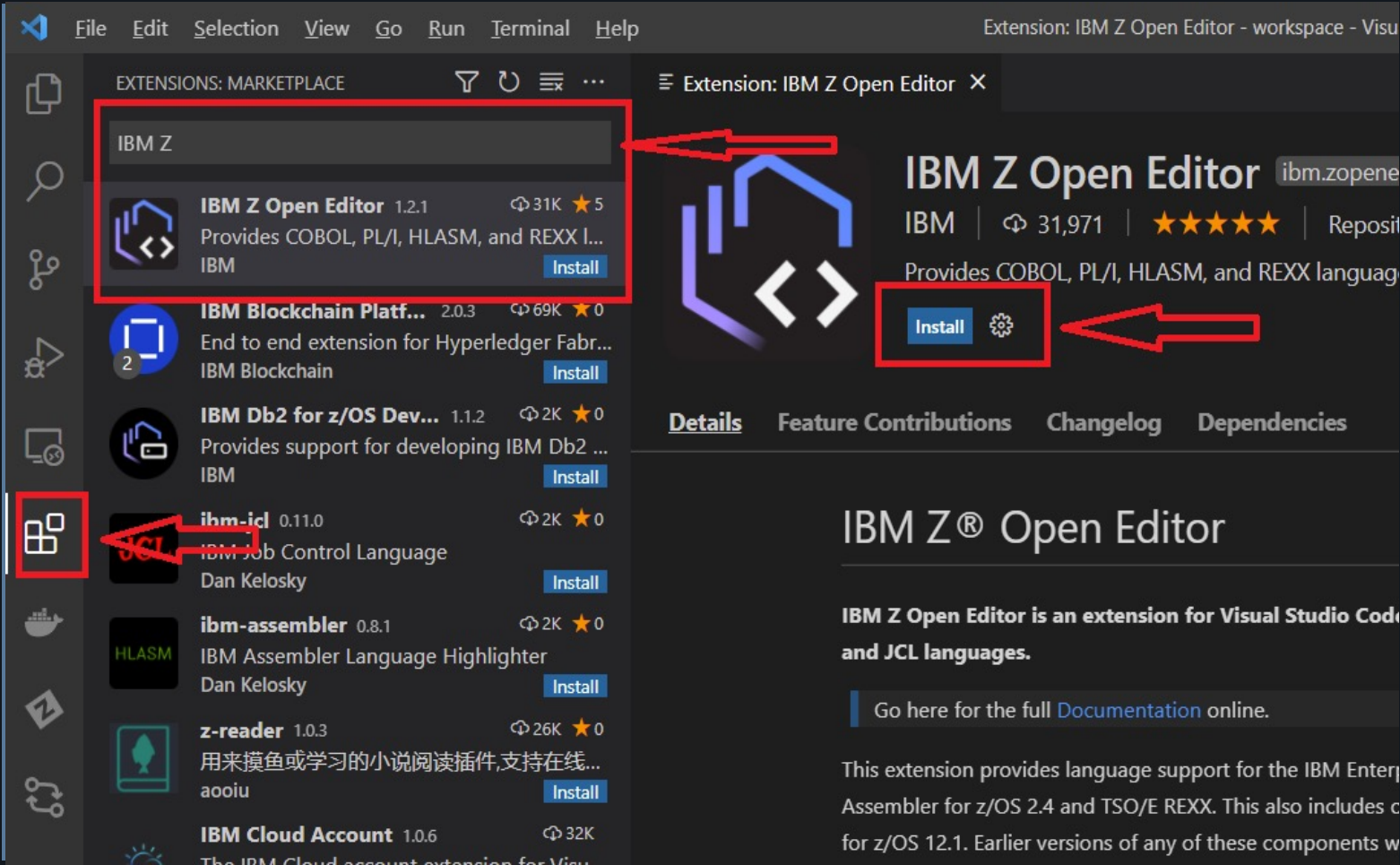
O DESAFIO

COBOL pode ser uma linguagem antiga, mas ainda roda o mundo!

Vamos abrir um programa COBOL que recebe um arquivo JSON como entrada e imprime um panfleto de supermercado. Você aprenderá o básico do COBOL no processo de conclusão dos desafios. Ser capaz de adicionar COBOL à sua própria lista pessoal de talentos definitivamente chamará sua atenção.

ANTES DE VOCÊ COMEÇAR

Nada além dos fundamentos (VS Code, conjuntos de dados, JCL) é necessário.



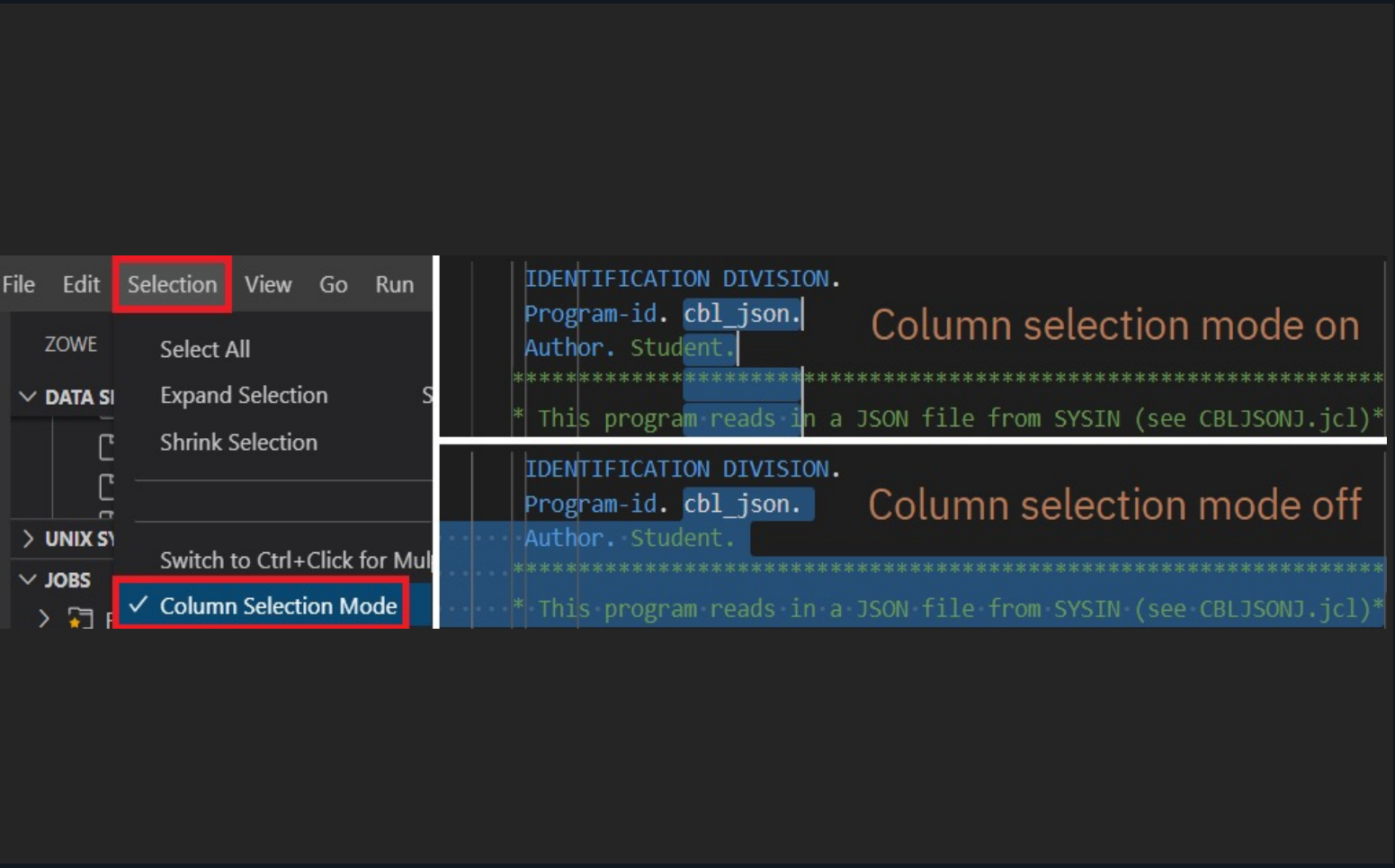
1. INSTALE O IBM Z OPEN EDITOR

Esta extensão VSCode fornece suporte à linguagem COBOL, incluindo realce de sintaxe

Se você ainda não tem o IBM Z Open Editor instalado, vamos fazer isso.

Abra o VSCode e, no menu de ferramentas à esquerda, selecione *Extensions* (Extensão).

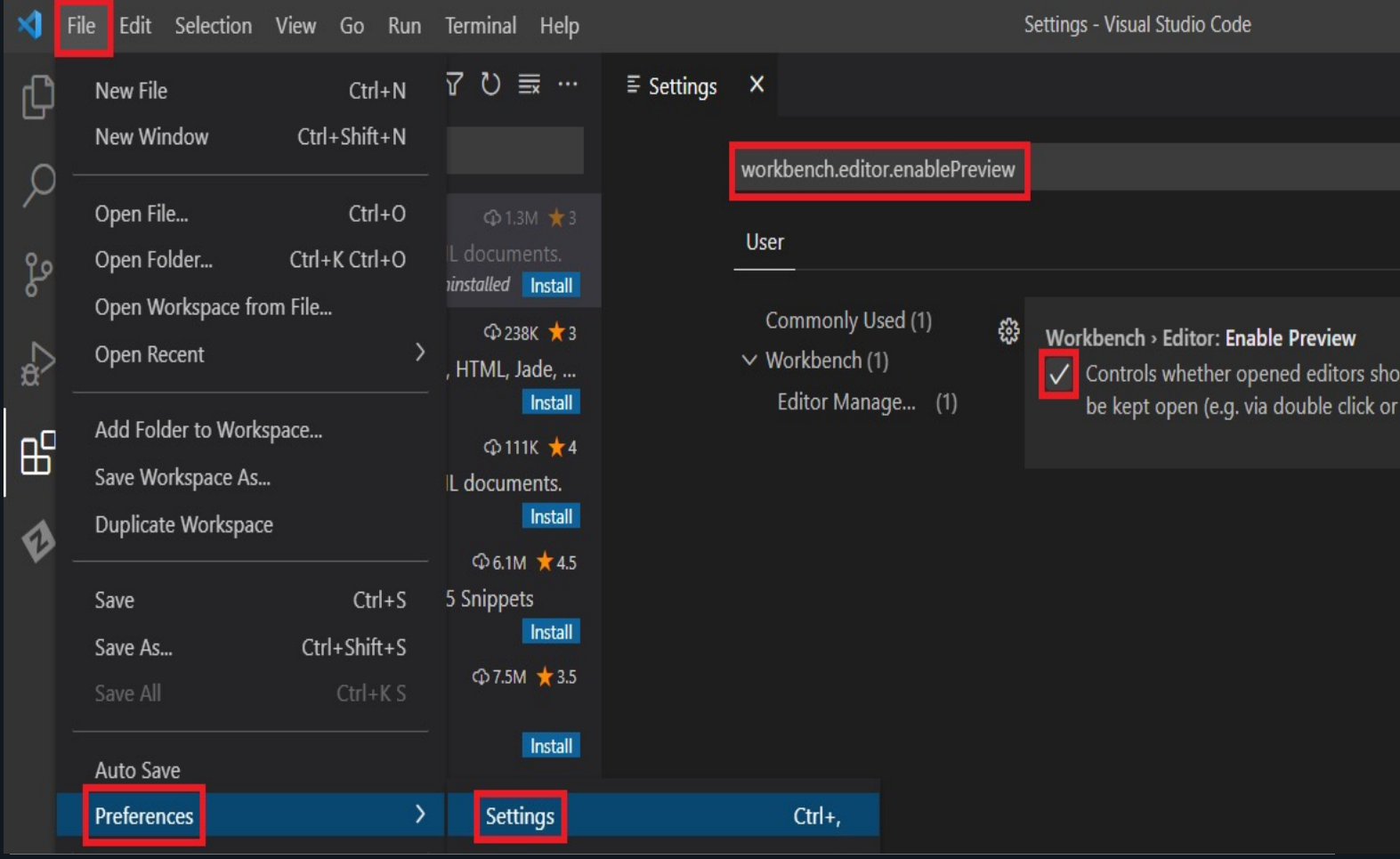
A partir daí, no campo de pesquisa “*Search Extensions in Marketplace*” (Pesquisar extensões no Marketplace), digite "IBM Z Open Editor". Os resultados da pesquisa começarão a ser preenchidos, selecione "IBM Z Open Editor" e clique em instalar.



2. MODO DE SELEÇÃO POR COLUNA

Por padrão, quando você seleciona linhas de código, elas serão destacadas em colunas, o que pode ser útil em algumas situações, mas provavelmente não neste desafio.

Se você quiser desativar esse recurso, na barra de menu superior, vá para *Selection->Column Selection Mode* (Seleção -> Modo de Seleção por Coluna) e desmarque.



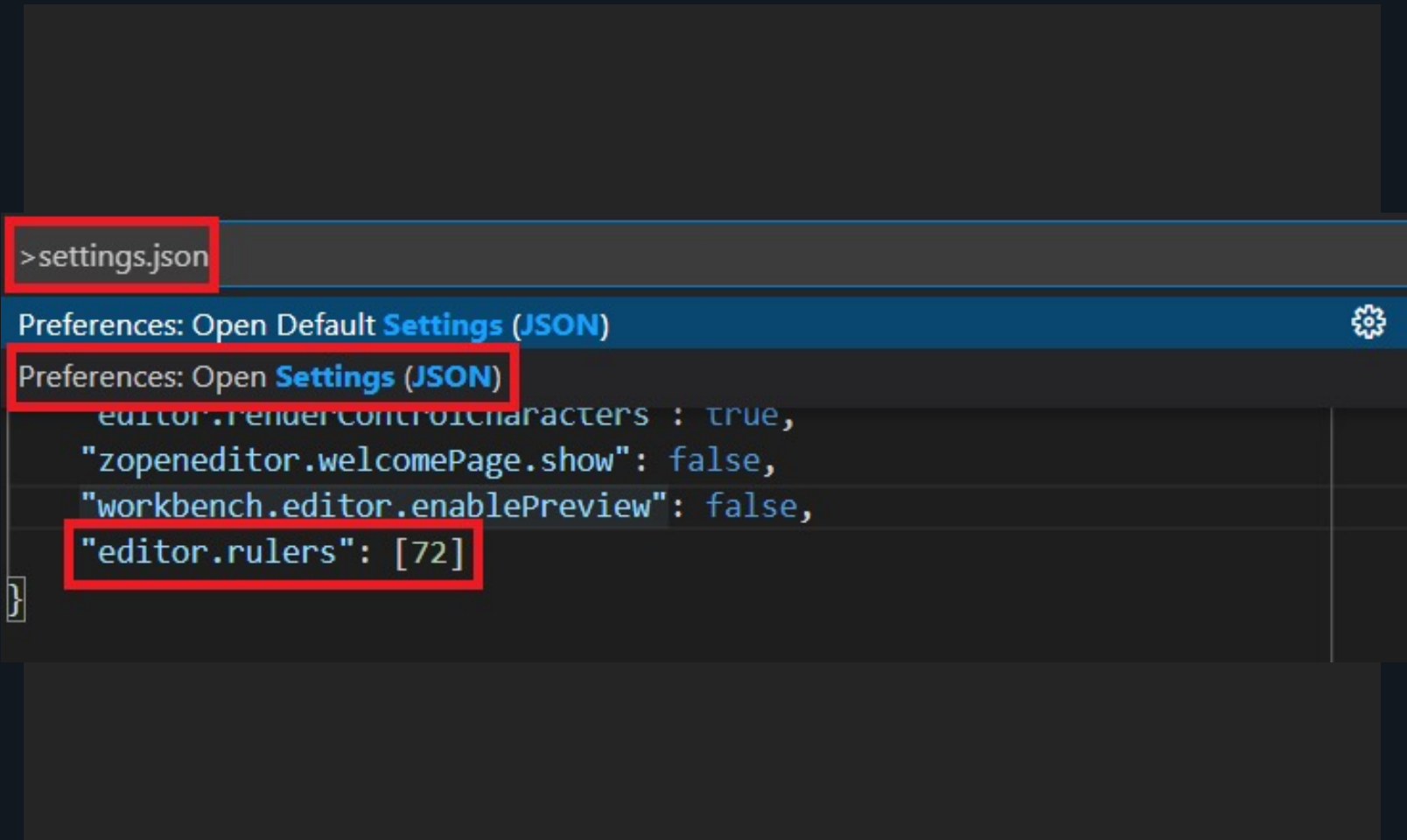
3. DEFINA PARA FICAR ABERTO

Crie sua própria biblioteca de fontes COBOL – do seu terminal VSCode, alocue um novo data set:

zowe files create pds --data-class spds 'Zxxxxxx.CBL'
Observação: para copiar e colar, certifique-se de que a formatação permaneça a mesma

Abra o data set de fontes 'ZXP.PUBLIC.SOURCE' e copie os membros JSONCBL e JSONJCL em seus próprios conjuntos de dados CBL e JCL, respectivamente. Não há necessidade de lê-los ainda.

Há um recurso padrão do VS Code que pode ser útil neste desafio para evitar que você precise fechar arquivos que você só precisa visualizar. Para tirar nossos arquivos principais do modo de visualização, clique duas vezes no nome da guia em *itálico*. Uma vez definido, o *itálico* é removido.

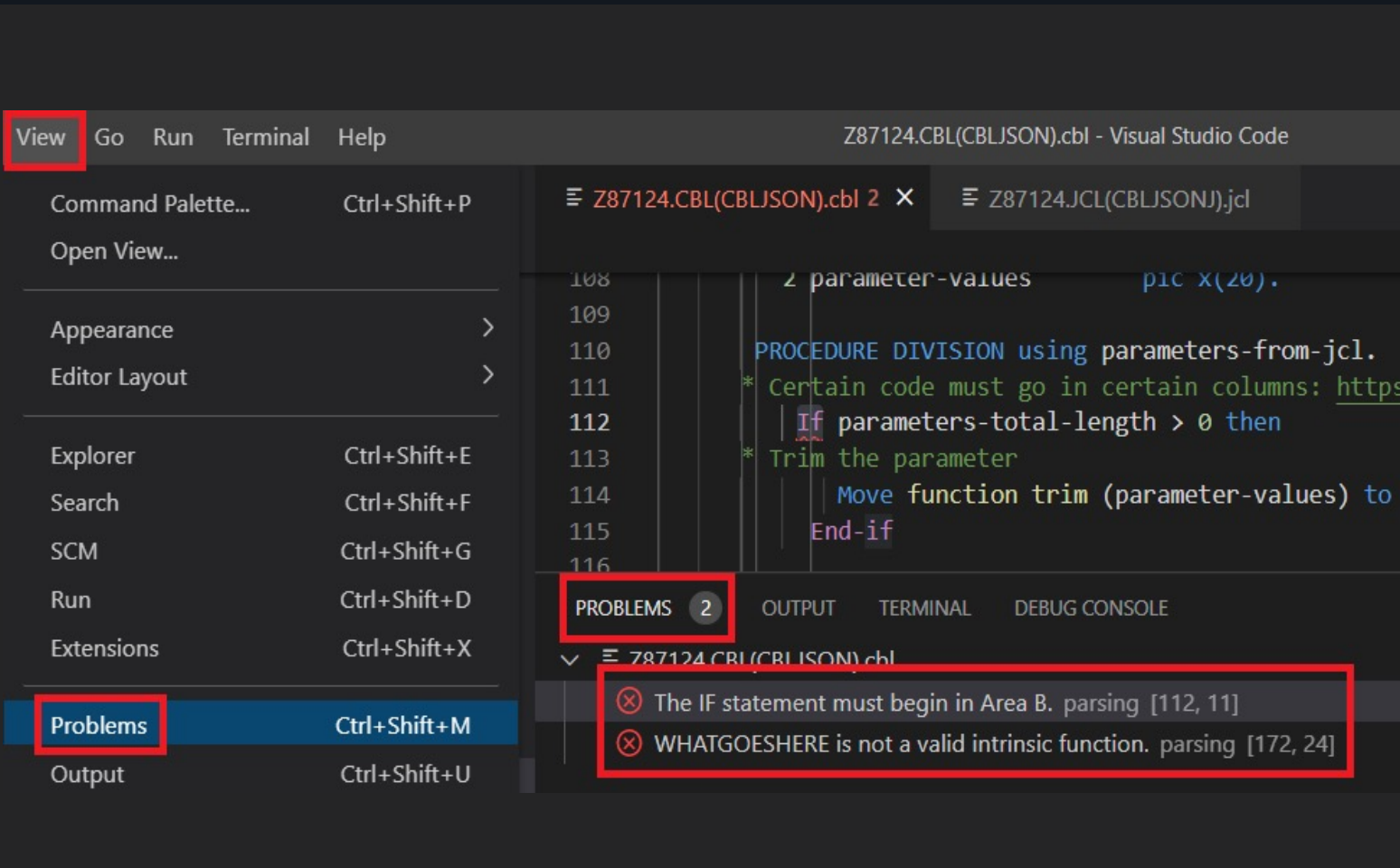


4. SETUP DAS COLUMN RULERS

O IBM Z Open Editor já configura as réguaas verticais necessárias para arquivos de origem COBOL.

As instruções JCL têm até 80 colunas de largura, mas as colunas 73-80 são ignoradas. Configure uma régua de coluna após a coluna 72 para não digitar além dela.

Abra o *Command Palette* – Paleta de Comandos (Ctrl+Shift+P no Windows) e digite “**settings json**” (Selecione “*Preferences: Open Settings (JSON)*”) → *Column rulers*, adicione isto no final: **"editor.rulers":[72]**
Certifique-se de adicionar uma vírgula no final da linha anterior.

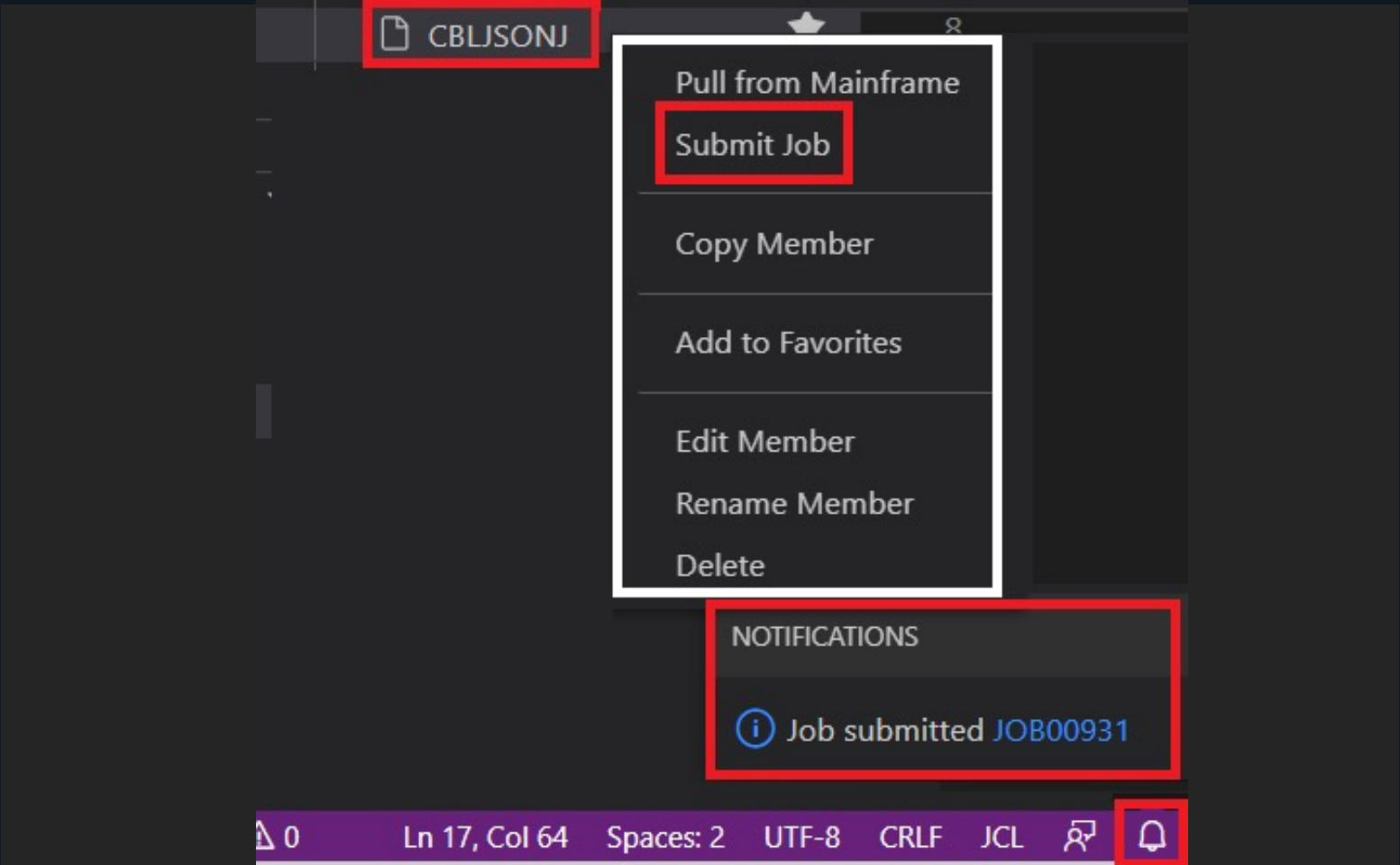


5. CHEQUE ERROS DE SINTAXE COBOL

Abra o arquivo COBOL e selecione ***View->Problems (Visualizar -> Problemas)*** no menu superior.

Uma lista de dois erros de sintaxe COBOL é exibida na guia Problemas na parte inferior. Clique duas vezes em cada problema e veja se você pode resolver os erros.

Depois de resolvidos, a guia Problemas deve mostrar 0 erros e não deve haver sublinhados vermelhos em seu programa COBOL.



6. SALVE, COMPILE E RODE

Salve usando ***File->Save (Arquivo -> Salvar)***.

Clique com o botão direito no membro de seu JCL, JSONJCL e ***Submit Job***.

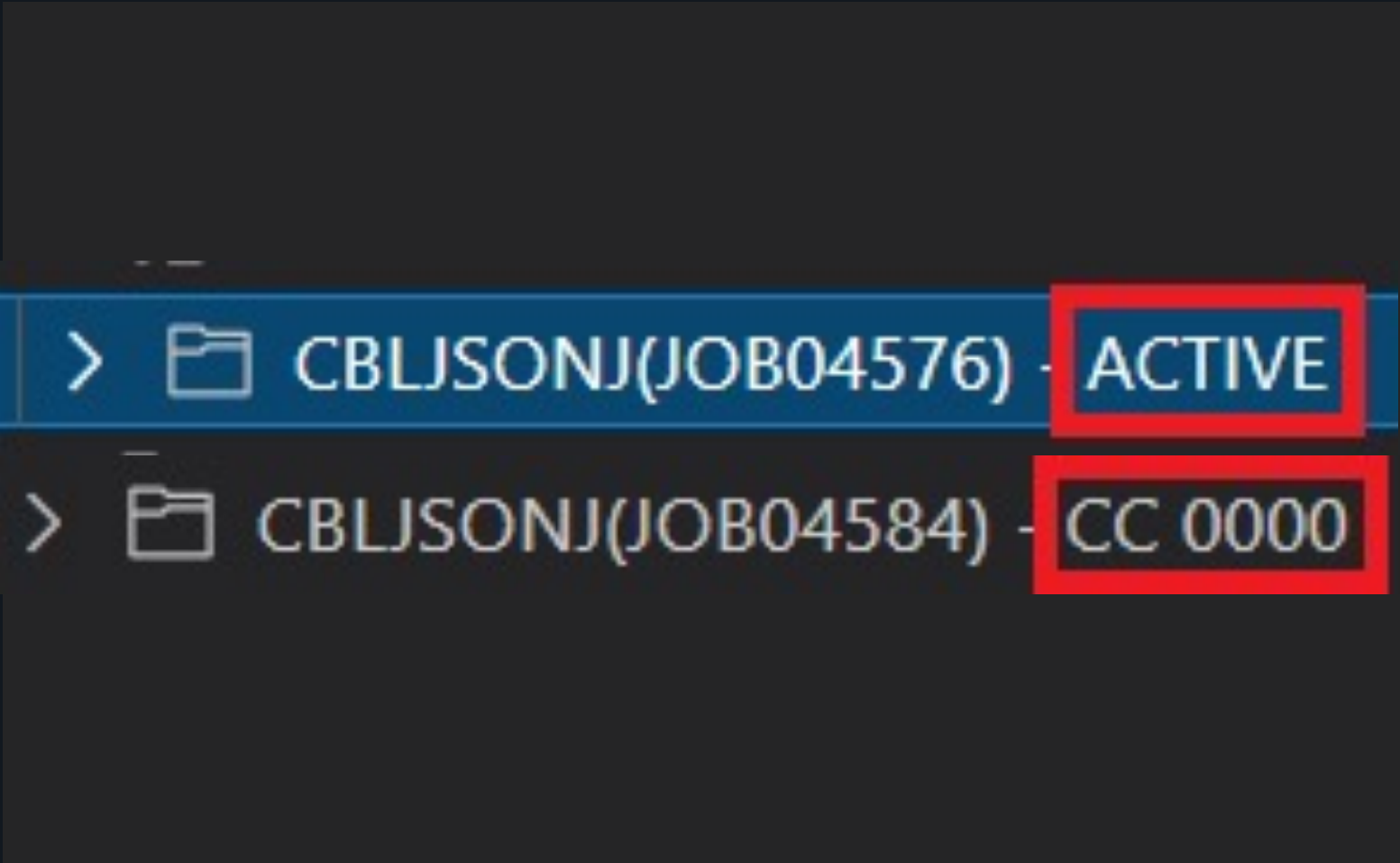
O script JCL compila e executa o programa COBOL. Uma notificação aparecerá informando que o trabalho foi enviado. Se desaparecer, clique no ícone de sino para ver as notificações recentes.

Clique no número do *job* para abrir os resultados do *job*.

POR QUE ESTAMOS APRENDENDO COBOL NESTA DÉCADA?
Resposta simples: COBOL está em toda parte..

Você ou alguém que você conhece já foi a um banco ou retirou dinheiro de um caixa eletrônico? Muito provavelmente, você está executando uma transação COBOL. Mais de um milhão de programas COBOL são executados a cada segundo de cada dia em mainframes. De fato, os clientes *Fortune 1000* da IBM trabalham com COBOL.

COBOL é auto-documentado, uma linguagem natural de processamento de strings e que faz aritmética de ponto fixo. Ele foi projetado para processamento de negócios, por isso é muito eficiente, adaptável e de fácil manutenção. Para saber mais sobre COBOL e sua relevância, consulte:
<https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/dan-zhang1/2020/04/02/should-we-still-use-cobol>

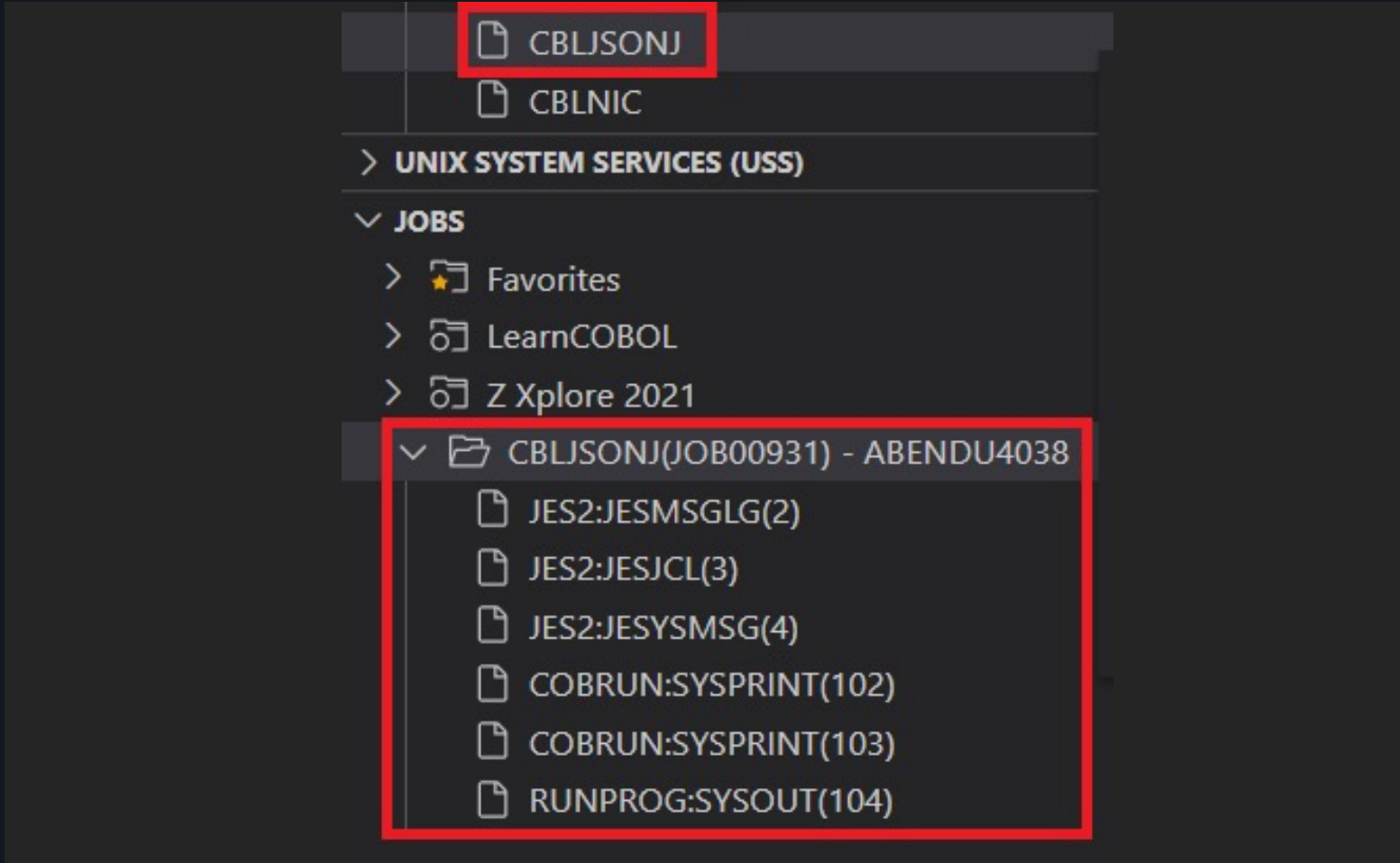


7. VERIFIQUE O CÓDIGO DE RETORNO

Se o programa mostrar “ACTIVE”, reduza e expanda novamente os resultados do job (ou pressione o ícone “atualizar” ao lado de JOBS) até ver que o programa terminou (o “ACTIVE” muda para um código de retorno).

Um código de retorno de CC 0000 significa sucesso e qualquer outra coisa significa que há um erro.

Se você corrigiu com êxito os problemas da etapa 5, ainda receberá uma mensagem de erro ABEND, que é o próximo desafio da etapa 8.

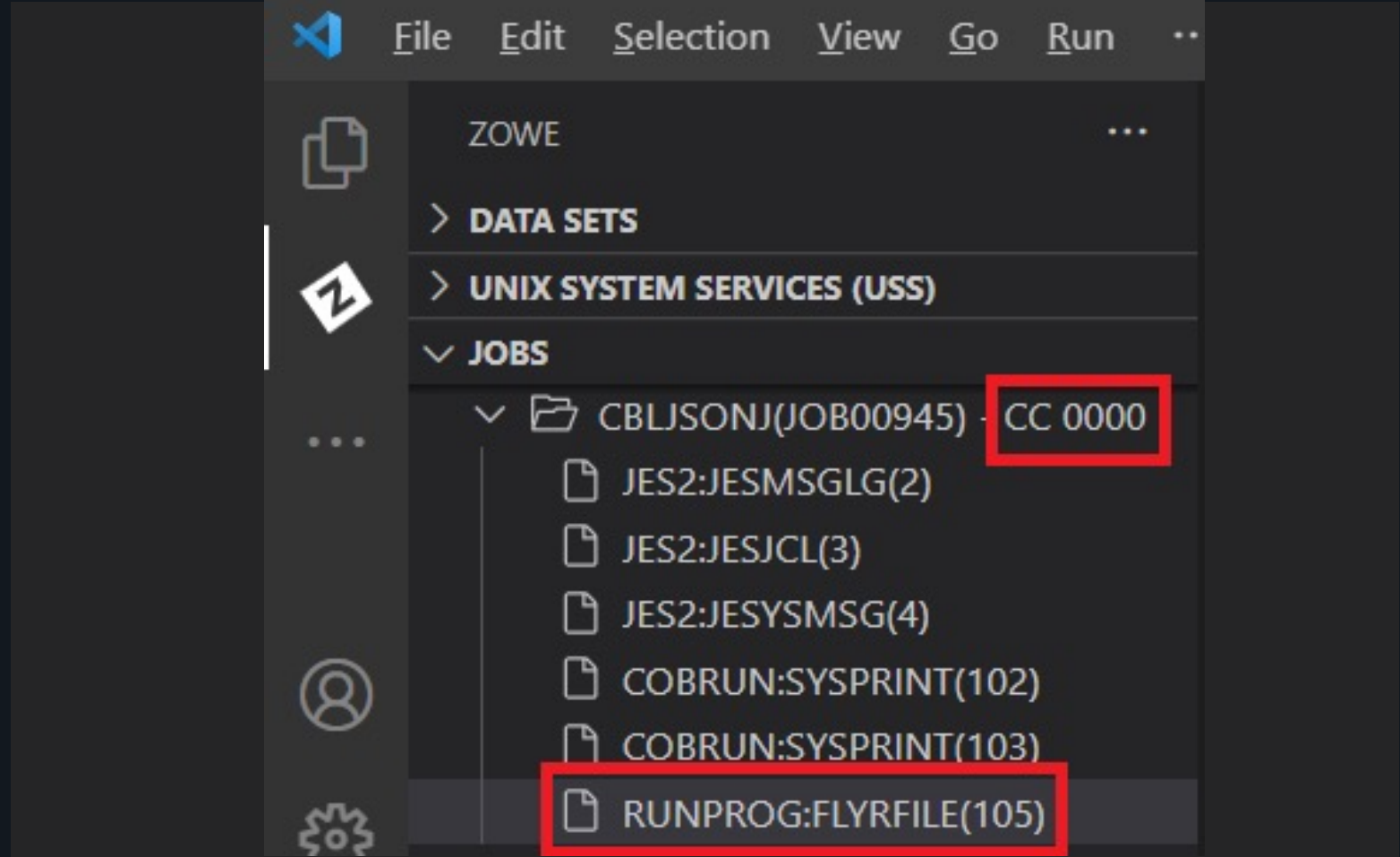


8. AH NÃO, UM ABEND

ABEND significa final anormal (abnormal end) e é um código de retorno de erro comum no z/OS.

Expanda os resultados para ver a lista de arquivos de output. Examine os arquivos de *output* para localizar aquele com a mensagem de erro.

A mensagem de erro refere-se a FLYYFILE. O que está errado? Examine ambos os arquivos, COBOL e JCL.



9. CORRIGIDO E PRONTO

Depois de corrigir o ABEND, envie o job JCL (etapa 6).

Você deverá ver o código de retorno CC 0000, que significa sucesso.

Você deve ter visto que este programa deve produzir um panfleto de supermercado. Você pode examinar o flyer no arquivo de saída RUNPROG:FLYRFILE(105).

Ele mostra apenas os produtos que estão prestes a expirar, com 50% de desconto. O folheto está em texto simples por enquanto, vamos corrigir isso daqui a pouco.

1	Canned Tuna	0.44	Was: 0.89
2	Green Beans	0.24	Was: 0.49
3	Peanut butter	1.99	Was: 3.99
4	Flyer in effect 2021-09-18 to 2021-09-25		

10. TÍTULO FALTANDO

O nome da nossa mercearia “Corner Grocery Store” está faltando no folheto de texto simples.

Esse aqui é complicadinho!

Identifique e corrija o erro e, em seguida, envie o job JCL novamente (da Etapa 8) para garantir que o título seja adicionado.

```
Linkage section.
1 parameters-from-jcl.
* System-inserted field for total string length of parameters
  2 parameters-total-length pic 9(4) usage comp.
* Flyer format parameter - TEXT or HTML
  2 parameter-values       pic x(20).

* Parameters are passed to the program from the JCL and moved
* into "flyerformat"
PROCEDURE DIVISION using parameters-from-jcl.
* Certain code must go in certain columns: https://ibm.co/3mwIUKw
  If parameters-total-length > 0 then
* Trim the parameter
    Move function trim (parameter-values) to flyerformat
  End-if
```

11. SÓ TEXTO PLANO NÃO TEM GRAÇA

Corrigimos o título, mas não podemos mostrar aos nossos clientes o flyer em texto simples como este!

Poderia haver um parâmetro para passar ao programa para adicionar formatação?

Quando você achar que descobriu, envie o trabalho JCL (novamente na Etapa 8) e verifique o folheto mais uma vez.

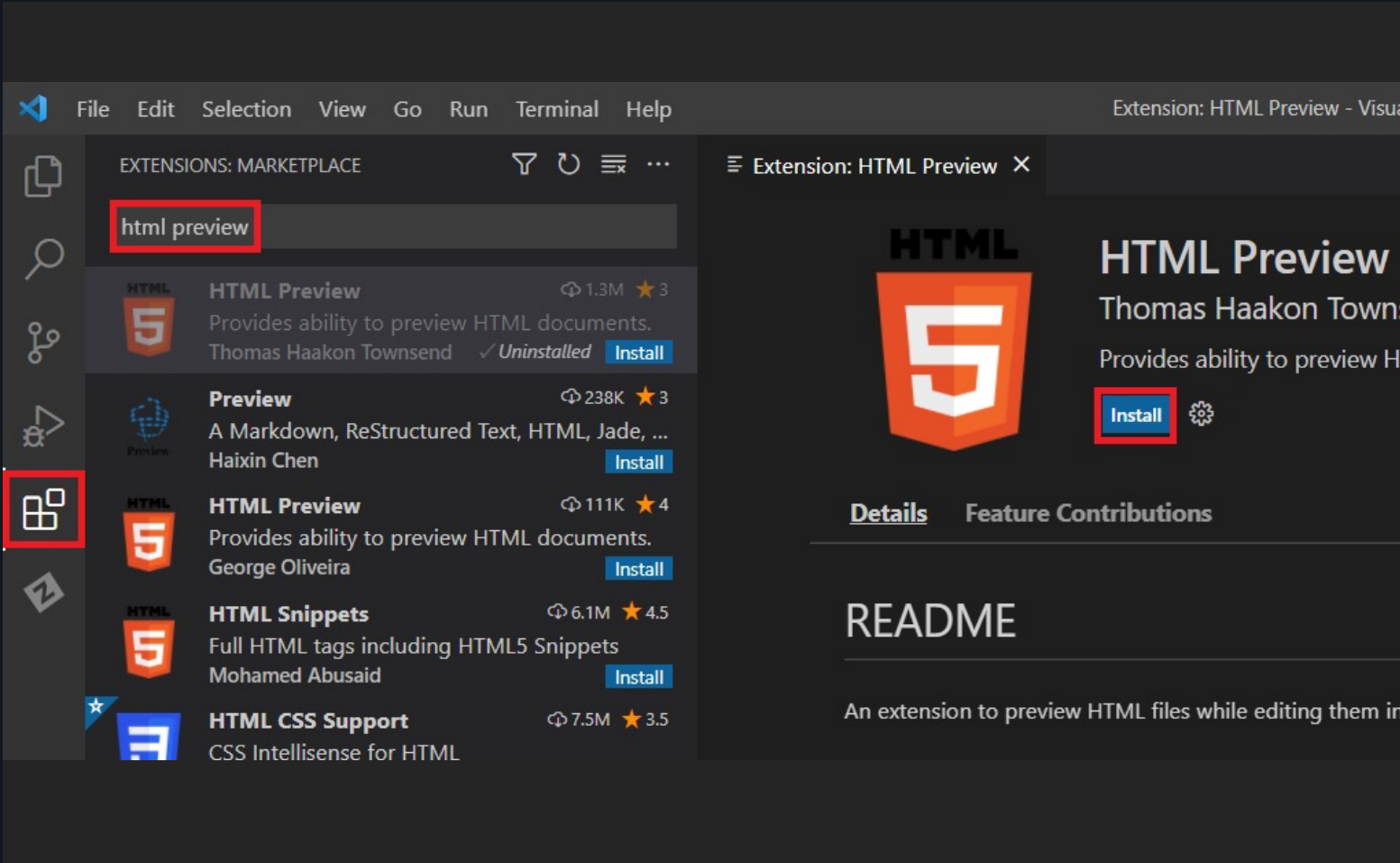
```
<html><head><style>body{font-family:IBM P.
</div><table><tr><td colspan=2><table><tr><td colspan=2><p>
Flyer in effect 2021-09-18 to 2021-09-25
</p></div></body></html>
```

12. RODE O CÓDIGO

Abra o dataset do job - RUNPROG:FLYRFILE(105).

Se você vir tags HTML começando com <html>, você conseguiu! Mas como podemos visualizá-lo para ver a formatação?

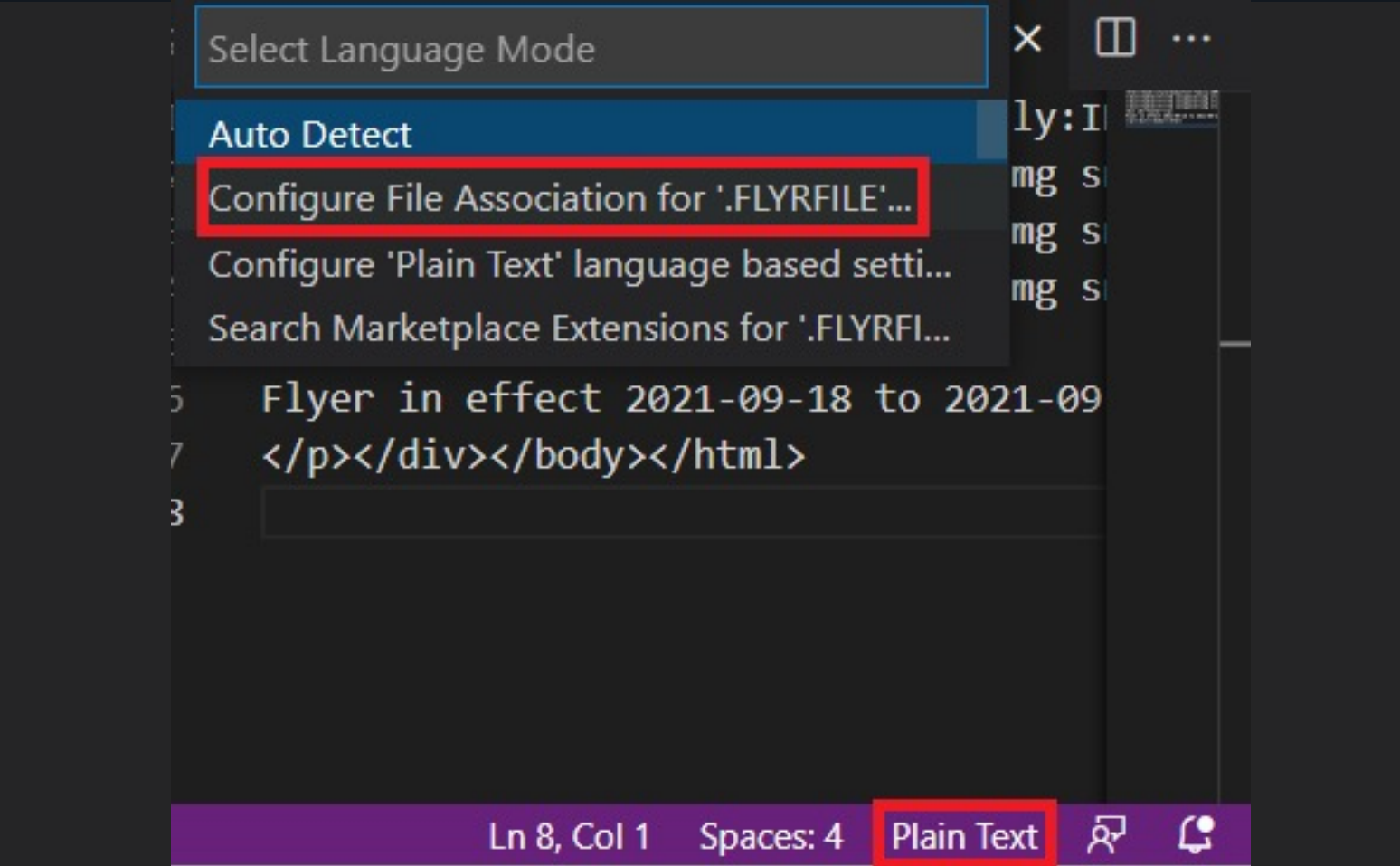




13. INSTALE A EXT. HTML PREVIEW

Selecione **Extensions (Extensões)** no menu de ferramentas do lado esquerdo do VSCode.

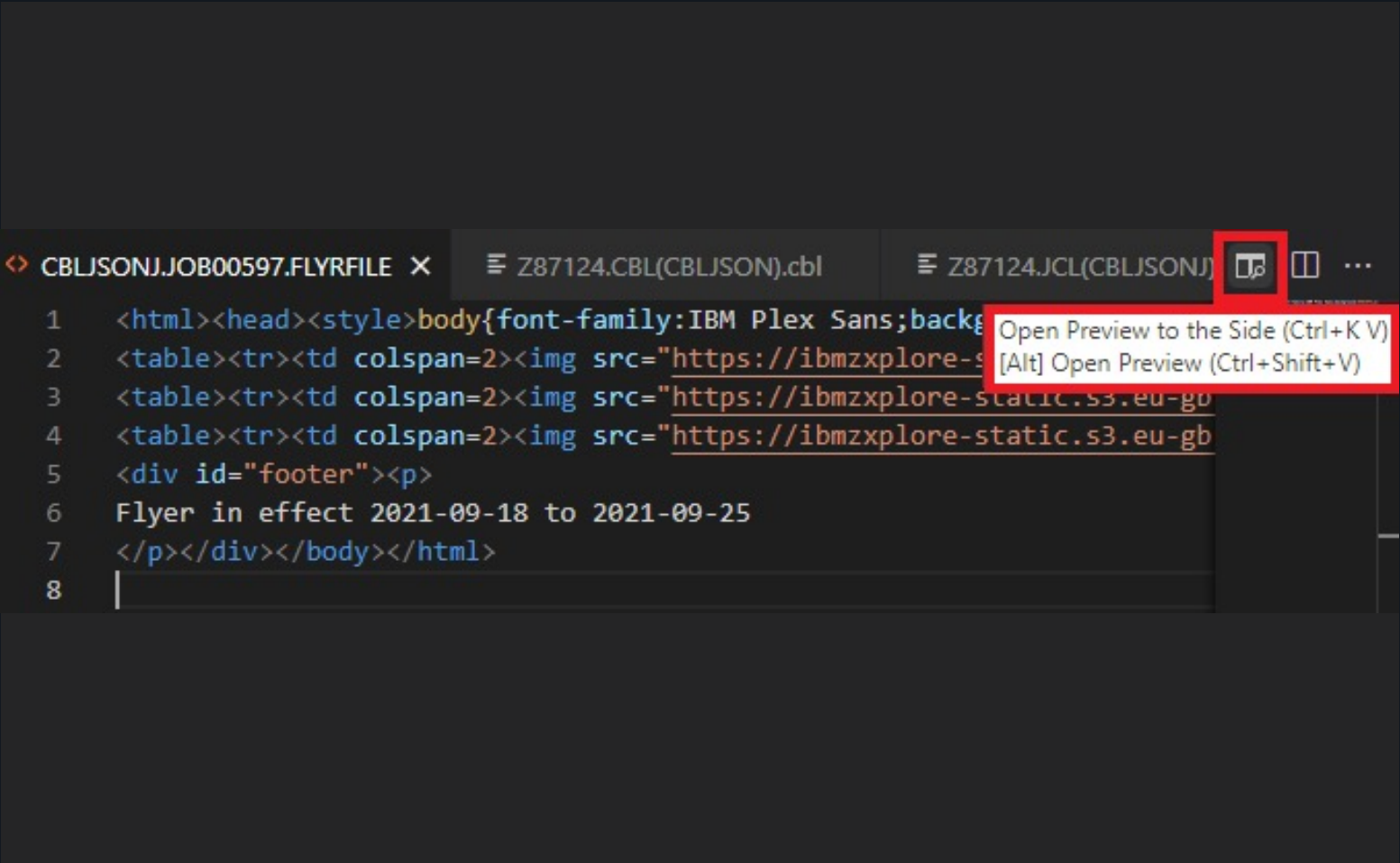
Pesquise HTML Preview e instale uma das opções, como a de Thomas Haakon Townsend.



14. DEFINA UM MODO DE LINGUAGEM

Com seu FLYRFILE aberto, no canto inferior direito, clique em **“Plain Text” (Texto plano)** e depois em **“Configure File Association for ‘.FLYRFILE’...” (Configurar Associação para ‘.FLYRFILE’...)** e digite **HTML**

Quando estiver feito, você verá o realce da sintaxe HTML.



15. ABRE-TE SÉSAMO

No canto superior direito, clique no ícone **“Open Preview to the Side”** (Abrir visualização ao lado).

O flyer está melhor formatado agora, mas ainda há alguns problemas.

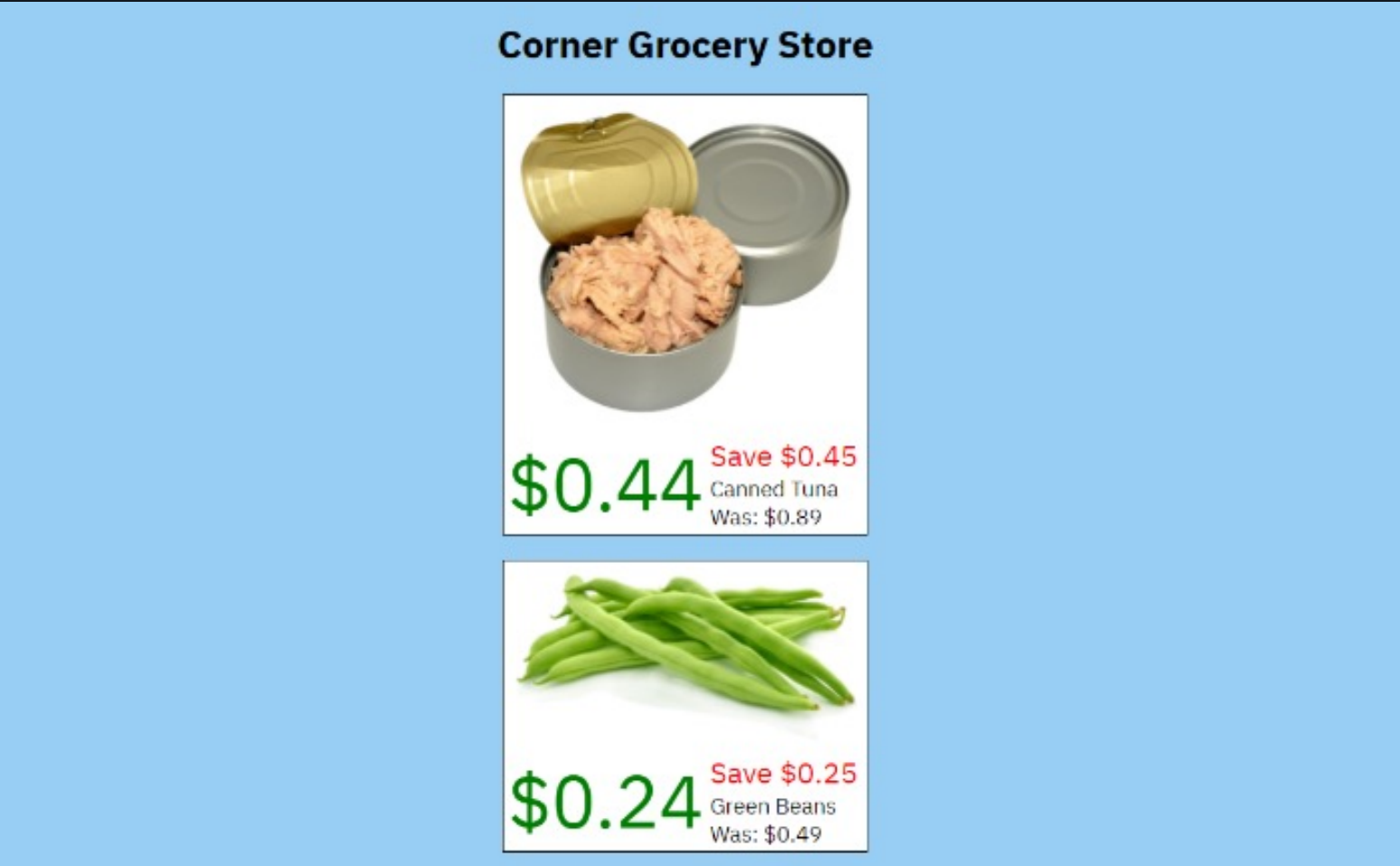


16. ADICIONE SÍMBOLOS MONETÁRIOS

A moeda deveria estar em dólares americanos (\$), mas o cifrão está faltando, o que há de errado?

Assim que o problema for corrigido, envie o job JCL (Etapa 8 novamente), abra a visualização HTML do flyer novamente para verificar seu trabalho.

Nosso JSON de entrada (no arquivo JCL) não continha links para imagens reais de produtos? Por que não estão aparecendo? Esquecemos de mapear essas imagens ao declarar a estrutura JSON em COBOL?



17. ADICIONE AS IMAGENS DOS PRODUTOS

Este é o desafio final. Depois de descobrir, envie o job JCL e abra a visualização HTML do folheto para garantir que ele fique bacana.

Para finalizar, crie uma cópia do html produzido em um arquivo **cobol.html** em seu diretório inicial do USS e envie o trabalho CHKACBLH de 'ZXP.PUBLIC.JCL'

Learning COBOL Programming



Welcome to your introduction to COBOL!

18. EXPANDA SEU CONHECIMENTO

Quer aprender mais? Faça o curso COBOL gratuito da IBM.

Inclui laboratórios práticos no VS Code:

- Versão em vídeo: <https://ibm.biz/learn-cobol-vscode>
- Versão em texto: <https://ibm.biz/github-cobol>

BOM TRABALHO! VAMOS RECAPITULAR

COBOL é uma das linguagens mais usadas no IBM Z atualmente e comanda a economia mundial, mas aqueles que conhecem a linguagem estão se aposentando. Sua ajuda é necessária para preencher essa lacuna. Você demonstrou uma sólida compreensão de alguns dos aspectos mais importantes do COBOL e tem alguns bons exemplos para compartilhar com possíveis empregadores. Você os impressionará especialmente demonstrando que conhece o suporte mais recente do COBOL para JSON, que eles podem nem conhecer ainda!

A SEGUIR...

Belo trabalho!

Você conseguiu expandir seu conhecimento em COBOL usando HTML. Continue sua experiência de aprendizado escolhendo outro desafio Avançado.

