

# Transforming Db2 for z/OS Application Development Experience

—  
Katherine Soohoo

[katherine.sooahoo@ibm.com](mailto:katherine.sooahoo@ibm.com)

Technical Lead Developer, Db2 Developer Extension

# Agenda

- Overview
- Installation
- Features
- Demo
- What's next?

# Overview

# Meet our persona

## Deb, New z/OS Developer

- Code, debug, and tune application programs (written in Java, COBOL, PL1, etc.) that access Db2 for z/OS via SQL
- Code, run, and tune SQL statements
- Code, deploy, and debug Db2 for z/OS stored procedures



# Pain points

## Deb, New z/OS Developer

- Requires Z domain specific skills to get started
- Needs to interact with different tools on different platforms
- Reduced productivity



# Db2 Developer Extension

An extension for Visual Studio Code that provides language support for developing Db2 for z/OS SQL applications.



# Why VS Code?

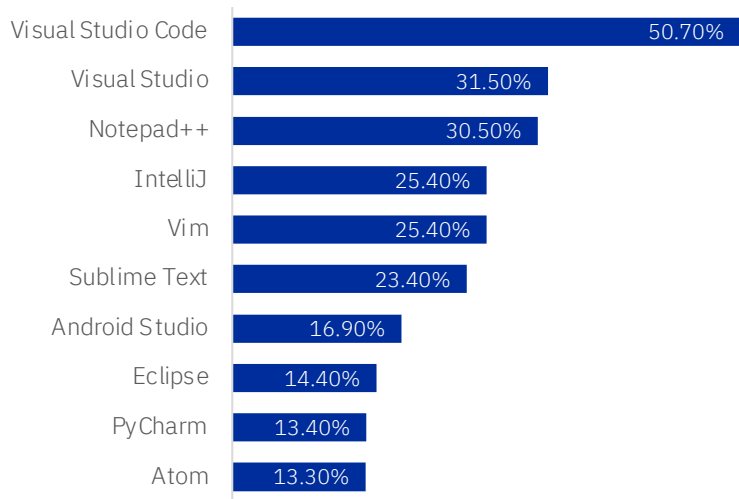
- Lightweight
- Rich git integration
- Integrated terminal support
- Broad language support
- Extensive and active extension support
- Z support with other extensions (Z Open Editor, Zowe Explorer)
- Extensions deployable to other platforms (Eclipse Theia, RedHat CodeReady Workspaces, etc.)



# Why VS Code?

Stack Overflow Developer Survey 2019

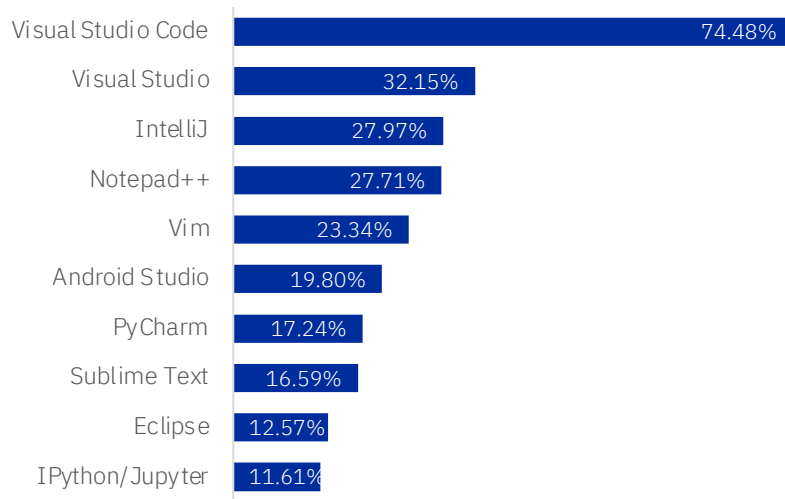
■ IDE used



Data from [Stack Overflow Developer Survey 2019](#)

Stack Overflow Developer Survey 2022

■ IDE used



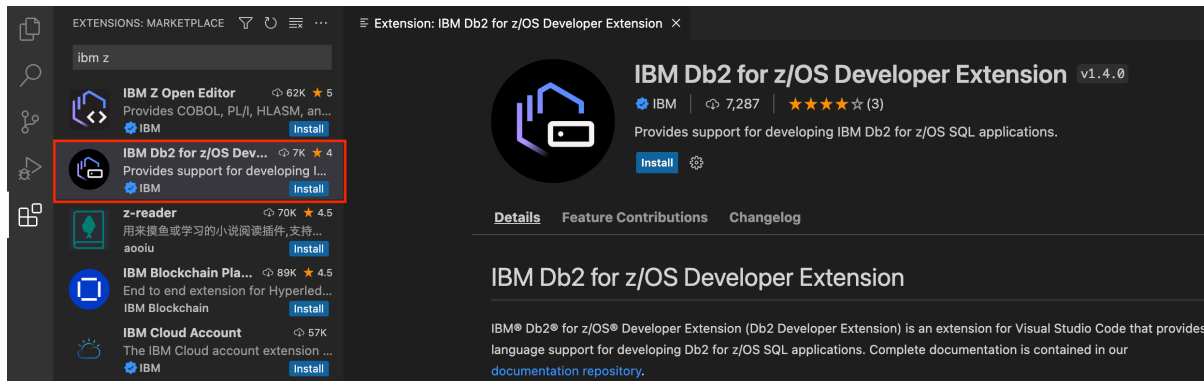
Data from [Stack Overflow Developer Survey 2022](#)



# Installation

# How to install

1. Install and open VS Code
2. Switch to the Extensions view
3. Search for "ibm z"
4. Click the "Install" button for "IBM Db2 for /OS Developer Extension"

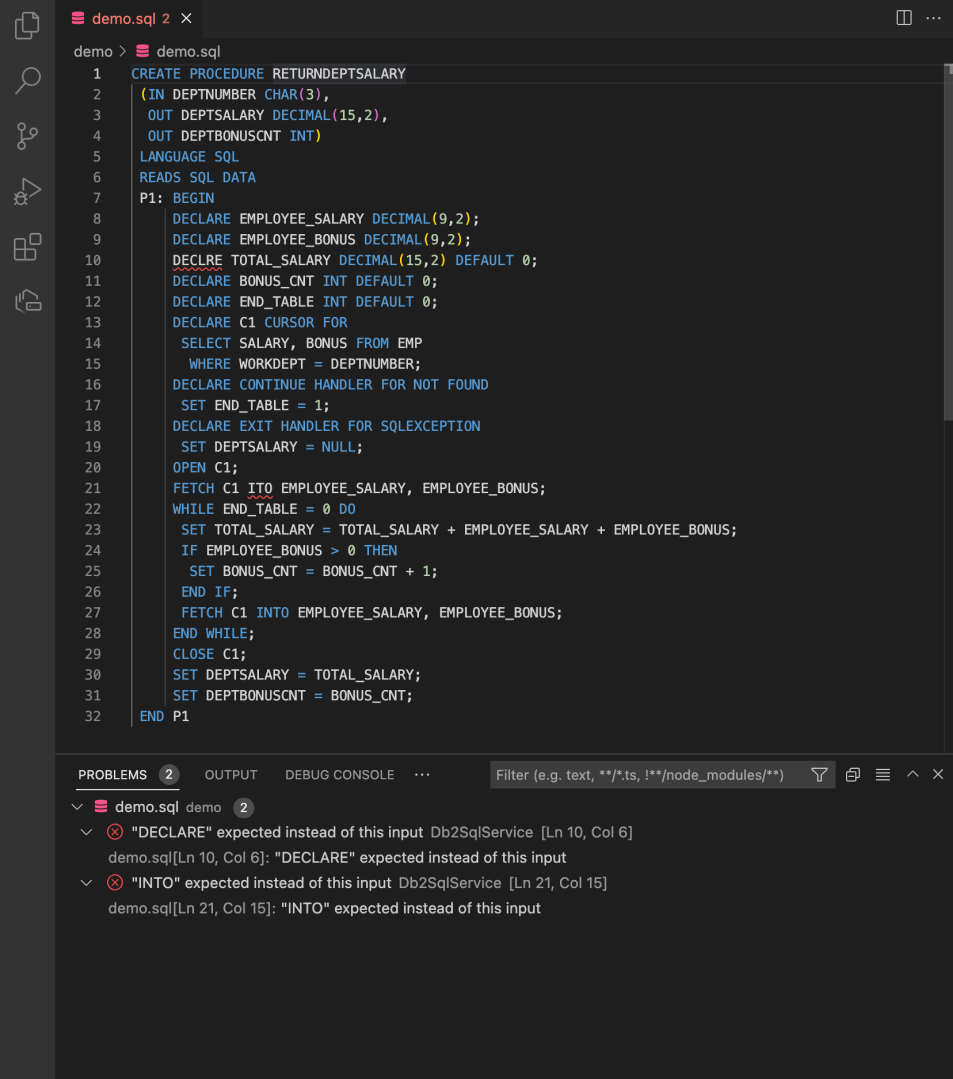


# Features

- SQL language support
  - Syntax highlighting
  - Syntax checking
  - Code completion
  - Signature help
  - Format SQL statements
- Run SQL statements
- Deploy, run, and debug native stored procedures
- Tune SQL
  - Visual Explain
  - Statistics Advisor
  - Capture Query Environment

# Syntax highlighting and syntax checking

- Improves readability of SQL statements by highlighting keywords
- Checks for SQL syntax errors and provides hints to resolve errors



The screenshot displays a code editor with a dark theme. The top panel shows a SQL script with syntax highlighting. The script is a stored procedure named `RETURNDEPTSALARY` that takes a department number as input and returns the total salary and bonus count for that department. The script includes variable declarations, a cursor, and a loop to calculate the total salary and bonus count.

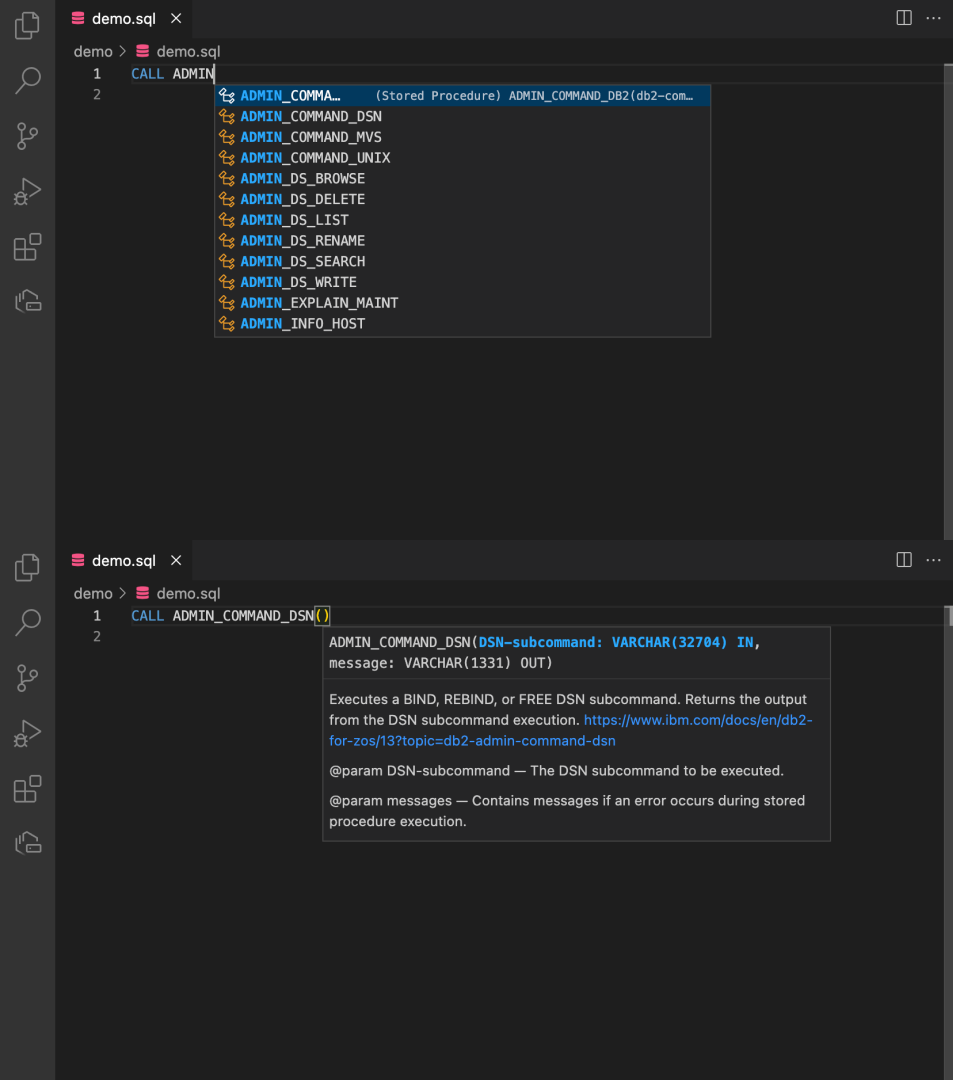
```
demo > demo.sql
1 CREATE PROCEDURE RETURNDEPTSALARY
2   (IN DEPTNUMBER CHAR(3),
3   OUT DEPTSALARY DECIMAL(15,2),
4   OUT DEPTBONUSCNT INT)
5   LANGUAGE SQL
6   READS SQL DATA
7   P1: BEGIN
8     DECLARE EMPLOYEE_SALARY DECIMAL(9,2);
9     DECLARE EMPLOYEE_BONUS DECIMAL(9,2);
10    DECLRE TOTAL_SALARY DECIMAL(15,2) DEFAULT 0;
11    DECLARE BONUS_CNT INT DEFAULT 0;
12    DECLARE END_TABLE INT DEFAULT 0;
13    DECLARE C1 CURSOR FOR
14      SELECT SALARY, BONUS FROM EMP
15      WHERE WORKDEPT = DEPTNUMBER;
16    DECLARE CONTINUE HANDLER FOR NOT FOUND
17      SET END_TABLE = 1;
18    DECLARE EXIT HANDLER FOR SQLEXCEPTION
19      SET DEPTSALARY = NULL;
20    OPEN C1;
21    FETCH C1 INTO EMPLOYEE_SALARY, EMPLOYEE_BONUS;
22    WHILE END_TABLE = 0 DO
23      SET TOTAL_SALARY = TOTAL_SALARY + EMPLOYEE_SALARY + EMPLOYEE_BONUS;
24      IF EMPLOYEE_BONUS > 0 THEN
25        SET BONUS_CNT = BONUS_CNT + 1;
26      END IF;
27      FETCH C1 INTO EMPLOYEE_SALARY, EMPLOYEE_BONUS;
28    END WHILE;
29    CLOSE C1;
30    SET DEPTSALARY = TOTAL_SALARY;
31    SET DEPTBONUSCNT = BONUS_CNT;
32  END P1
```

The bottom panel shows the PROBLEMS tab with two errors listed:

- ❌ "DECLARE" expected instead of this input Db2SqlService [Ln 10, Col 6]  
demo.sql[Ln 10, Col 6]: "DECLARE" expected instead of this input
- ❌ "INTO" expected instead of this input Db2SqlService [Ln 21, Col 15]  
demo.sql[Ln 21, Col 15]: "INTO" expected instead of this input

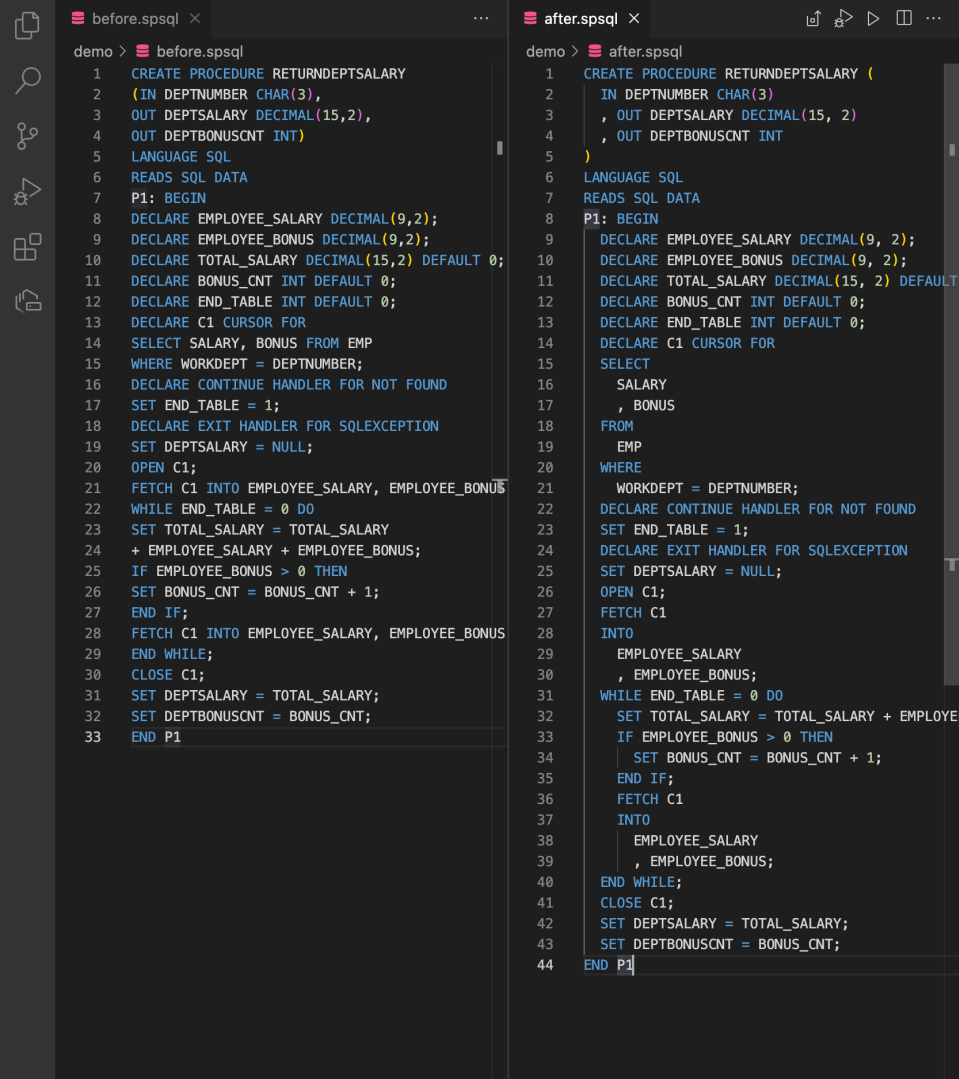
# Code completion and signature help

- Suggests names for built-in Db2 functions and Db2-supplied stored procedures
- Provides parameter information for built-in Db2 functions and Db2-supplied stored procedures



# Format SQL statements

- Easier to parse large blocks of code and understand the relationship between different blocks of SQL elements and clauses



```
demo > before.spsql
1 CREATE PROCEDURE RETURNDEPTSALARY
2 (IN DEPTNUMBER CHAR(3),
3 OUT DEPTSALARY DECIMAL(15,2),
4 OUT DEPTBONUSCNT INT)
5 LANGUAGE SQL
6 READS SQL DATA
7 P1: BEGIN
8 DECLARE EMPLOYEE_SALARY DECIMAL(9,2);
9 DECLARE EMPLOYEE_BONUS DECIMAL(9,2);
10 DECLARE TOTAL_SALARY DECIMAL(15,2) DEFAULT 0;
11 DECLARE BONUS_CNT INT DEFAULT 0;
12 DECLARE END_TABLE INT DEFAULT 0;
13 DECLARE C1 CURSOR FOR
14 SELECT SALARY, BONUS FROM EMP
15 WHERE WORKDEPT = DEPTNUMBER;
16 DECLARE CONTINUE HANDLER FOR NOT FOUND
17 SET END_TABLE = 1;
18 DECLARE EXIT HANDLER FOR SQLEXCEPTION
19 SET DEPTSALARY = NULL;
20 OPEN C1;
21 FETCH C1 INTO EMPLOYEE_SALARY, EMPLOYEE_BONUS;
22 WHILE END_TABLE = 0 DO
23 SET TOTAL_SALARY = TOTAL_SALARY
24 + EMPLOYEE_SALARY + EMPLOYEE_BONUS;
25 IF EMPLOYEE_BONUS > 0 THEN
26 SET BONUS_CNT = BONUS_CNT + 1;
27 END IF;
28 FETCH C1 INTO EMPLOYEE_SALARY, EMPLOYEE_BONUS;
29 END WHILE;
30 CLOSE C1;
31 SET DEPTSALARY = TOTAL_SALARY;
32 SET DEPTBONUSCNT = BONUS_CNT;
33 END P1

demo > after.spsql
1 CREATE PROCEDURE RETURNDEPTSALARY (
2 IN DEPTNUMBER CHAR(3)
3 , OUT DEPTSALARY DECIMAL(15, 2)
4 , OUT DEPTBONUSCNT INT
5 )
6 LANGUAGE SQL
7 READS SQL DATA
8 P1: BEGIN
9 DECLARE EMPLOYEE_SALARY DECIMAL(9, 2);
10 DECLARE EMPLOYEE_BONUS DECIMAL(9, 2);
11 DECLARE TOTAL_SALARY DECIMAL(15, 2) DEFAULT 0;
12 DECLARE BONUS_CNT INT DEFAULT 0;
13 DECLARE END_TABLE INT DEFAULT 0;
14 DECLARE C1 CURSOR FOR
15 SELECT
16 SALARY
17 , BONUS
18 FROM
19 EMP
20 WHERE
21 WORKDEPT = DEPTNUMBER;
22 DECLARE CONTINUE HANDLER FOR NOT FOUND
23 SET END_TABLE = 1;
24 DECLARE EXIT HANDLER FOR SQLEXCEPTION
25 SET DEPTSALARY = NULL;
26 OPEN C1;
27 FETCH C1
28 INTO
29 EMPLOYEE_SALARY
30 , EMPLOYEE_BONUS;
31 WHILE END_TABLE = 0 DO
32 SET TOTAL_SALARY = TOTAL_SALARY + EMPLOYEE_SALARY + EMPLOYEE_BONUS;
33 IF EMPLOYEE_BONUS > 0 THEN
34 SET BONUS_CNT = BONUS_CNT + 1;
35 END IF;
36 FETCH C1
37 INTO
38 EMPLOYEE_SALARY
39 , EMPLOYEE_BONUS;
40 END WHILE;
41 CLOSE C1;
42 SET DEPTSALARY = TOTAL_SALARY;
43 SET DEPTBONUSCNT = BONUS_CNT;
44 END P1
```

# Connect to Db2 for z/OS

- Supports different authentication methods:
  - User ID and password
  - RACF PassTicket
  - Multi-factor authentication (MFA) token
- Specify JDBC properties
- Enable JDBC trace

**Add Db2 Connection** X

**Db2 connection information**

Location name  
Location name of the Db2 subsystem

Host  
Hostname or IP address of the Db2 subsystem

Port  
Port number assigned to the hostname or IP address of the Db2 subsystem

Connection name (optional)  
Unique name for the connection profile

Connection URL  
Auto-generated connection URL

**User information**

Login method  
☒ Password ☐ PassTicket ☐ MFA token

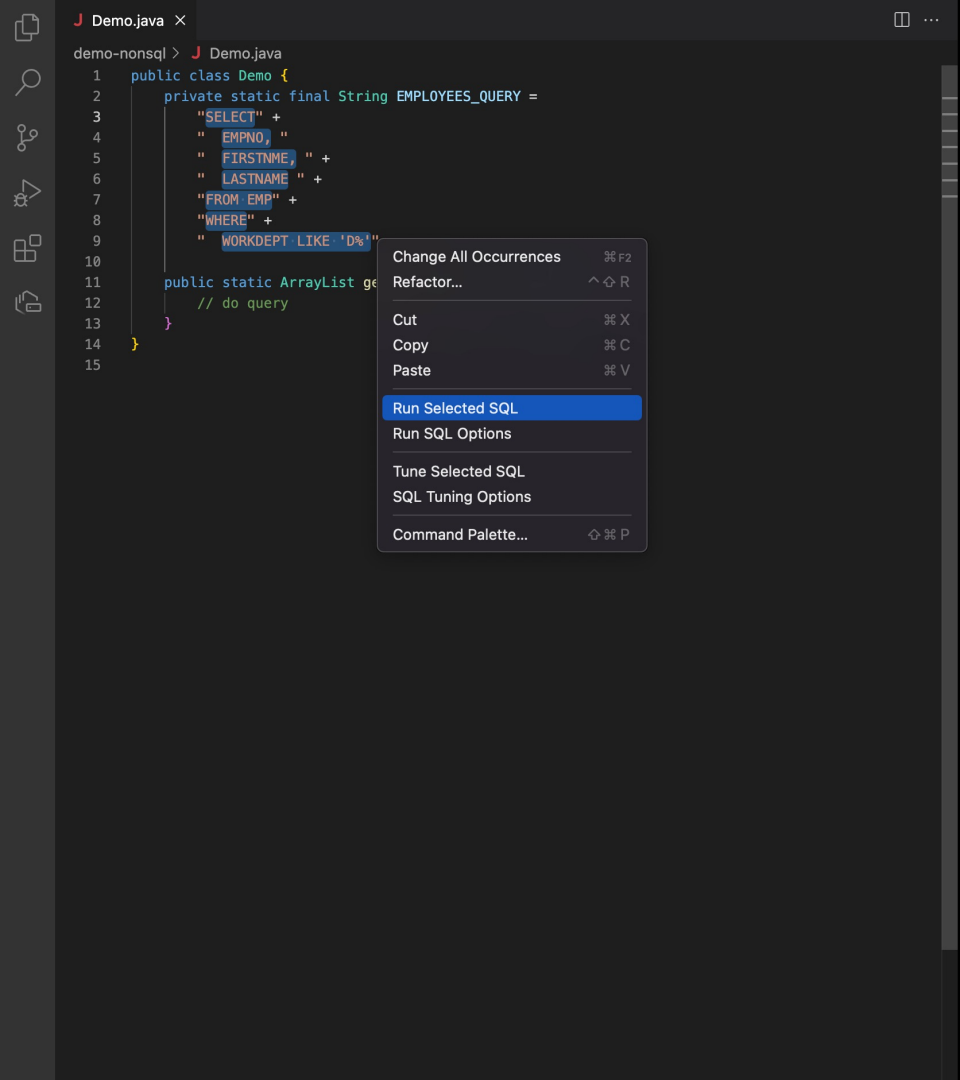
User ID  
Valid Db2 user ID to connect to the Db2 subsystem

Password  
Password for the Db2 user ID

☐ Save password

# Run SQL

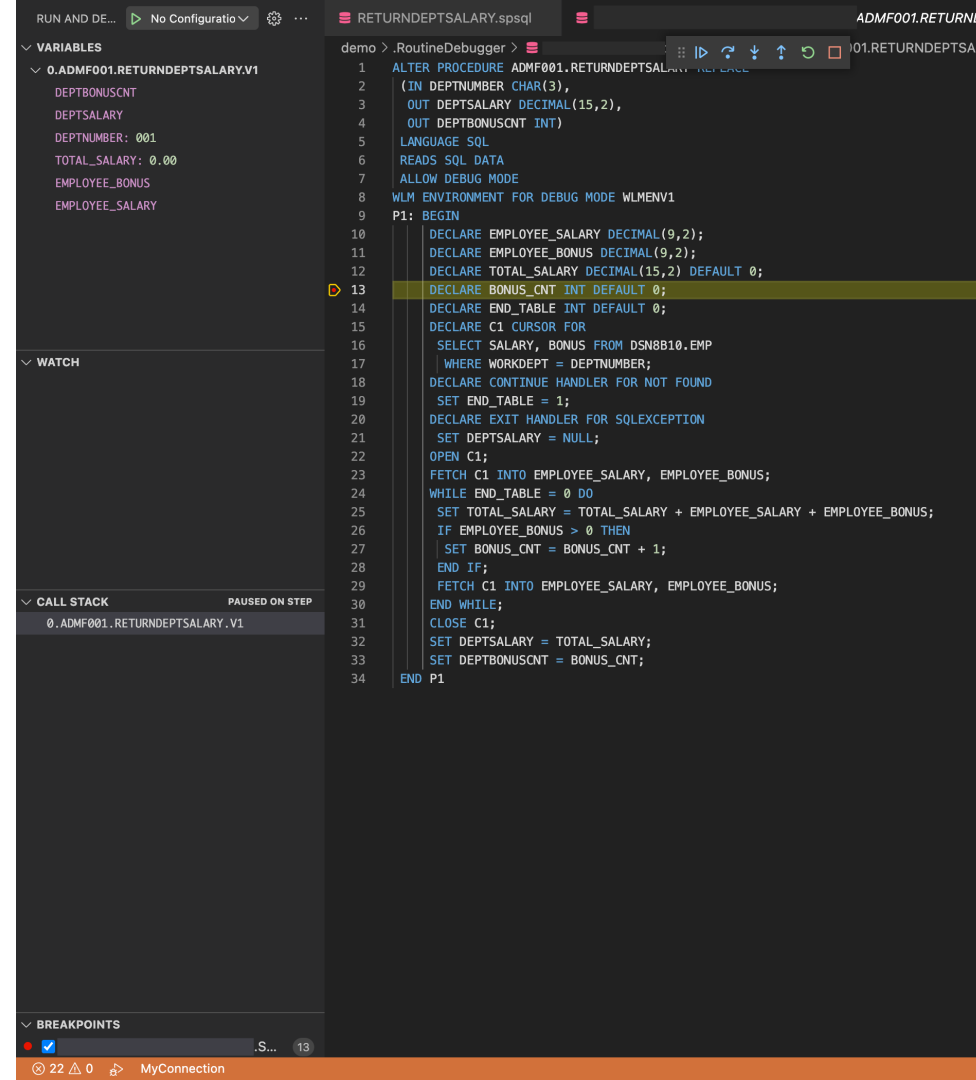
- Run selected SQL from any type of file
- Run all SQL statements in a file
- Change commit and rollback options





# Native stored procedures

- Deploy, run, and debug native stored procedures
- File extension must end with .spsql



# Tune SQL

- View graphical representation of access path
- Generate statistics to improve query performance
- Capture statement's environment to troubleshoot problems when tuning an SQL statement

Tuning Result: Visual Explain

SQL Statement Warnings Environment & Explain Options

Query

Description Search

Properties

NAME	VALUE
Index only	N
Type	I
Matching Columns	1
Cost Information	
Input RIDs	42
Index leaf pages	1
Scanned leaf pages	1
Scanned RIDs	7
Scanned rows	7
Output RIDs	7
Matching Filter Factor	0.16666663
Total Filter Factor	0.16666663

Predicates

> Matching predicates 1

```
graph TD
    Q1["(1) QUERY"] --> NLJOIN3["(3) NLJOIN 12"]
    NLJOIN3 --> F4["(4) FETCH 7"]
    NLJOIN3 --> F8["(8) FETCH 2"]
    F4 --> IXSCAN5["(5) IXSCAN 7"]
    F4 --> EMP7["(7) EMP 42"]
    IXSCAN5 --> XEMP26["(6) XEMP2 8"]
    F8 --> IXSCAN9["(9) IXSCAN 2"]
    F8 --> EMPPROJECT11["(11) EMPPROJECT 74"]
    IXSCAN9 --> XEMPPROJECT210["(10) XEMPPROJECT2 30"]
```

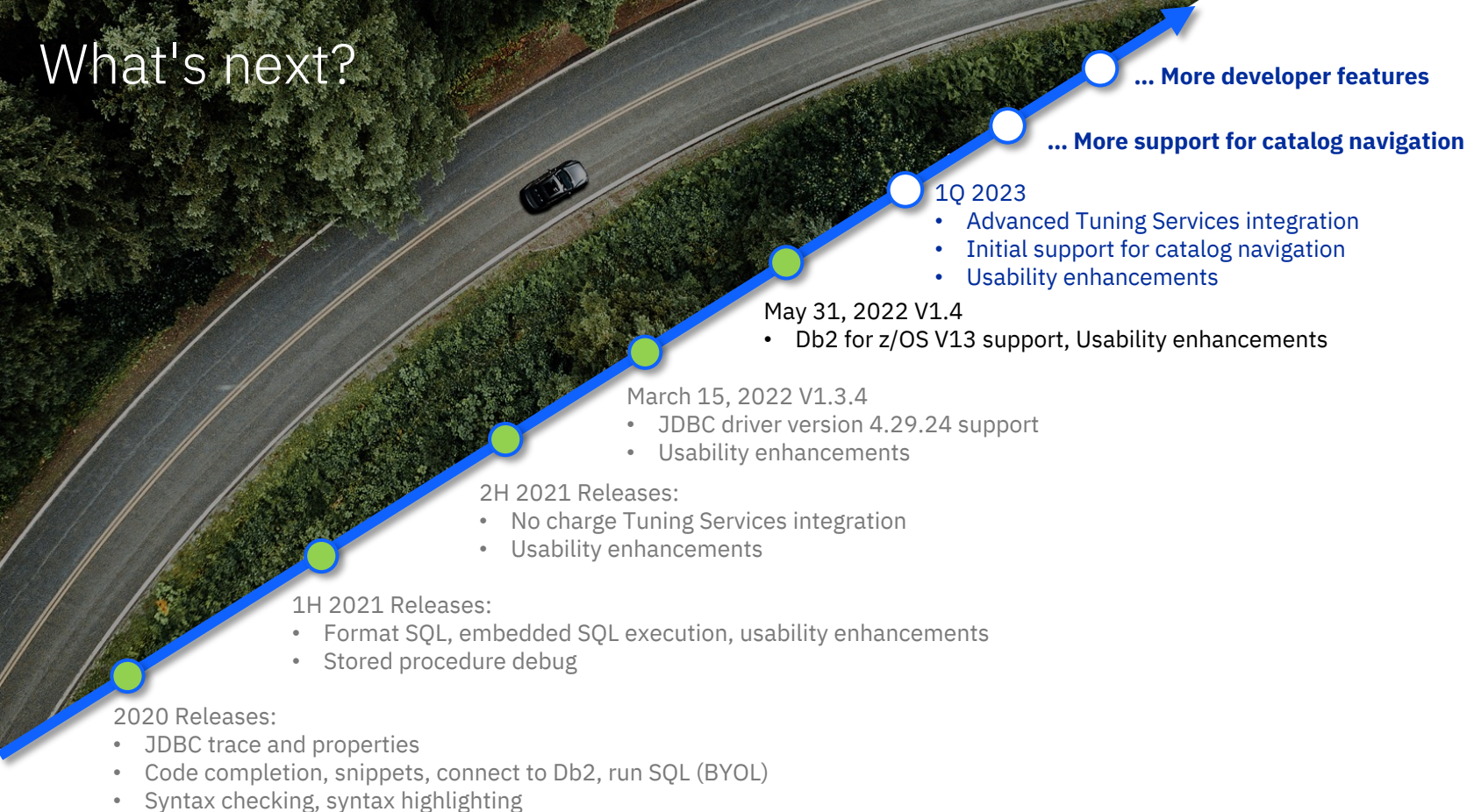
# Prerequisites for SQL Tuning Services

- [SQL Tuning Services documentation](#)
- [IBM DB2 Accessories Suite for z/OS, 4.2](#)
- [APAR PH42944](#)

# Demo

- SQL language support
- Connecting to Db2 and running SQL
- Working with native stored procedures
- Tuning SQL
- Running SQL from z/OS (with Zowe Explorer)

# What's next?



# Advanced tuning feature: Index Advisor

## Summary

Number of recommendations: 2

Number of existing indexes: 8

CPU cost saving: 93.8730853391685%

## Candidate indexes

Table	DDL	Explanation	Key columns ⓘ	Full key cardinality	Index size
SYSIBM.SYSCOLUMNS	CREATE INDEX "SYSIBM"."SYSCOLUMNS_VIRT_IDX_166443307115256240" ON "SYSIBM"."SYSCOLUMNS" ( "TBNAME" ASC ) NOT PADDED FREEPAGE 0 PCTFREE 10	Supports join predicates.	TBNAME(ASC):-1	370	0.1015625MB
SYSIBM.SYSTABLES	CREATE INDEX "SYSIBM"."SYSTABLES_VIRT_IDX_166443307114184282" ON "SYSIBM"."SYSTABLES" ( ( "DBID" + 1 ) ASC ) NOT PADDED FREEPAGE 0 PCTFREE 10	Supports local predicates.	( "DBID" + 1 ) (ASC):20	20	0.0234375MB

## Existing indexes

Index	Table	Key columns ⓘ	Unique	Clustering	DPSI	Exclude null keys	Include columns	Time last used
SYSIBM.DSNDXC01	SYSIBM.SYSCOLUMNS	TBCREATOR(ASC):-1 TBNAME(ASC):-1 NAME(ASC):-1	Y	N	N	N		2022-09-28
SYSIBM.DSNDXC02	SYSIBM.SYSCOLUMNS	TYPESCHEMA(ASC):-1 TYPENAME(ASC):-1	N	N	N	N		
SYSIBM.DSNDXC05	SYSIBM.SYSCOLUMNS	TBCREATOR(ASC):-1 TBNAME(ASC):-1	N	N	N	N		2022-09-28
SYSIBM.DSNDXC06	SYSIBM.SYSCOLUMNS	TBCREATOR(ASC):-1 TBNAME(ASC):-1	N	N	N	N		2022-09-26

## Advanced tuning feature: Access Path Advisor

## Summary

Health status: Bad

Number of recommendations: 3

High severity recommendations: 1

Medium severity recommendations: 0

Low severity recommendations: 2

## Recommendations

1	Severity: High	Avoid reading all index keys on an index scan (QBLOCKNO = 1, PLANNO = 2). The table SYSIBM.SYSCOLUMNS is accessed by a non-matching index scan (QBLOCKNO = 1, PLANNO = 2). If a table is accessed by non-matching index scan, then all the index keys and their RIDs are read. When a large number of keys and RIDs are accessed, Db2 might be using an inefficient access path. Consider run the Statistics Advisor or run the Index Advisor to determine whether creating an index might improve the access path.	▼
2	Severity: Low	Avoid table space scans (QBLOCKNO = 1, PLANNO = 1) on table SYSIBM.SYSTABLES. The table is accessed by a table space scan. Consider running the Statistics Advisor, because the improving statistics might improve the access path. Also, consider running the Index Advisor to determine whether creating an index might improve the access path.	▼
3	Severity: Low	Avoid sorting (QBLOCKNO = 1, PLANNO = 3) on a large number of records. A sort is used. When a large number of records are returned before sorting, Db2 might be using an inefficient access path. Consider rewriting the query or designing an index to avoid the sort if possible.	^
Explanation		Db2 uses a sort to process operations such as join processing, GROUP BY operations, ORDER BY operations, the removal of duplicates, and sub-query processing. If the number of rows to be sorted is large, the cost of the sort will be high. You can use indexes to order data, sometimes eliminating the need for sorting. Some sorts can be avoided if index keys are in the order needed by ORDER BY, GROUP BY, a join operation, or DISTINCT in an aggregate function.	
Example		Consider the following query: SELECT C1,C2,C3 FROM T WHERE C1 > 1 ORDER BY C1 OPTIMIZE FOR 1 ROW; An ascending index on C1 or an index on (C1, C2, C3) might eliminate a sort. OPTIMIZE FOR 1 ROW can also be used to avoid sorts because it can have a significant effect on the access path.	

# Resources

## IBM Db2 for z/OS Developer Extension

- VS Code Marketplace:  
<https://marketplace.visualstudio.com/items?itemName=IBM.db2forzosdeveloperextension>
- Documentation:  
<https://ibm.github.io/db2forzosdeveloperextension-about/>
- Report issues or give suggestions:  
<https://github.com/IBM/db2forzosdeveloperextension-about/issues>

## Other Resources

- Visual Studio Code:  
<https://code.visualstudio.com/>
- SQL Tuning Services:  
<https://www.ibm.com/docs/en/db2-for-zos/13?topic=db2-sql-tuning-services>
- IBM Z Open Editor:  
<https://marketplace.visualstudio.com/items?itemName=IBM.zopeneditor>
- Zowe Explorer:  
<https://marketplace.visualstudio.com/items?itemName=Zowe.vscode-extension-for-zowe>



