

Db2

Bancos de dados SQL e relacionais

 15 etapas  60 minutos

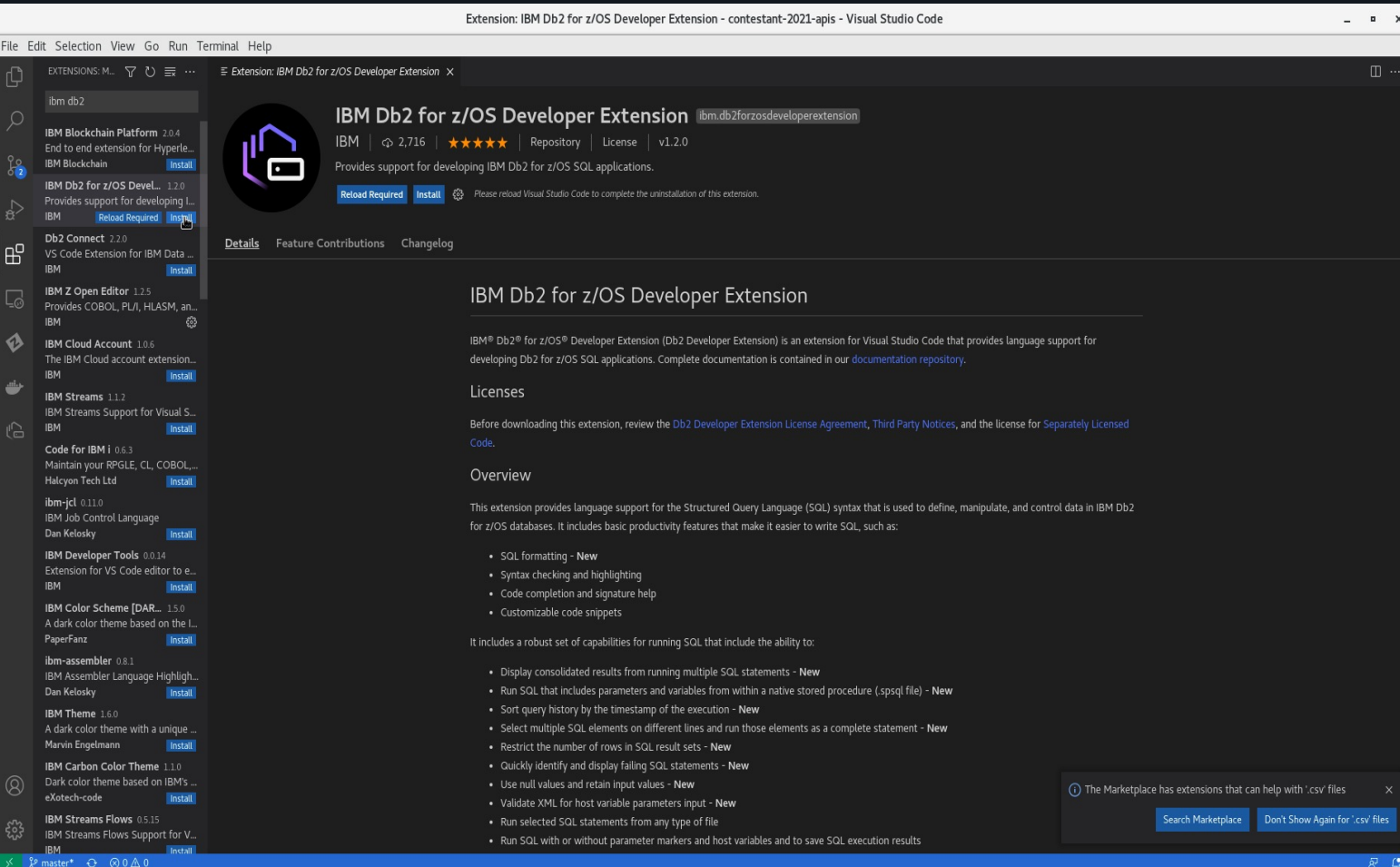
O DESAFIO

Os bancos de dados relacionais são uma parte importante de praticamente todos os softwares atuais. Quando você organiza dados em tabelas e, em seguida, vincula essas tabelas com base em dados comuns, é possível recuperar e organizar esses dados para entender melhor as relações entre eles.

Neste desafio, usaremos SQL para a comunicação com bancos de dados relacionais IBM DB2.

ANTES DE COMEÇAR

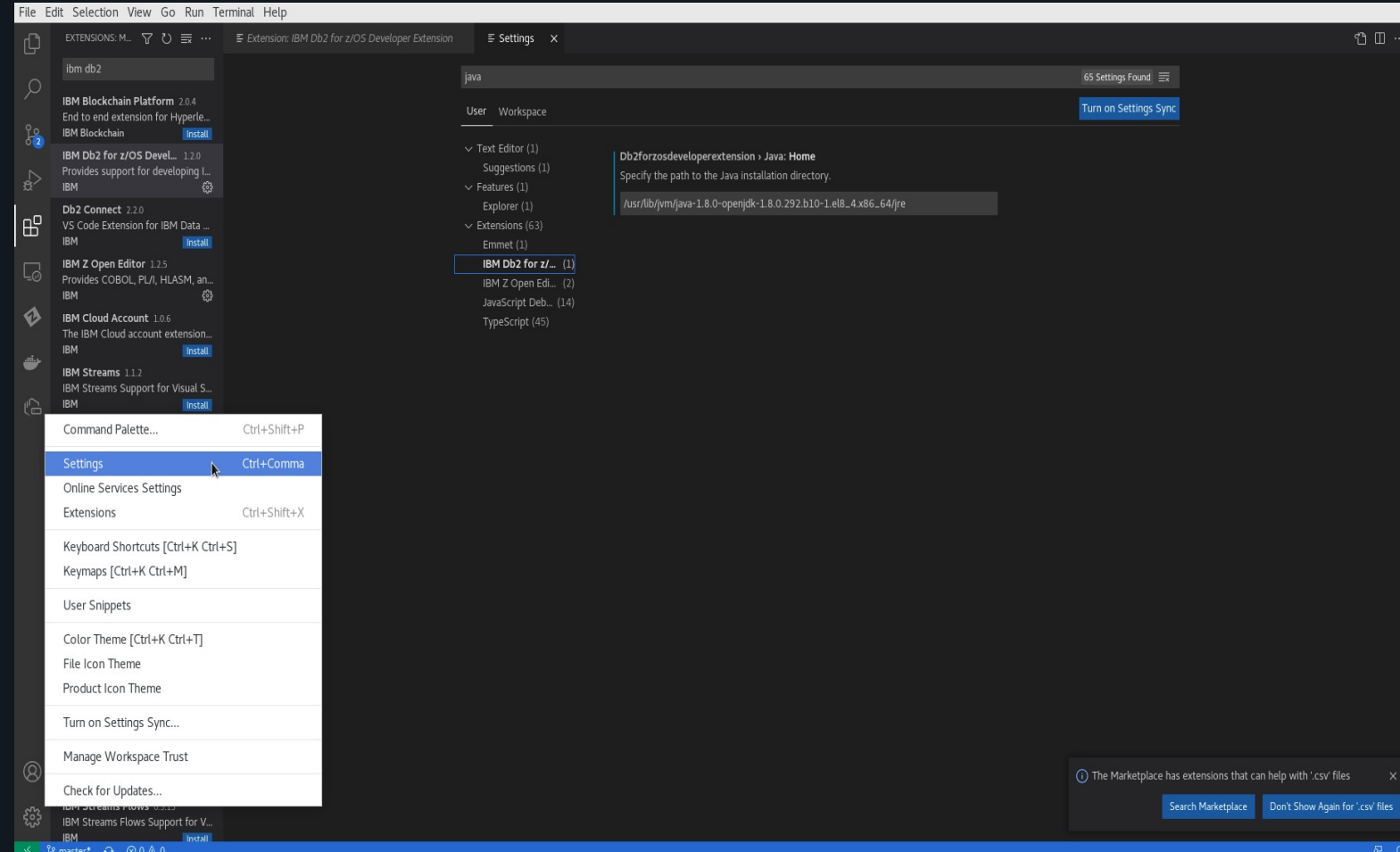
Por isso, se tudo estiver corretamente configurado no VS Code, você estará pronto para começar.



1. INSTALE A EXTENSÃO DO DB2

Clique no ativador de Extensões na barra de atividades e procure por IBM DB2. Em seguida, clique no botão "Instalar" para "IBM DB2 for z/OS Developer Extension". Essa é uma extensão do VS Code que permite trabalhar com bancos de dados DB2.

A instalação deve demorar apenas alguns segundos na maioria das conexões de Internet de banda larga.

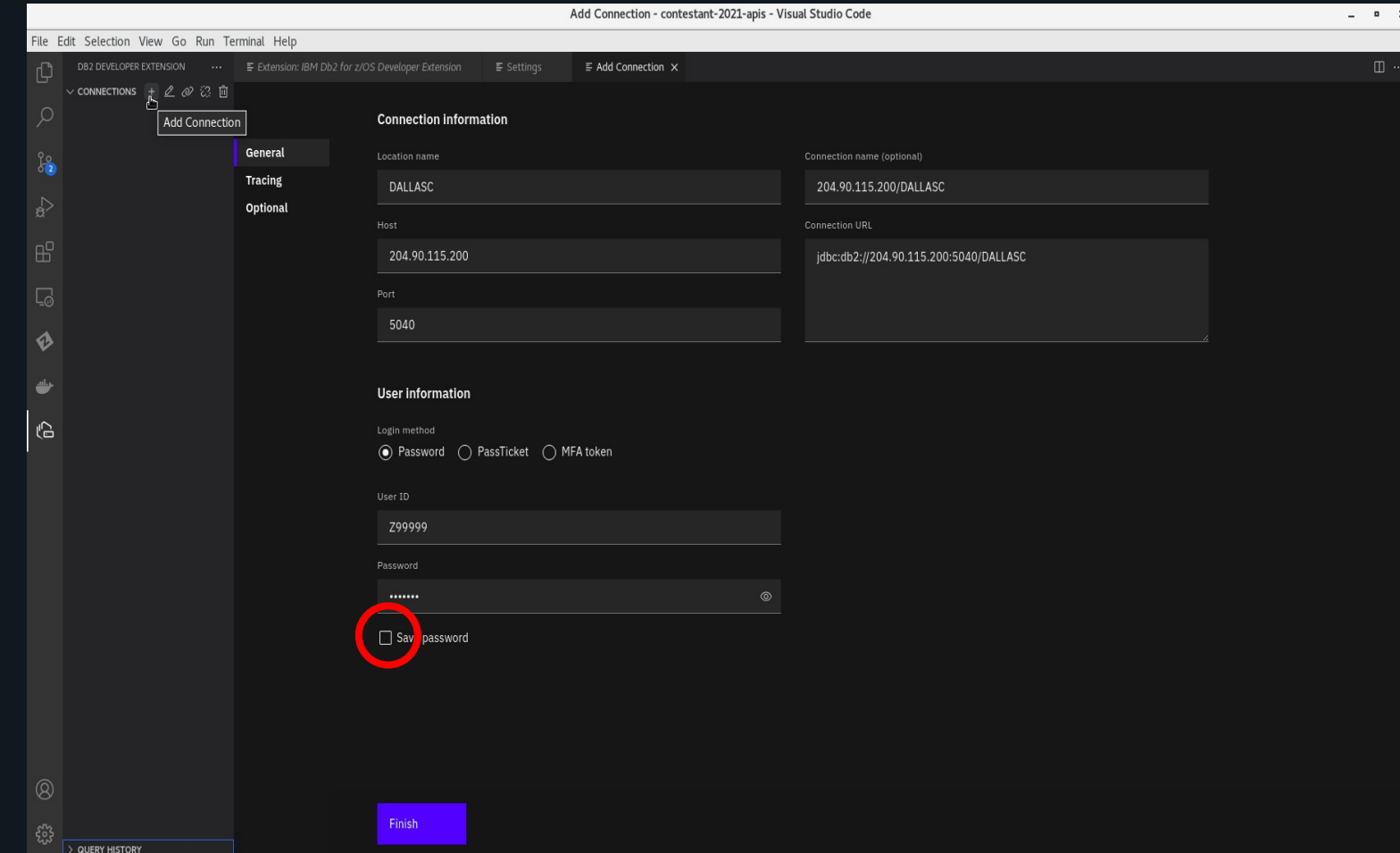


2. INSTALE O JAVA

É necessário ter um Java JDK instalado. Não apenas um JRE, mas um JDK completo. Há algumas opções, como usar o [Oracle Java JDK 8](#) ou a versão 8 ou 11 do [OpenJDK](#). Após a instalação, acesse as configurações do VS Code e procure Java na barra de procura. Deve haver uma entrada para DB2 em Extensões, na qual é possível clicar e fornecer o caminho para bibliotecas de tempo de execução Java.

No Mac, por exemplo, o caminho normalmente é /Library/Java/JavaVirtualMachine/jdk1.8.0_181.jdk/Contents/Home

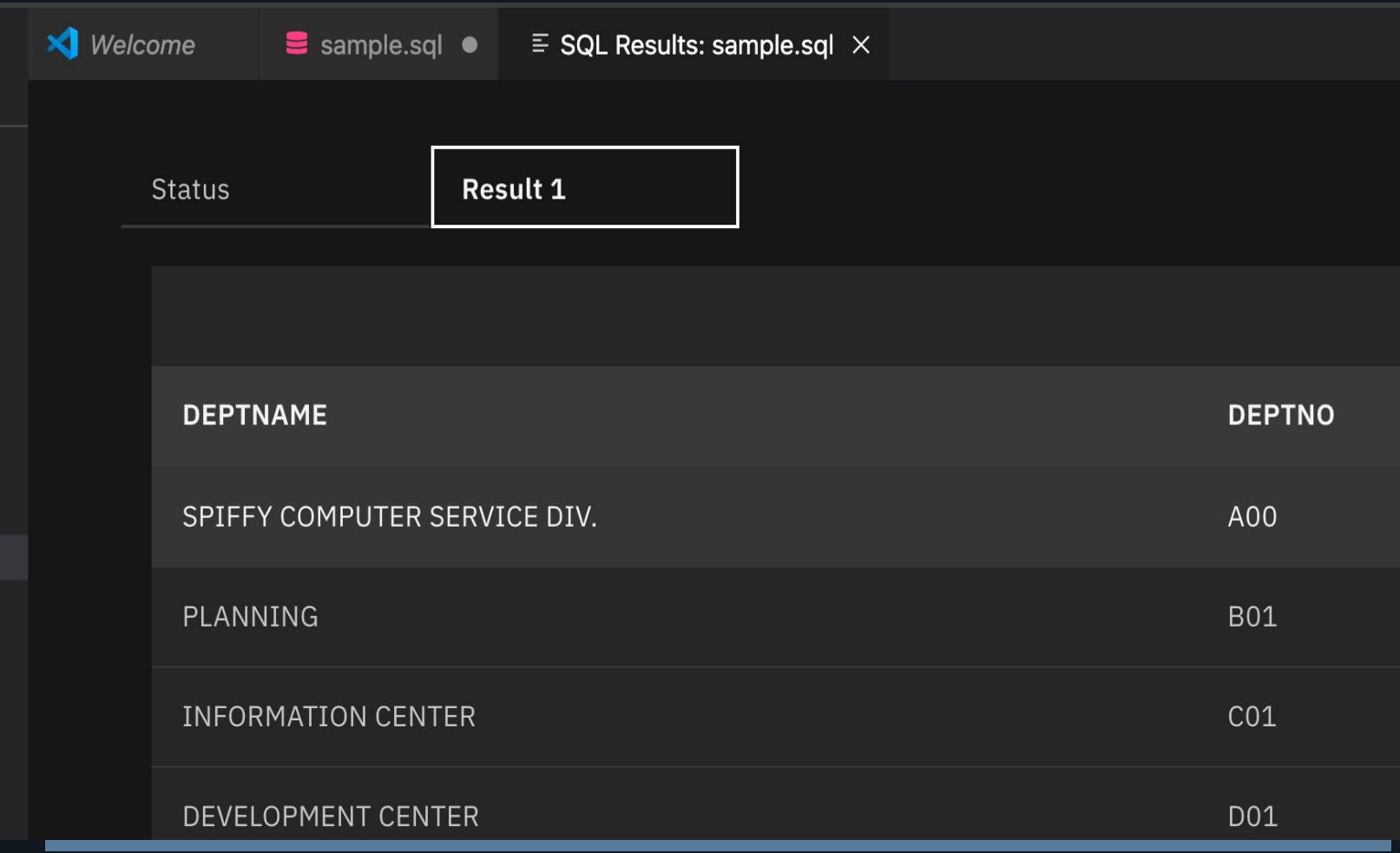
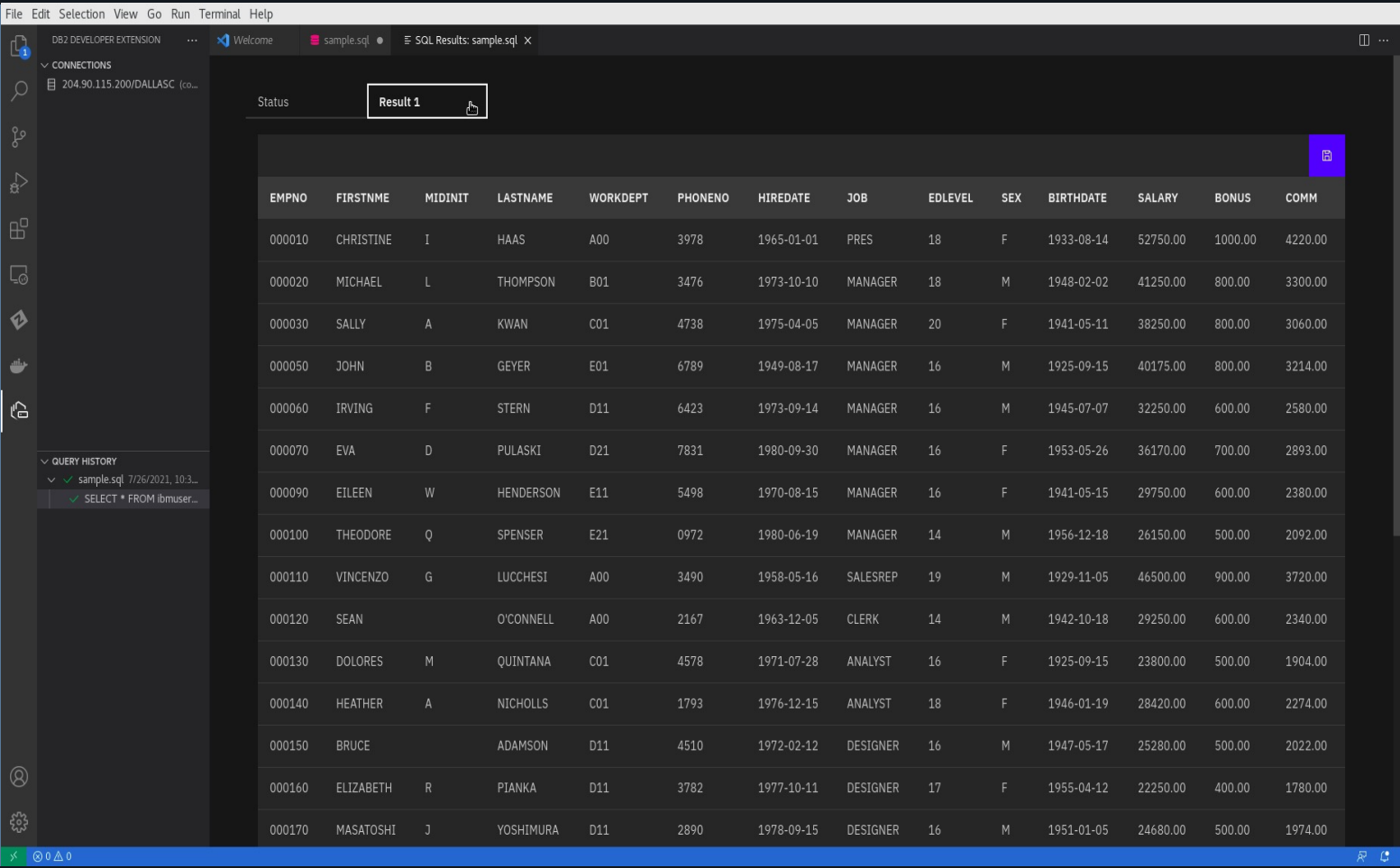
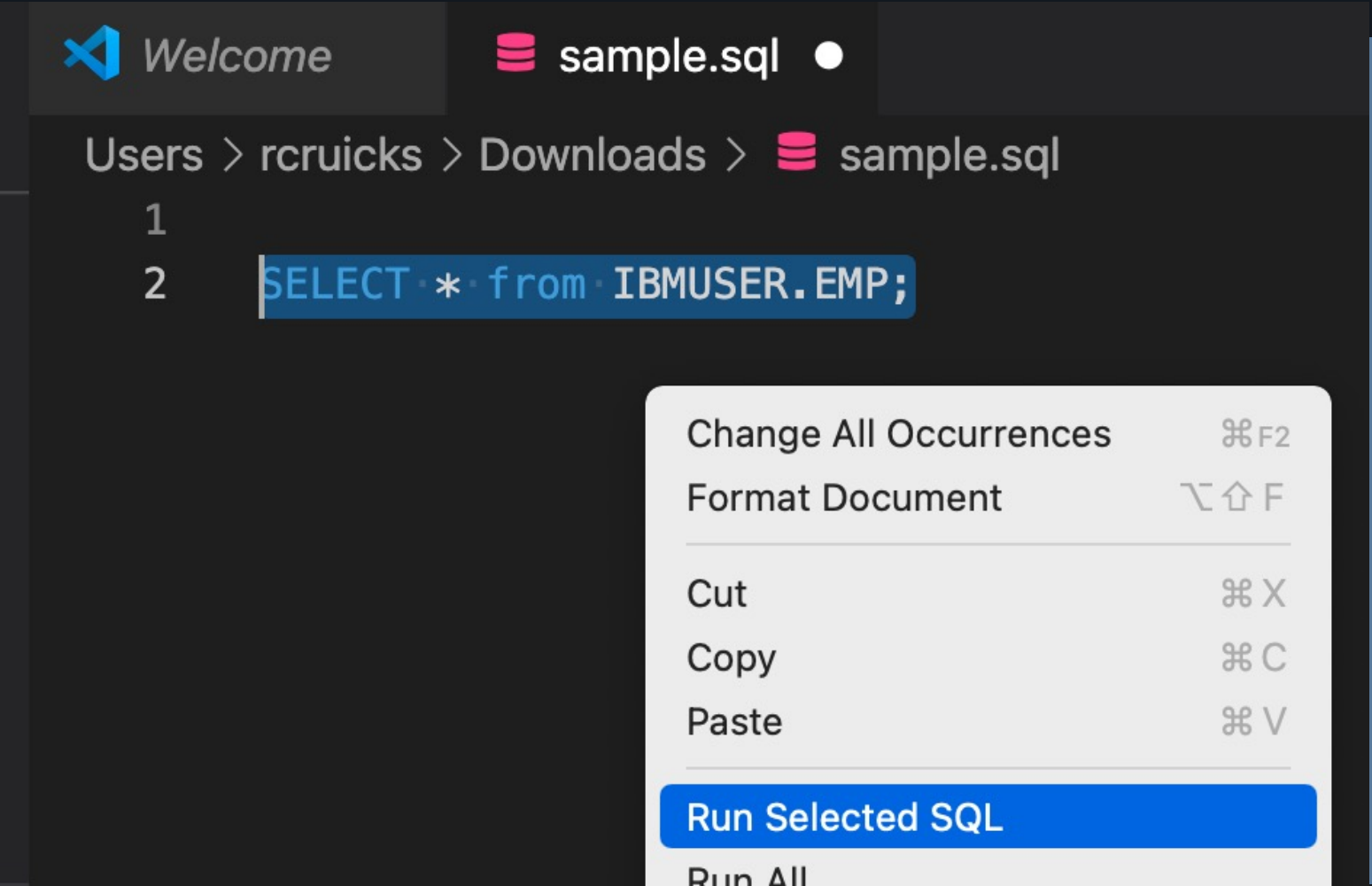
No Windows, provavelmente, é C:\Program Files\Java\jdk1.8.0_181



3. ALGUMAS INFORMAÇÕES

Acesse a Extensão do DB2 e clique no sinal de mais para incluir um novo perfil de conexão. Insira DALLASC como o nome do local, 204.90.115.200 como o host, 5040 como a porta e, em seguida, seu login do z/OS para o ID de usuário e a senha. A menos que você prefira inserir a senha sempre que fizer consultas SQL, marque a caixa “Salvar senha”.

Em seguida, clique em Concluir na parte inferior e você estará pronto para usar o DB2.



4. CRIANDO UMA INSTRUÇÃO

Crie uma pasta em sua máquina chamada *DB2 Challenges*, abra essa pasta no VSCode e crie um arquivo chamado *sample.sql*, conforme mostrado acima.

Aqui, você criará uma instrução SQL simples. Digite a instrução na captura de tela acima, `SELECT * from IBMUSER.EMP`, destaque a instrução e clique com o botão direito para ver as opções disponíveis. Escolha "Executar SQL selecionado"

5. OBTENHA ALGUNS RESULTADOS

A guia "Resultados de SQL" aparecerá depois que a instrução for executada e será possível visualizar a tabela criada como resultado de sua instrução SQL. A seção Status exibirá a instrução executada e o código de retorno 0 indicando uma execução bem-sucedida. A guia Resultado 1 contém a tabela resultante.

Como é possível notar, há muitas colunas na tabela. Ao colocar o asterisco após `SELECT` na instrução SQL, o banco de dados recupera todas as colunas da tabela *ibmuser.emp* para exibi-las nos resultados.

6. APROFUNDE-SE AINDA MAIS

Observe que, se você não precisar de todas as colunas, será possível especificar quais delas você deseja.

Por exemplo, é possível exibir apenas a tarefa dos funcionários digitando `"select JOB from ibmuser.emp"`. Também é possível exibir diversas colunas. Digite `"SELECT LASTNAME, EMPNO, PHONENO from ibmuser.emp"` e veja o resultado. Observe que a exibição ocorre na ordem exata que foi especificada.

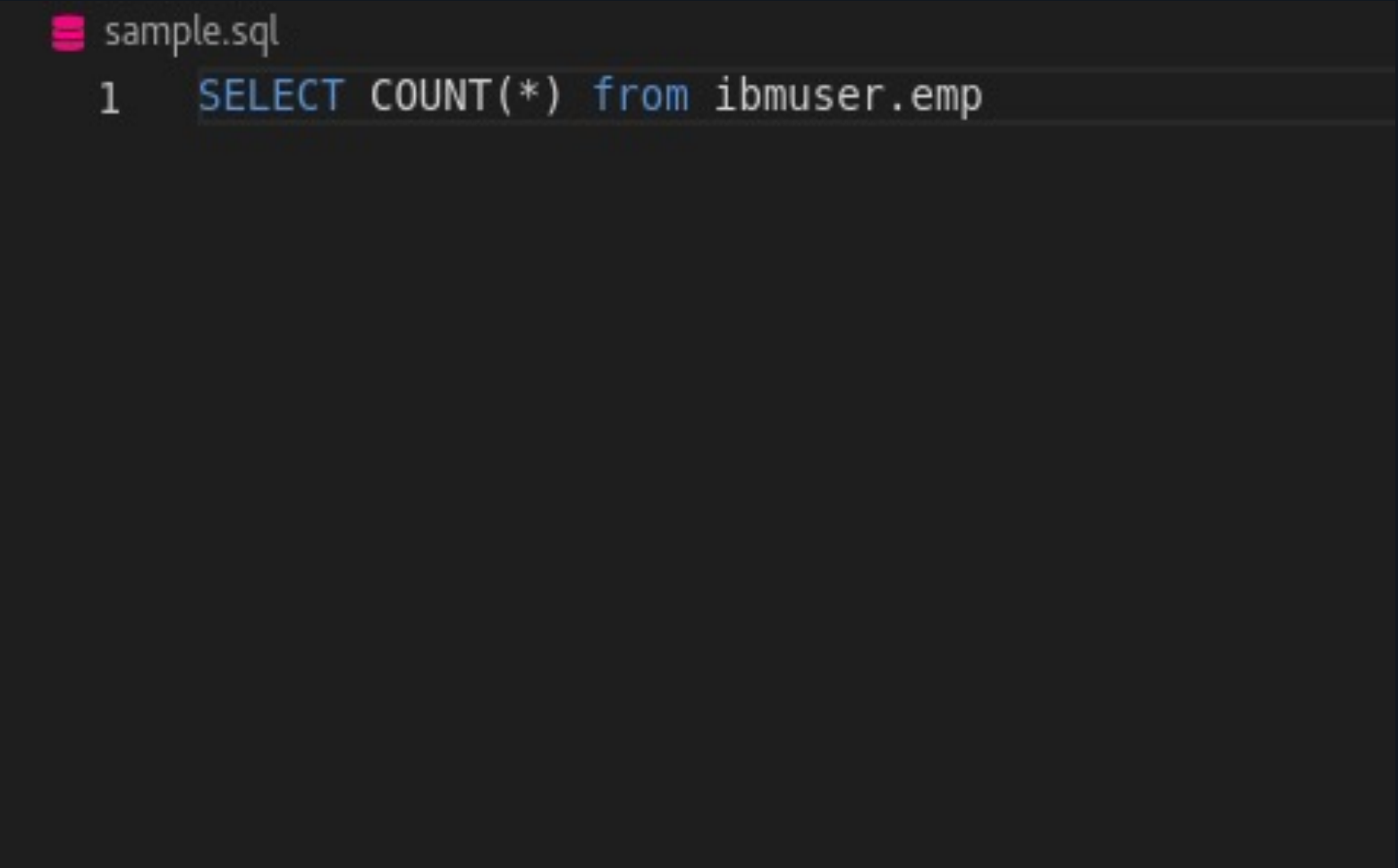
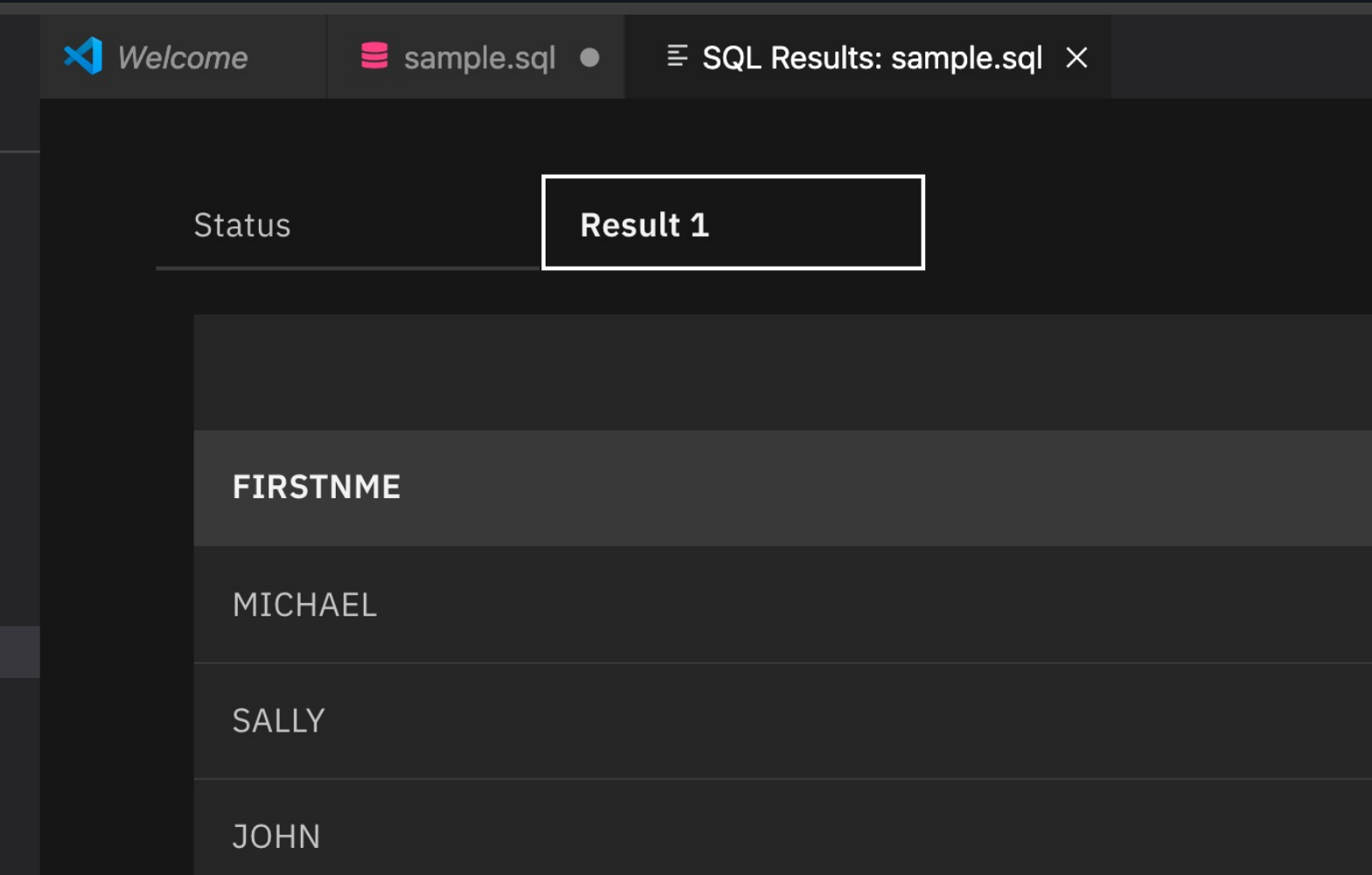
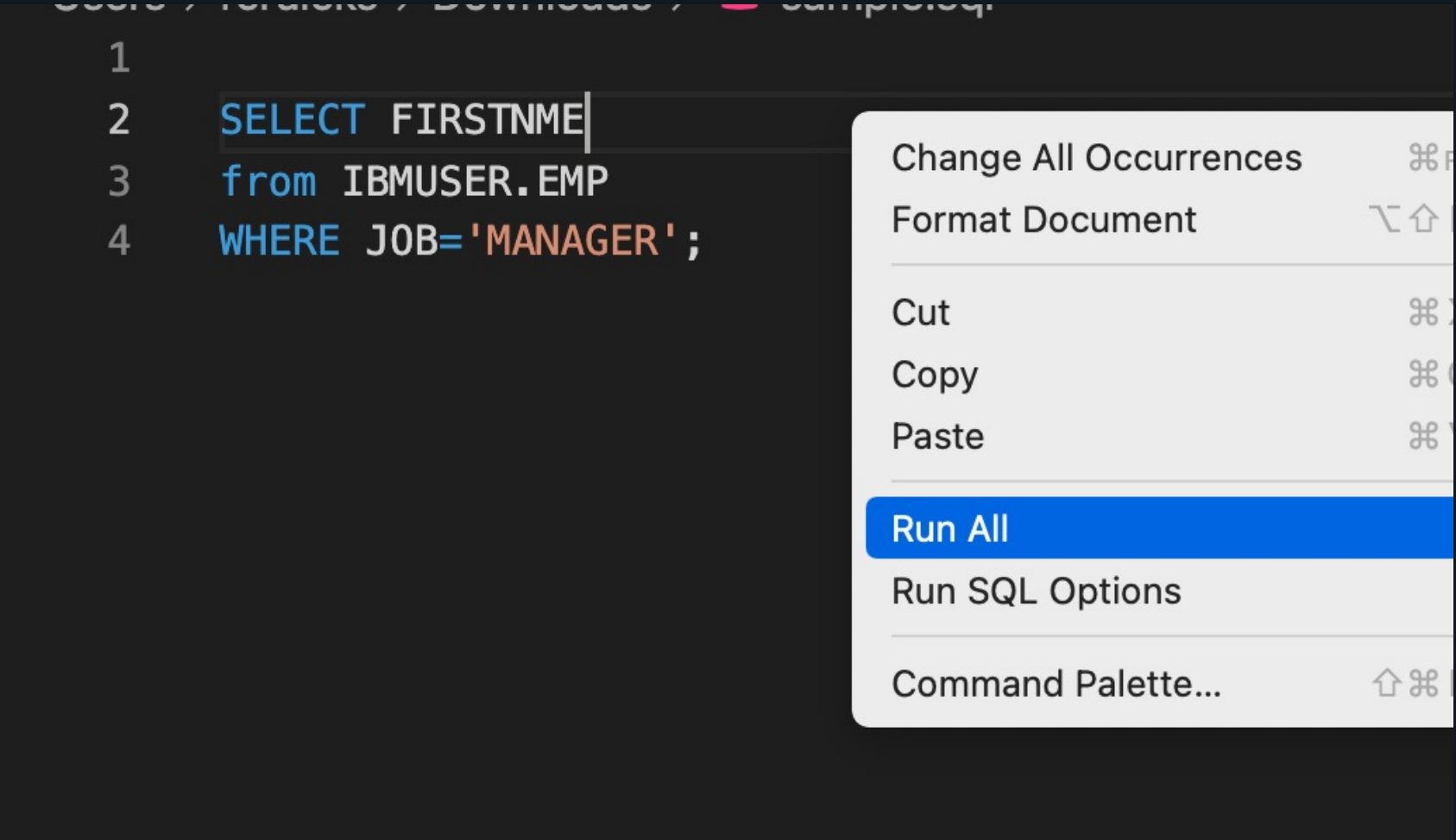
A tabela que você está usando se chama `"ibmuser.emp"` e ela contém alguns dados de funcionários totalmente fictícios. Há também outra tabela que pode ser usada, `"ibmuser.dept"`, e ela contém dados sobre alguns departamentos diferentes dessa empresa fictícia.

Observe o conteúdo completo da tabela e, em seguida, crie sua própria instrução SQL que recriará a tabela na captura de tela acima.

"O que é exatamente o DB2?"

O DB2 é conhecido como um Sistema de Gerenciamento de Banco de Dados Relacional (RDBMS). Ele usa os conceitos de bancos de dados relacionais para armazenar muitos tipos diferentes de dados e, em seguida, otimiza a recuperação deles. Há muitos tipos diferentes de RDBMS e o DB2 é somente a versão da IBM, que existe desde o início dos bancos de dados relacionais. Na verdade, o conceito de banco de dados relacional foi desenvolvido na IBM pelo cientista da computação Edgar Frank “Ted” Codd.

A linguagem usada para conversar com o DB2 é SQL e ela é usada basicamente por todos os RDBMSs modernos.



7. *WHERE* ONDE FOR PRECISO

É possível especificar tanto as colunas quanto as linhas que serão exibidas. Para isso, especifique o valor de uma das colunas para que sejam exibidas somente as linhas que atendem ao valor dela.

Consegue adivinhar o que a consulta acima retornará? Execute a instrução multilinhas da captura de tela acima com a cláusula *WHERE* e descubra. O que você acha que aconteceria com `JOB='manager'`?

8. UM CONJUNTO ESPECIAL DE RESULTADOS

Acesse os resultados da tabela para analisá-los. Observe que somente os funcionários que são gerentes aparecem. Esse é o resultado esperado?

A cláusula *WHERE* é um filtro que extrai somente os registros que atendem à condição especificada. É possível usar sequências de caracteres e números para espeficações, outros operadores como ">" para a filtragem por um intervalo maior ou "<=" para a filtragem por um intervalo menor ou igual.

9. FAÇA AS CONTAS (DE VERDADE!)

Há também algumas funções que permitem contar, calcular a média ou somar as linhas, chamadas de funções agregadas. A sintaxe é bem simples, basta escrever `COUNT(column_name)`, `AVG(column_name)` ou `SUM(column_name)` após `SELECT`. Veja um exemplo na captura de tela acima.

Cada um deles simplesmente utiliza o resultado da consulta e executa a função em cada linha da coluna resultante. Portanto, ao incluir *WHERE* na consulta, ela conta somente a quantidade de linhas que atendem à condição.

Agora, estamos prestes a transformar as perguntas normais que podem ser feitas, como "Há quantos gerentes?", em consultas SQL reais.

Experimente agora mesmo. Veja se você é capaz de criar uma consulta que conte o número de funcionários que são gerentes.

"É possíve mudar o nome de exibição das funções agregadas para algo mais legal?"

Sim. Basta incluir o que será exibido após a função agregada.

Desta forma:
`SELECT COUNT(*) AS NUM_EMPLOYEES from IBMUSER.EMP`

Também é importante notar que essas funções agregadas funcionam em colunas numéricas e não numéricas e contam todas as linhas da coluna que têm uma entrada. Quando o asterisco é inserido, ocorre a contagem do número total de linhas.

Status

Result 1

EMPNO	FIRSTNAME	MIDINIT	LASTNAME	WORKDEPT	PHONENO	HIRE
000010	CHRISTINE	I	HAAS	A00	3978	19/01
000020	MICHAEL	L	THOMPSON	B01	3476	19/10
000030	SALLY	A	KWAN	C01	4738	19/05
000050	JOHN	B	GEYER	E01	6789	19/17
000060	IRVING	F	STERN	D11	6423	19/14
000070	EVA	D	PULASKI	D21	7831	19/30
000090	EILEEN	W	HENDERSON	E11	5498	19/15
000100	THEODORE	Q	SPENSER	E21	0972	19/19
000110	VINCENZO	G	LUCCHESI	A00	3490	19/16
000120	SEAN		O'CONNELL	A00	2167	19/05
000130	DOLORES	M	QUINTANA	C01	4578	19/28
000140	HEATHER	A	NICHOLLS	C01	1793	19/15
000150	BRUCE		ADAMSON	D11	4510	19/12
000160	ELIZABETH	R	PIANKA	D11	3782	19/11
000170	MASATOSHI	J	YOSHIMURA	D11	2890	19/15

Status

Result 1

DEPTNO	DEPTNAME	MGRNO	ADMRDEPT	LOCATION
A00	SPIFFY COMPUTER SERVICE DIV.	000010	A00	
B01	PLANNING	000020	A00	
C01	INFORMATION CENTER	000030	A00	
D01	DEVELOPMENT CENTER	*NULL*	A00	
D11	MANUFACTURING SYSTEMS	000060	D01	
D21	ADMINISTRATION SYSTEMS	000070	D01	
E01	SUPPORT SERVICES	000050	A00	
E11	OPERATIONS	000090	E01	
E21	SOFTWARE SUPPORT	000100	E01	
F22	BRANCH OFFICE F2	*NULL*	E01	
G22	BRANCH OFFICE G2	*NULL*	E01	
H22	BRANCH OFFICE H2	*NULL*	E01	
I22	BRANCH OFFICE I2	*NULL*	E01	
J22	BRANCH OFFICE J2	*NULL*	E01	

Items per page: 25 1-14 of 14 items 1 of 1 page

10. TUDO SOBRE JUNÇÕES

Uma junção é o processo de combinar dados de duas ou mais tabelas com base em algum domínio comum de informações. As linhas de uma tabela são emparelhadas com as linhas de outra quando as informações nas linhas correspondentes atendem aos critérios de junção.

Para isso, mais uma vez, uma cláusula WHERE é útil. Basta relacionar duas colunas comuns de duas tabelas diferentes configurando ambas como iguais entre si na cláusula WHERE. O exemplo acima é das tabelas de funcionários e departamentos, que podem ser unidas na coluna WORKDEPT comum e na coluna DEPTNO.

"É possível juntar mais de duas tabelas?"

Sim. Isso é chamado de junção multitabelas. O processo é bastante semelhante em estrutura à junção de duas tabelas, mas é necessário incluir uma cláusula AND após WHERE que especifica uma junção com outra tabela.

A aparência será como a seguir:
SELECT company, stock_num, menu_code
FROM customer c, orders o, items i
WHERE c.customer_num = o.customer_num
AND o.order_num = i.order_num;

11. CRIANDO JUNÇÕES

Execute o exemplo acima.

Observe como essa consulta junta IBMUSER.DEPT e IBMUSER.EMP nas colunas DEPTNO e WORKDEPT.Observe também uma variável fornecida na cláusula FROM que representa cada tabela (D é ibmuser.dept e E é ibmuser.emp). Essas variáveis são conhecidas como “identificadores de correlação”.

Depois que você realizar a consulta, veremos algumas outras coisas relacionadas à sintaxe.

```
sample.sql
1 SELECT DEPTNAME, COUNT(EMPNO)
2 FROM ibmuser.dept D, ibmuser.emp E
3 WHERE D. DEPTNO = E.WORKDEPT
4 GROUP BY DEPTNAME
5 ORDER BY DEPTNAME
```

Status

Result 1

DEPTNAME	2
ADMINISTRATION SYSTEMS	7
INFORMATION CENTER	4
MANUFACTURING SYSTEMS	11
OPERATIONS	7

12. SIMPLEMENTE RESULTADOS FILTRADOS

Observe que o resultado conta com sucesso o número de funcionários de cada departamento. Para isso, a consulta usou a cláusula GROUP BY. Essa cláusula agrupa linhas com os mesmos valores em linhas de resumo. Isso é comumente usado para funções agregadas (count, sum, average, max, min), como no nosso exemplo.

Pode ser um pouco confuso compreender GROUP BY, por isso, teste a consulta, veja o que funciona ou não e experimente diferentes combinações para ver o que elas produzem.

A última parte do desafio é a cláusula ORDER BY, que simplesmente exibe as informações em ordem alfabética pela coluna especificada.

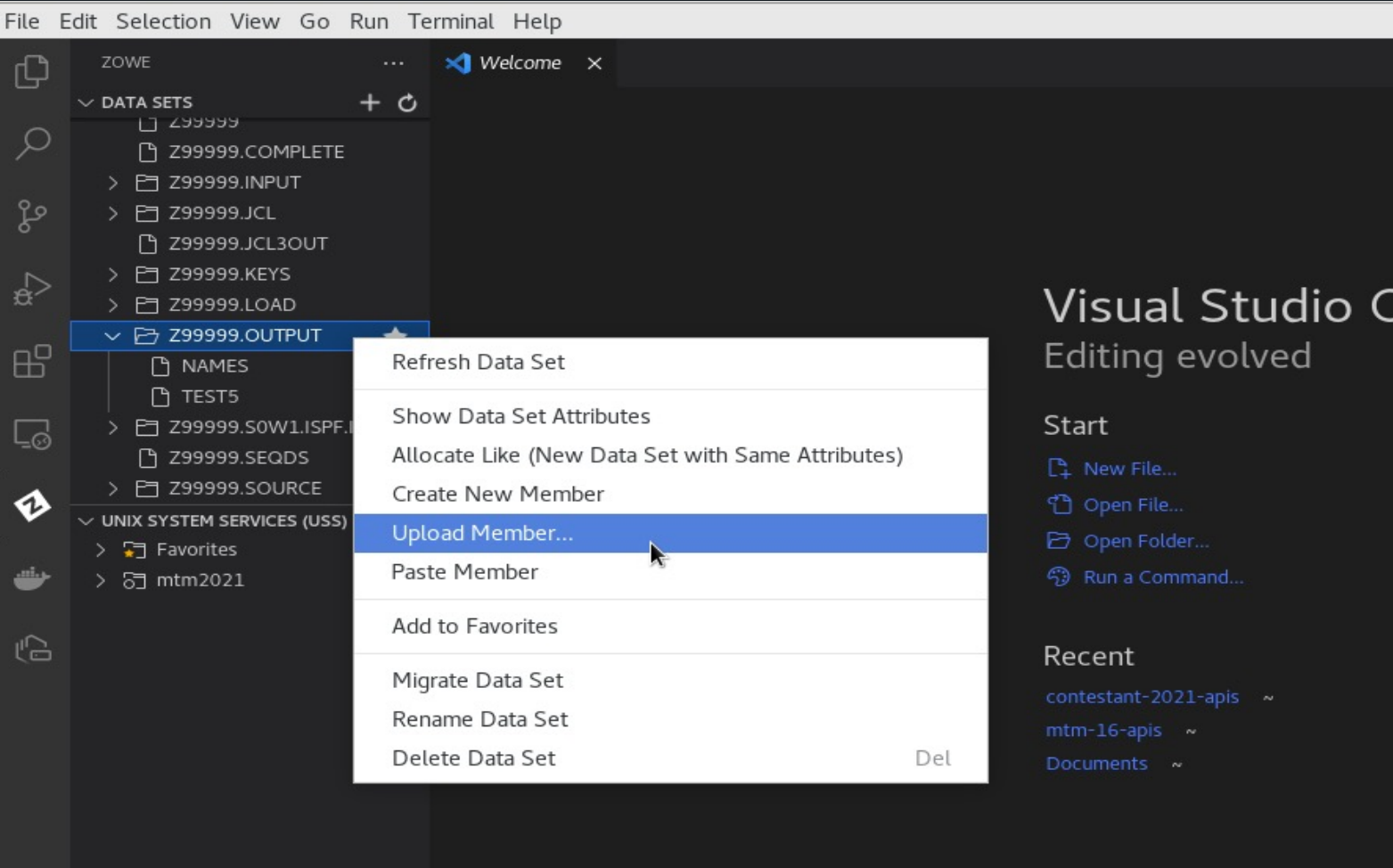
```
sample.sql
1 SELECT MAX(SALARY) SAL
2   from ibmuser.emp
3
```

Status

Result 1

Save results

DEPTNAME	MAX_COMPENSATION
ADMINISTRATION SYSTEMS	XXX XX XX
INFORMATION CENTER	XXX XX XX
MANUFACTURING SYSTEMS	XXX XX XX



13. ESCOLHA MAX

Mais uma novidade sobre o SQL para você aumentar seus conhecimentos são as funções min e max. Elas são o que parecem, ou seja, retornam o menor valor e o maior valor, respectivamente, das colunas especificadas. Elas são usadas da mesma forma que count, sum e avg. Também é possível nomear a saída dessas funções como você quiser incluindo um nome imediatamente após a cláusula da função, conforme mostrado na captura de tela acima.

Mais uma vez, teste as funções para familiarizar-se antes de realizar o desafio!

Uma das melhores maneiras de aprender é experimentar por conta própria e observar os resultados de diferentes combinações de sintaxe.

14. ORGANIZE TUDO

Seu desafio é consultar a remuneração máxima (todas as formas de pagamento de alguém) em cada departamento pelo nome do departamento. Exiba DEPTNAME e MAX_COMPENSATION. Mantenha em mente tudo o que você aprendeu sobre a sintaxe SQL, pois isso será necessário.Saiba todas as colunas de cada tabela. Uma boa prática é fazer uma lista delas. Quando você tiver a saída desejada, faça download dela como um arquivo .csv clicando no ícone de arquivo na guia de resultados. Isso fará o download dela em sua máquina.

Dica: há mais de uma coluna que conta como compensação!

15. ÚLTIMA VERIFICAÇÃO PARA ENCERRAR

É hora do envio. Depois de fazer download do arquivo .csv, clique com o botão direito em seu conjunto de dados ZXXXXX.OUTPUT e em “Fazer upload do membro”. Em seguida, faça upload do seu arquivo de envio .csv. Após o upload, renomeie o arquivo como DB2OUT para encontrá-lo, e pronto.

Agora é hora de executar a verificação para ver se a consulta estava correta. Clique com o botão direito em ‘ZXP.PUBLIC.JCL(CHKSQL)’ e envie essa tarefa. Bom trabalho!

BOM TRABALHO! VAMOS RECAPITULAR

Você aprendeu muitas coisas novas sobre bancos de dados relacionais e sintaxes SQL. Lembre-se de que o DB2 é somente um sistema de banco de dados relacional, como muitos outros, mas que o SQL permanece sempre o mesmo. Agora você sabe como consultar todos os tipos de informações de tabelas e como juntar tabelas para relacionar informações em tabelas diferentes.

Isso é bem legal.

EM SEQUÊNCIA...

Confira os outros desafios avançados disponíveis

