



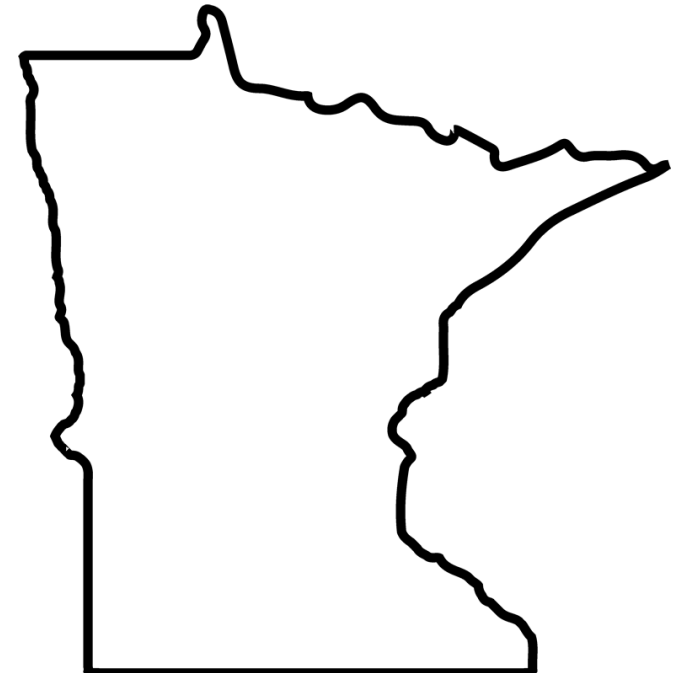
SQL Development in a Developer's World

Dec 12, 2020

Drew Skwiers-Koballa, he/him

Program Manager

Drew.SkwiersKoballa@microsoft.com



Who?

Drew Skwiers-Koballa (squire's co- ball –a)

he/him

Former small company DB “person” and dev
Into pugs, chickens, gardening, and SQL

Program Manager @ Microsoft

Ask me about:

- Starting a new job remotely
- Moving across the country
- SQL tools

Agenda

-
- (Database) Developers
 - SQL database projects
 - Development environments

Roles that interact with databases:

Application
Developer

Database
Developer

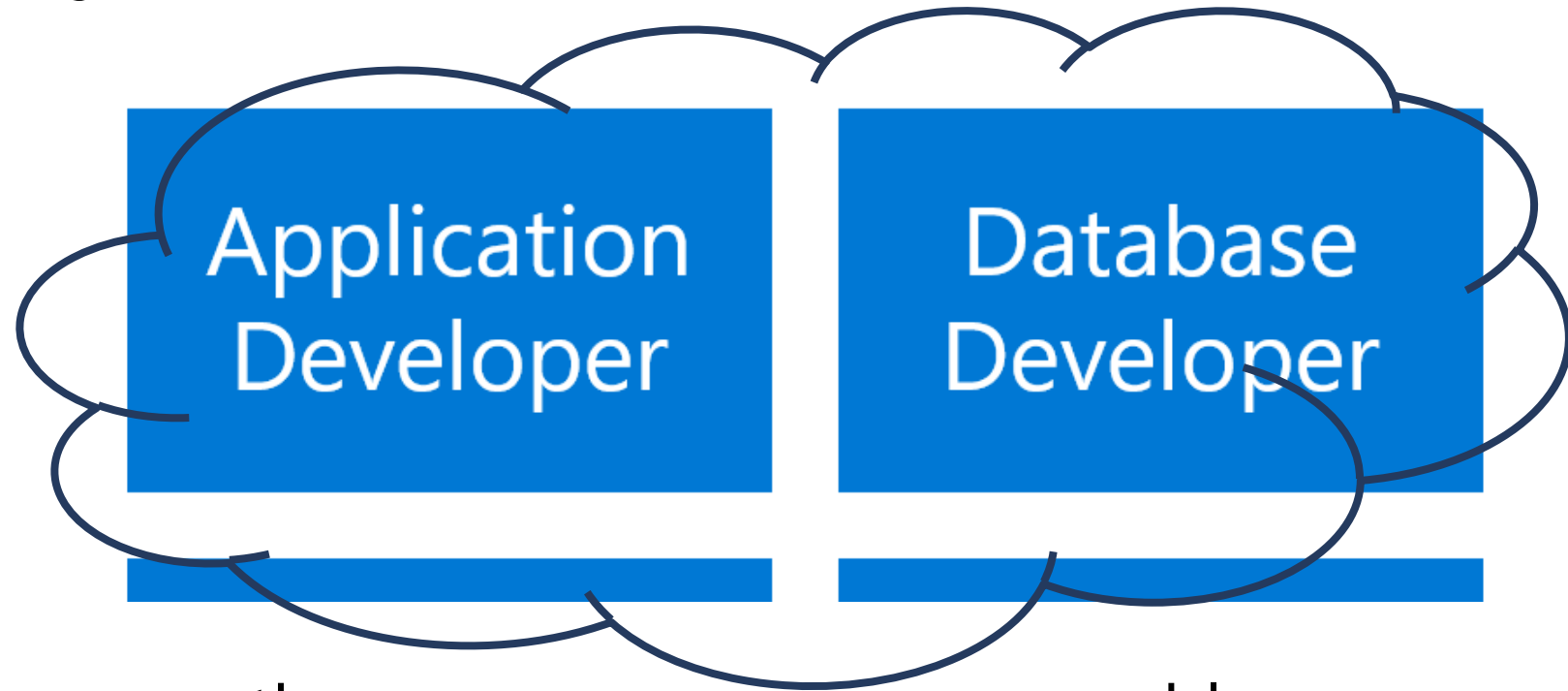
Database
Administrator

Business
Analyst

More specifically:

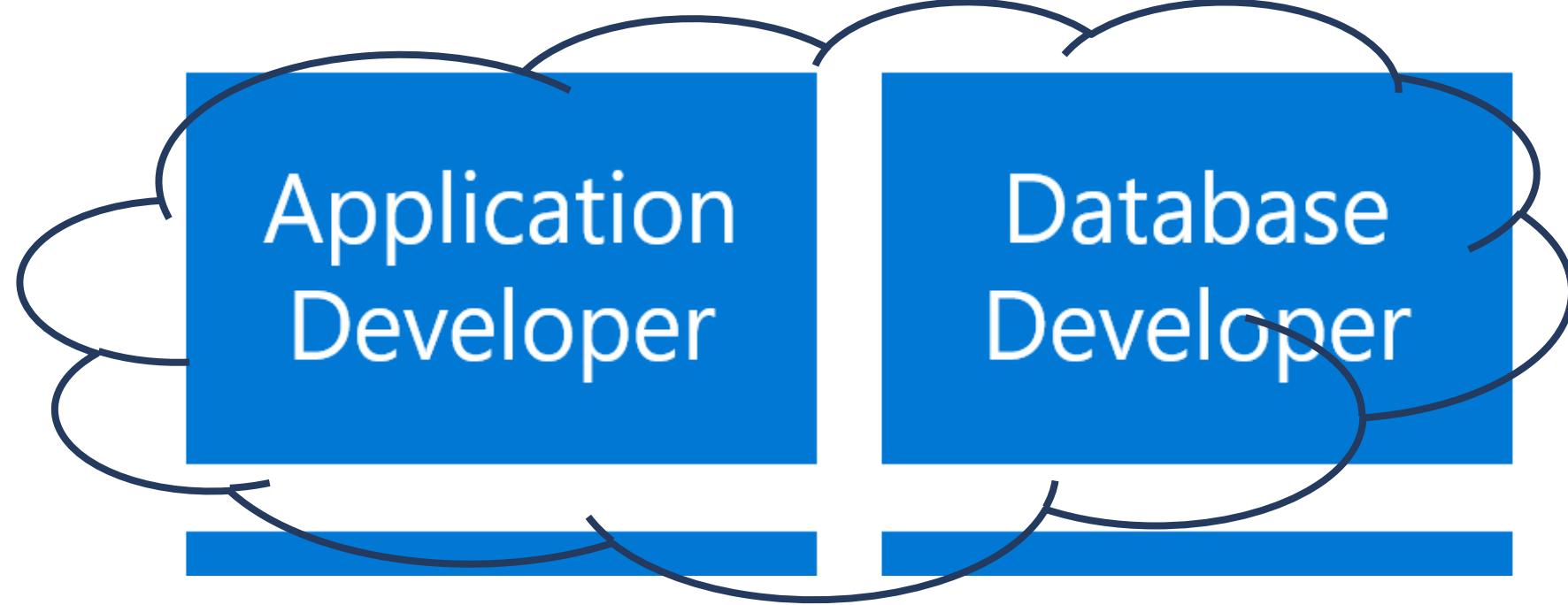
- Business decision maker
- Data analyst
- Data architect
- Data engineer
- Database developer
- Data officer
- Data scientist
- Big data resource manager
- Database administrator
- DevOps
- IT Pro
- App Developer
- Cloud Developer
- Solution Architect
- ...

The Developer's World



- More development languages than any person can reasonably use (C#, JavaScript, Python, Java, Go, Ruby, ...)
- Open source software everywhere like glitter
- Platforms (Windows, macOS, Linux, connected, disconnected, etc)
- Deploy to cloud services and/or in containers

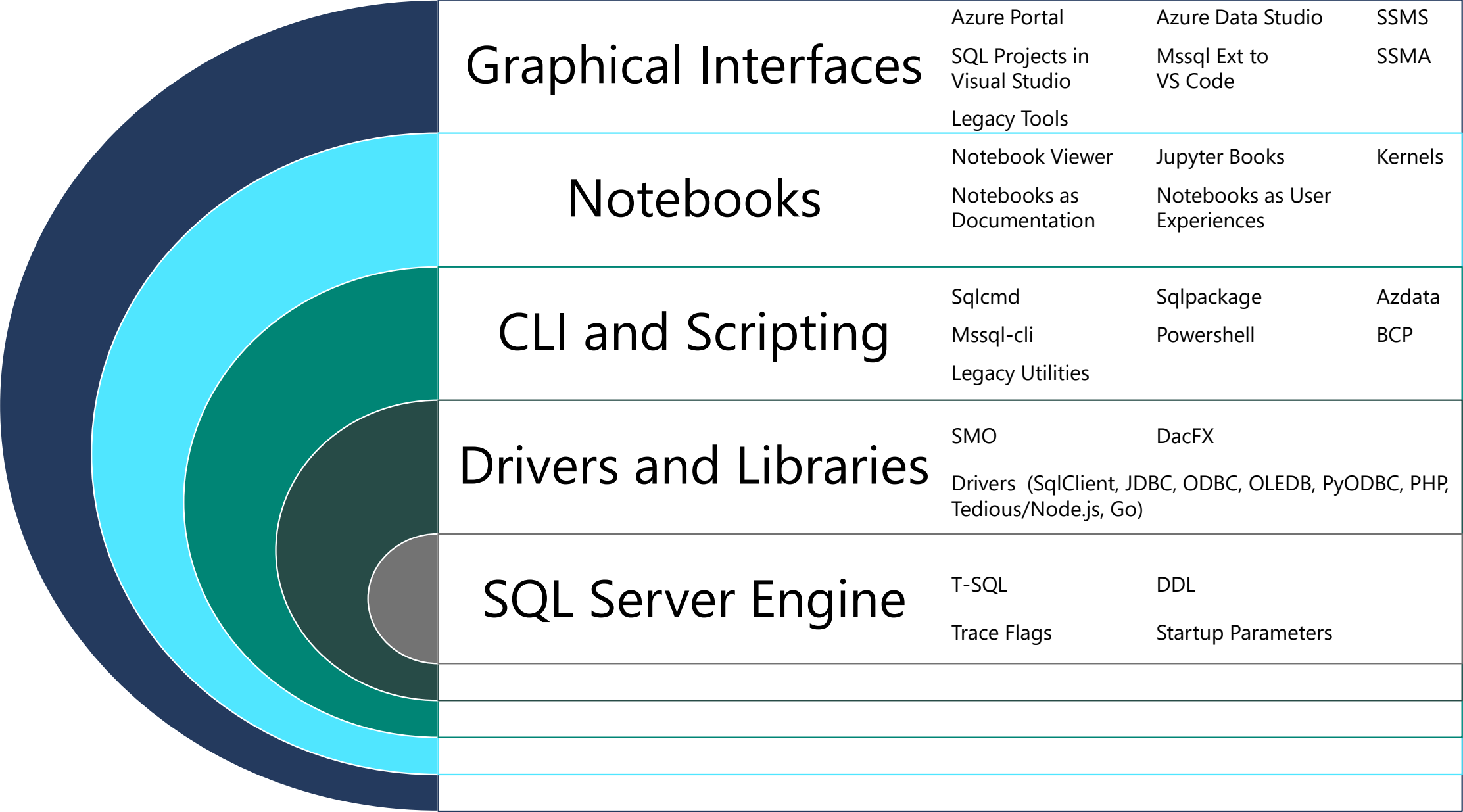
**Thankfully,
we have tools**



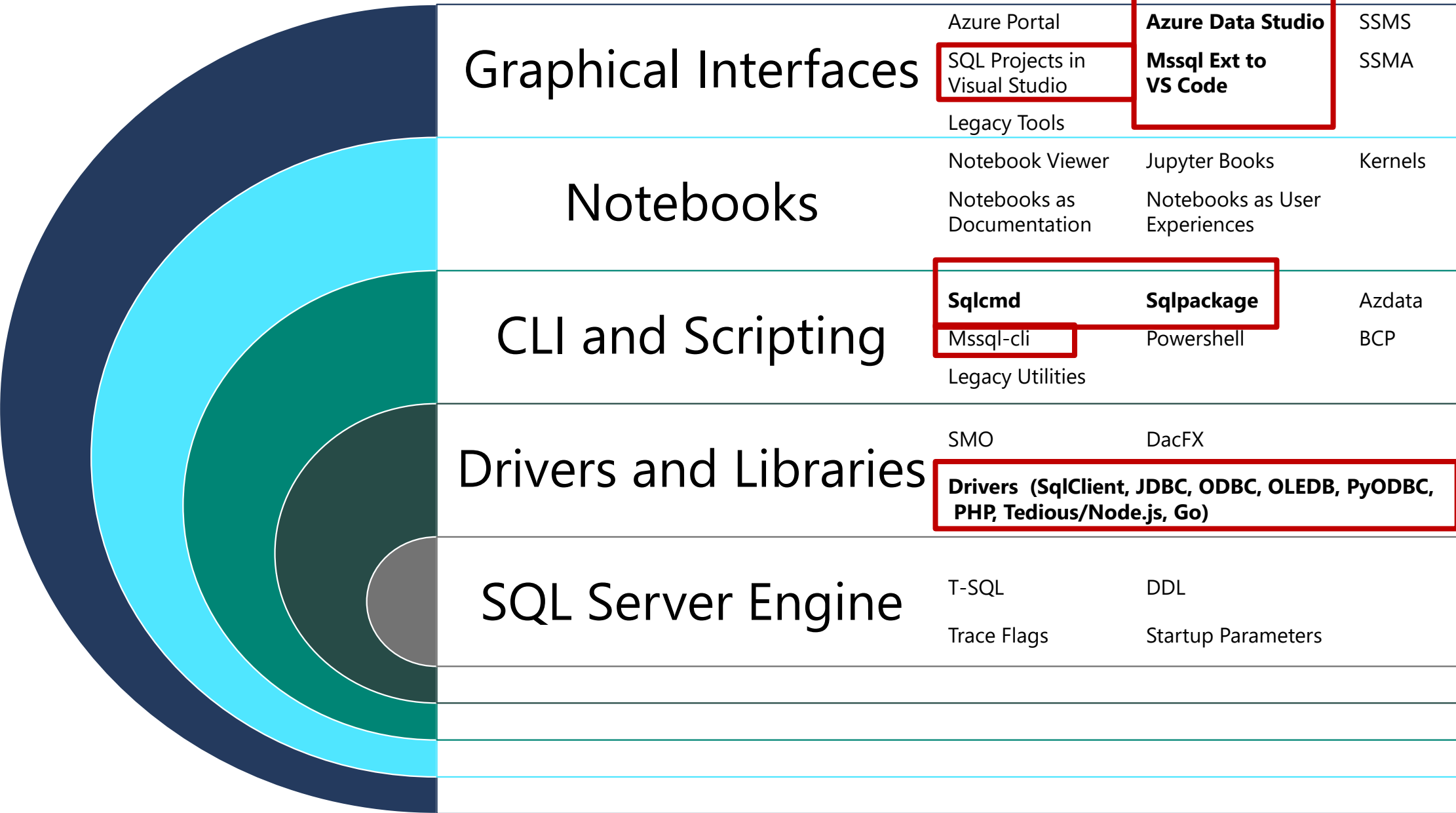
- Source control systems
- Application delivery pipelines
- Development environments
- Automation
- Frameworks (.NET Core, NodeJS, React, Angular, Rails, ...)



SQL Tools



SQL Tools



A familiar pathway for developers:



1. Write code in a tool of your choice
2. Compile, test locally, pull request review
3. Bundled or packaged code is deployed

Data-tier Applications

Database Schema
(tables, stored procs, etc)

Dacpac

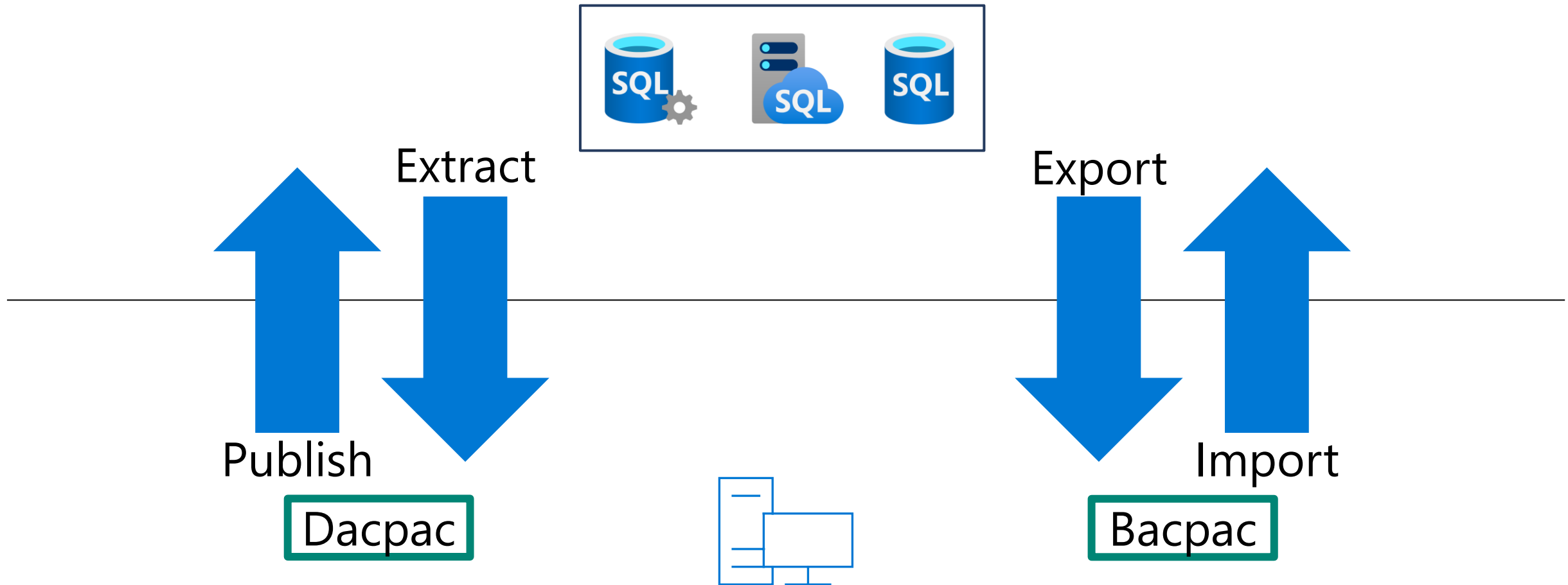
Database Schema (tables, stored procs, etc)

Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.
Rows and rows of data.

Bacpac

SqlPackage

Command line interface to DacFx framework



SqlPackage Command Syntax

```
SqlPackage /Action:{action}  
/{additionalParameter}:{parameterValue}
```

```
SqlPackage /Action:Extract /SourceServerName:localhost  
/SourceDatabaseName:AdventureWorksLT2019  
/TargetFile:c:\Users\drskwier\AdventureWorksLT.dacpac
```

Parameters have long and short form names

/Action:	/a:
/SourceServerName:	/ssn:
/TargetFile:	/tf:

Database Projects

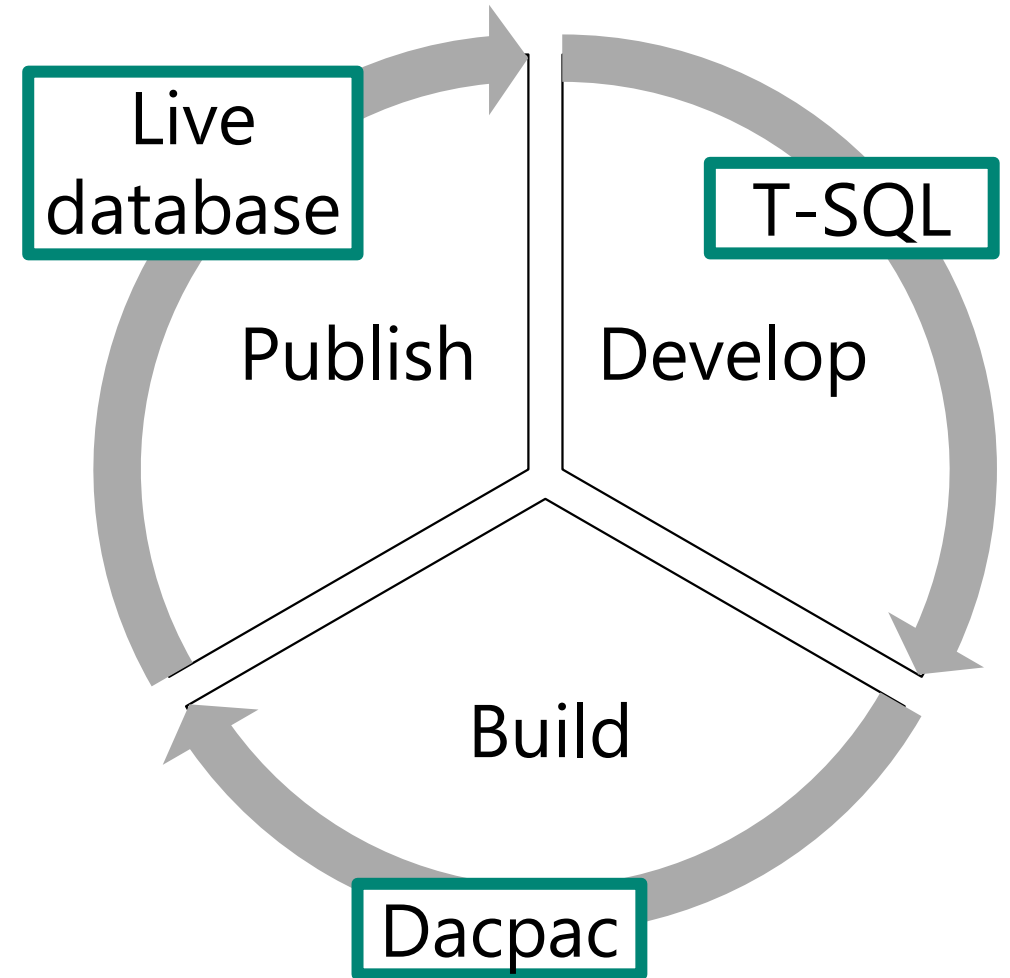
Declarative Database Development

Development of the desired end state

Also known as state based development
Tooling manages change and migration during deployment

Facilitates version control

Dacpac is the result from building a database project



SQL Project Development



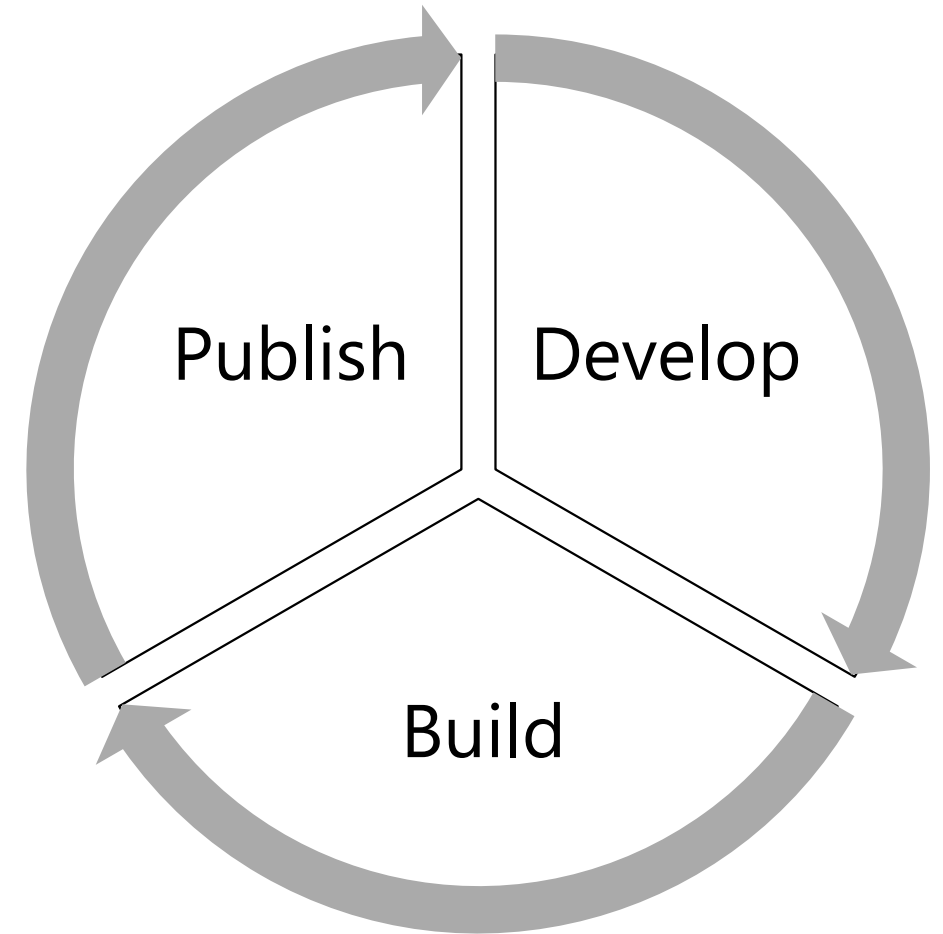
SQL Server Data Tools

Available in "Data Platform" workload of Visual Studio 2019



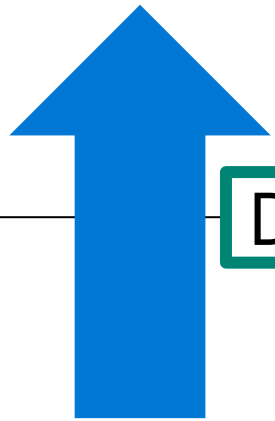
SQL Database Projects extension

Extension in Azure Data Studio
Leverages .NET Core SDK to bring SQL Projects to macOS and Linux



<https://aka.ms/azuredatstudio-sqlprojects>

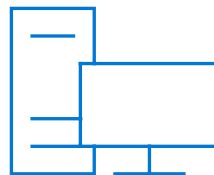
SqlPackage



Dacpac

Publish

Publish is an incremental operation
Updates the existing database
schema to match the dacpac



Database projects in Azure Data Studio

- Create project from existing database
- Update/change the project
- Generate scripts, review, and deploy
- Use post-deployment script to insert some data
- Use schema compare to view differences between production and test

File Edit View Help

Welcome - AdventureWorksLT (Workspace) - Azure Data Studio

CONNECTIONS

SERVERS

SQL Servers

- production
- test
 - Databases
 - System Databases
 - AdventureWorksLT
 - WideWorldImportersDW
 - Security
 - Server Objects
- Azure Infra

Azure

SQL SERVER BIG DATA CLUSTERS

Azure Data Studio

New Open file... Open folder...

Create a connection

Connect to a database instance through the connection dialog.

Run a query

Interact with data through a query editor.

Deploy a server

Create a new instance of a relational data service on the platform of your choice.

Create Project From Database

- Data-tier Application wizard
- Schema Compare
- First Responder Kit: Run sp_BlitzIndex
- First Responder Kit: Run sp_BlitzLock
- Import wizard Ctrl+I
- QE Boost: New Query Ctrl+N
- Generate Scripts...
- Properties

Show more

☒ Show welcome page on startup

Useful Links

Getting Started

Discover the capabilities offered by Azure Data Studio and learn how to make the most of them.

Choose SQL Language

test

> System Databases

> WideWorldImportersDW

> Server Objects

Resources

Name _____

[Show more](#)☒ Show welcome page on startup

Useful Links

[Getting Started](#) 

Discover the capabilities offered by Azure Data Studio and learn how to make the most of them.



Azure Data Studio

Open file...

Open folder...



Create a connection

Connect to a database instance through the connection dialog.



Create a notebook

Build a new notebook using a native notebook editor.

AdventureWorksLT > Tables > Customer.sql

connect Change Connection Select Database Explain Enable SQLCMD Export as Notebook

```
TABLE [SalesLT].[Customer] (
    CustomerID INT IDENTITY (1, 1) NOT NULL,
    NameStyle [dbo].[NameStyle] CONSTRAINT [DF_Customer_NameStyle] DEFAULT ((0)) NOT NULL,
```

```
] NVARCHAR (8) NULL,
Name] [dbo].[Name] NOT NULL,
```

```

eName]      [dbo].[Name]      NULL,
ame]        [dbo].[Name]      NOT NULL,

```

```

x] NVARCHAR (10) NULL,
anyName] NVARCHAR (128) NULL,

```

```
Person] NVARCHAR (256) NULL,
Address] NVARCHAR (50) NULL,
```

```
dsAcctId] INT NULL,
laryPhoneNumber dbo.Phone NULL, -
```

```
[dbo].[Phone] NULL,
```

```
wordSalt] VARCHAR (10) NOT NULL,  
id] UNIQUEIDENTIFIER CONSTRAINT
```

```

[ModifiedDate] DATETIME CONSTRAINT [DF_Customer_ModifiedDate] DEFAULT (getdate()) NOT NULL,
CONSTRAINT [PK_Customer_CustomerID] PRIMARY KEY CLUSTERED ([CustomerID] ASC).

```

```
CONSTRAINT [AK_Customer_rowguid] UNIQUE NONCLUSTERED ([rowguid] ASC)
```

```

NCLUSTERED INDEX [IX_Customer_EmailAddress]
ON [SalesLT].[Customer]([EmailAddress] ASC):

```

```
customer: { email: 'adam@cs.cmu.edu' }
```

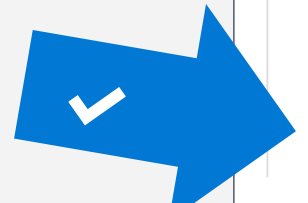
PROJECTS

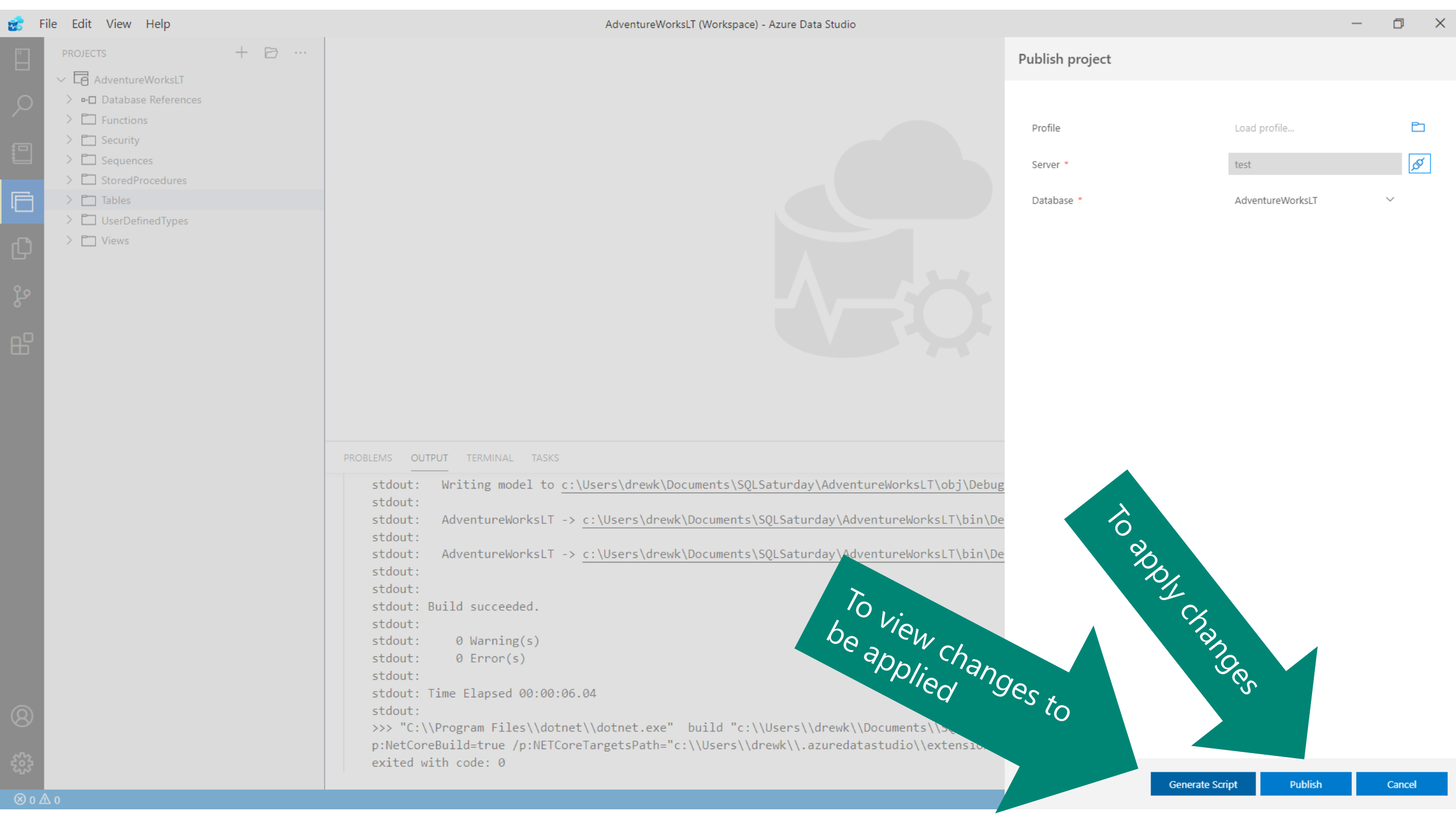
- AdventureWorksLT
 - Database References
 - Functions
 - Security
 - Sequences
 - StoredProcedures
 - Tables
 - Address.sql
 - BuildVersion.sql
 - Customer.sql
 - CustomerAddress.sql
 - ErrorLog.sql
 - Interests.sql
 - Product.sql
 - ProductCategory.sql
 - ProductDescription.sql
 - ProductModel.sql
 - ProductModelProductDescription.sql
 - RegularUser.sql
 - SalesOrderDetail.sql
 - SalesOrderHeader.sql
 - SampleInsertNull.sql
 - UserDefinedTypes
 - Views



Database Projects

```
stdout: Writing model to c:\Users\drewk\Documents\SQLSaturday\AdventureWorksLT\obj\Debug\Model.xml...
stdout:
stdout: AdventureWorksLT -> c:\Users\drewk\Documents\SQLSaturday\AdventureWorksLT\bin\Debug\AdventureWorksLT.dll
stdout:
stdout: AdventureWorksLT -> c:\Users\drewk\Documents\SQLSaturday\AdventureWorksLT\bin\Debug\AdventureWorksLT.dacpac
stdout:
stdout:
stdout: Build succeeded.
stdout:
stdout: 0 Warning(s)
stdout: 0 Error(s)
stdout:
stdout: Time Elapsed 00:00:06.04
stdout:
>>> "C:\Program Files\dotnet\dotnet.exe" build "c:\Users\drewk\Documents\SQLSaturday\AdventureWorksLT\AdventureWorksLT.sqlproj" /
p:NetCoreBuild=true /p:NETCoreTargetsPath="c:\Users\drewk\azuredatstudio\extensions\microsoft.sql-database-projects-0.5.1\BuildDirectory" ...
exited with code: 0
```





To view changes to be applied

To apply changes

PROJECTS

+

📁

...

AdventureWorksLT

>

Database References

>

Functions

>

Security

>

Sequences

>

StoredProcedures

>

Tables

>

UserDefinedTypes

>

Views

SQLQuery_1 - disconnected

×

▶ Run

❌ Cancel

🔌 Connect

🔄 Change Connection

Select Database

⌵

📖 Explain

🔧 Enable SQLCMD

📄 Export as Notebook

1

/*

2

Deployment script for AdventureWorksLT

3

4

This code was generated by a tool.

5

Changes to this file may cause incorrect behavior and will be lost if

6

the code is regenerated.

7

*/

8

9

GO

10

SET ANSI_NULLS, ANSI_PADDING, ANSI_WARNINGS, ARITHABORT, CONCAT_NULL_YIELDS_NULL, QUOTED_IDENTIFIER ON;

11

12

SET NUMERIC_ROUNDABORT OFF;

13

14

15

GO

16

:setvar DatabaseName "AdventureWorksLT"

17

:setvar DefaultFilePrefix "AdventureWorksLT"

18

:setvar DefaultDataPath ""

19

:setvar DefaultLogPath ""

20

21

GO

22

:on error exit

23

GO

24

/*

25

Detect SQLCMD mode and disable script execution if SQLCMD mode is not supported.

26

To re-enable the script after enabling SQLCMD mode, execute the following:

27

SET NOEXEC OFF;

28

*/

29

:setvar __IsSqlCmdEnabled "True"

30

GO

31

IF N'\$(__IsSqlCmdEnabled)' NOT LIKE N'True'

32

BEGIN

33

PRINT N'SQLCMD mode must be enabled to successfully execute this script.';

34

SET NOEXEC ON;

35

END

36

37










38

GO


39


USE [\$(DatabaseName)];


40





PROJECTS


+...


▼  AdventureWorksLT


>  Database


>  Functions


>  Security

>  Sequence

>  StoredProc

>  Tables

>  UserDefin

>  Views

Build

Publish

Schema Compare

Add Item...

Add Folder

Add Table

Add View

Add Stored Procedure

Add External Streaming Job

Add Script

Add Pre-Deployment Script

Add Post-Deployment Script


Change Target Platform


Edit .sqlproj File




Open Containing Folder




Remove Project



Show Servers 

New SQL File 

New Notebook  +  + 

Find in Files  +  + 

PROJECTS +

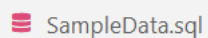
- AdventureWorksLT
 - Database References
 - Functions
 - Security
 - Sequences
 - StoredProcedures
 - Tables
 - UserDefinedTypes
 - Views
 - SampleData.sql

SampleData.sql - 192.16...LT (sa)

AdventureWorksLT > SampleData.sql

Run Cancel Disconnect Change Connection AdventureWorksLT Explain Enable SQLCMD Export as Notebook

```
1  -- This file contains SQL statements that will be executed after the build script.
2
3  INSERT INTO SALESLT.Customer
4  (NameStyle, FirstName, LastName, PasswordHash, PasswordSalt, ModifiedDate)
5  VALUES
6  (1, 'Drew', 'Skwiers-Koballa', '##HashBrowns##', 'VERYSALTY', getdate())
7
8  -- Insert more sample data, update permissions, or perform other operations
```



Remove Project


Ctrl + Shift + F


$\otimes 0 \Delta$


+

re \times 

AdventureWorksLT

>  Database References

>  Functions

>  Security

> Sequences

> StoredProcedures

>  Tables

> UserDefinedTypes

>  Views

 SampleData.sql

So

Ta

c:\Users\drewk\Documents\SQLSaturday\AdventureWorksLT\bin\Debug\AdventureWor...

...



192.168.6.62,61433.AdventureWorksLT

Type	Source Name	Include	Action	Target Name
SqlTable	SalesLT.Customer	<input checked="" type="checkbox"/>	Change	SalesLT.Customer
SqlInlineTableValuedFunction	dbo.ufnGetCustomerInformation	<input checked="" type="checkbox"/>	Change	dbo.ufnGetCustomerInformation

COMPARE DETAILS

```

1 CREATE TABLE [SalesLT].[Customer] (
2     [CustomerID] INT IDENTITY (1, 1) NOT NULL,
3+    [NameStyle] [dbo].[NameStyle] CONSTRAINT [DF_Customer_NameStyle] DEFAULT (
4     [Title] NVARCHAR (8) NULL,
5     [FirstName] [dbo].[Name] NOT NULL,
6     [MiddleName] [dbo].[Name] NULL,
7     [LastName] [dbo].[Name] NOT NULL,
8     [Suffix] NVARCHAR (10) NULL,
9     [CompanyName] NVARCHAR (128) NULL,
10    [SalesPerson] NVARCHAR (256) NULL,
11    [EmailAddress] NVARCHAR (50) NULL,
12    [RewardsAcctId] INT NULL,
13+    [SecondaryPhoneNumber] [dbo].[Phone] NULL,
14    [Phone] [dbo].[Phone] NULL,
15    [PasswordHash] VARCHAR (128) NOT NULL,
16    [PasswordSalt] VARCHAR (10) NOT NULL,
17+    [rowguid] UNIQUEIDENTIFIER CONSTRAINT [DF_Customer_rowguid] DEFAULT (newid
18+    [ModifiedDate] DATETIME CONSTRAINT [DF_Customer_ModifiedDate] DEFAULT (get

```

```

1 CREATE TABLE [SalesLT].[Customer] (
2     [CustomerID] INT IDENTITY (1, 1) NOT NULL,
3     [NameStyle] [dbo].[NameStyle] NOT NULL,
4     [Title] NVARCHAR (8) NULL,
5     [FirstName] [dbo].[Name] NOT NULL,
6     [MiddleName] [dbo].[Name] NULL,
7     [LastName] [dbo].[Name] NOT NULL,
8     [Suffix] NVARCHAR (10) NULL,
9     [CompanyName] NVARCHAR (128) NULL,
10    [SalesPerson] NVARCHAR (256) NULL,
11    [EmailAddress] NVARCHAR (50) NULL,
12    [RewardsAcctId] INT NULL,
13    [Phone] [dbo].[Phone] NULL,
14    [PasswordHash] VARCHAR (128) NOT NULL,
15    [PasswordSalt] VARCHAR (10) NOT NULL,
16    [rowguid] UNIQUEIDENTIFIER NOT NULL,
17    [ModifiedDate] DATETIME NOT NULL

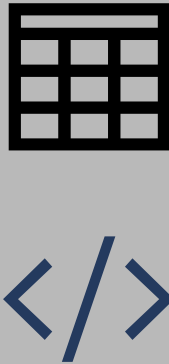
```

SQL Database Projects extension for Azure Data Studio

Open Existing
or Create
New Project



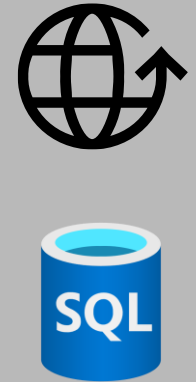
Edit Project



Build Project



Publish
Project



Development Environments

Development Environment

Tools needed to create an application (and connect it to a SQL server)

1. Code editor, preferably with IntelliSense and a debugger
2. SQL driver/library to connect our application code to a database
3. Command line utility for running queries and/or automating setup

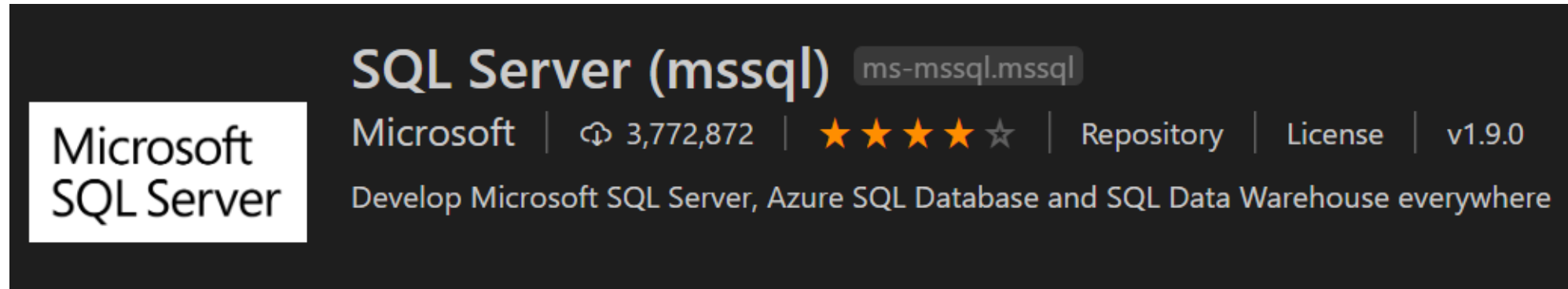
TODO: Create a .NET Core app with SQL Server

Visual Studio Code (aka VS Code)



mssql extension

Connect and query database servers – MS SQL and Azure SQL



Remote – Container development connects VS Code UI directly to container instance(s)

Microsoft.Data.SqlClient: Driver for .NET

.NET Core and .NET Framework Interoperability

Side-by-side replacement for System.Data.SqlClient

Consolidated support for .NET Core and .NET Framework

v2 Released June 2020

Adds Active Directory Integrated, Active Directory Interactive, and Active Directory Service Principal



Target Platform Support

- .NET Framework 4.6+ (Windows x86, Windows x64)
- .NET Core 2.1+ (Windows x86, Windows x64, Windows ARM64, Windows ARM, Linux, macOS)
- .NET Standard 2.0+ (Windows x86, Windows x64, Windows ARM64, Windows ARM, Linux, macOS)

sqlcmd

- Command line utility for connecting and executing queries (string or file)
- Installed with SQL Server and is available in the Azure Cloud Shell
- Available for Windows, macOS, Linux
- **Automate** pulling data from a SQL server
- Execute scripts in setting up a **test** instance

```
~$ /opt/mssql-tools/bin/sqlcmd -S servername -U username  
-P password -d databasename -i yourscripfile.sql
```

mssql-cli

- *Open source* command line utility for connecting and executing queries (string or file)
- Interactive and non-interactive modes
- Similar tools for other DB engines under same family
- Available for Windows, macOS, and Linux

```
~$ mssql-cli -S servername -U username  
    -P password -d databasename -i yourscripfile.sql
```

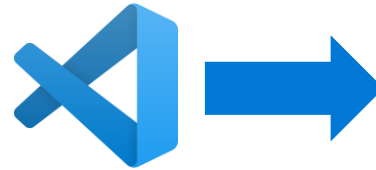
Create a .NET Core app with SQL Server

- Use remote development containers to work in a .NET core environment with a dedicated SQL Server instance
- Reuse Dacpac from earlier to create our schema and sample data

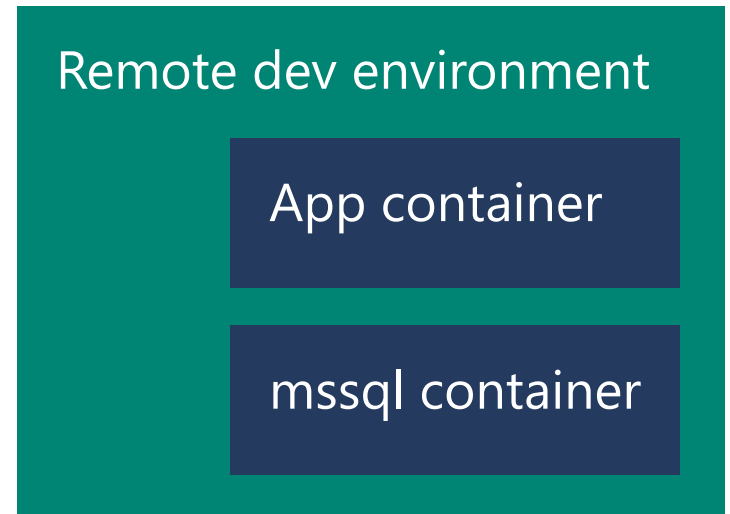
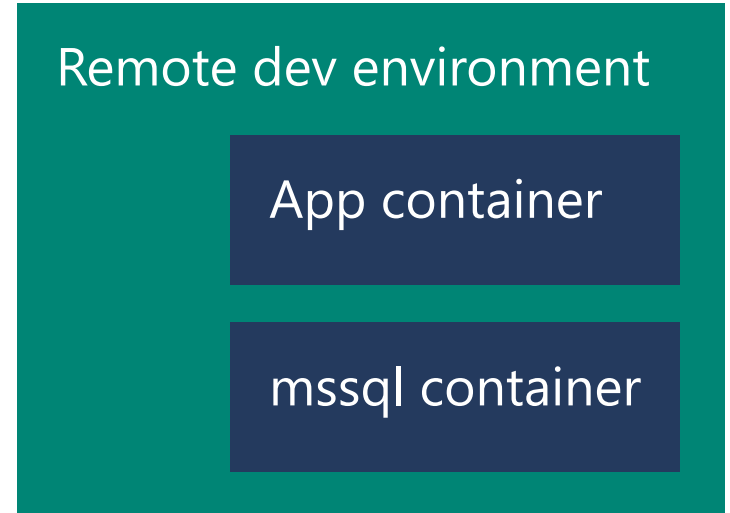
<https://github.com/microsoft/vscode-dev-containers/tree/master/containers/dotnet-mssql>

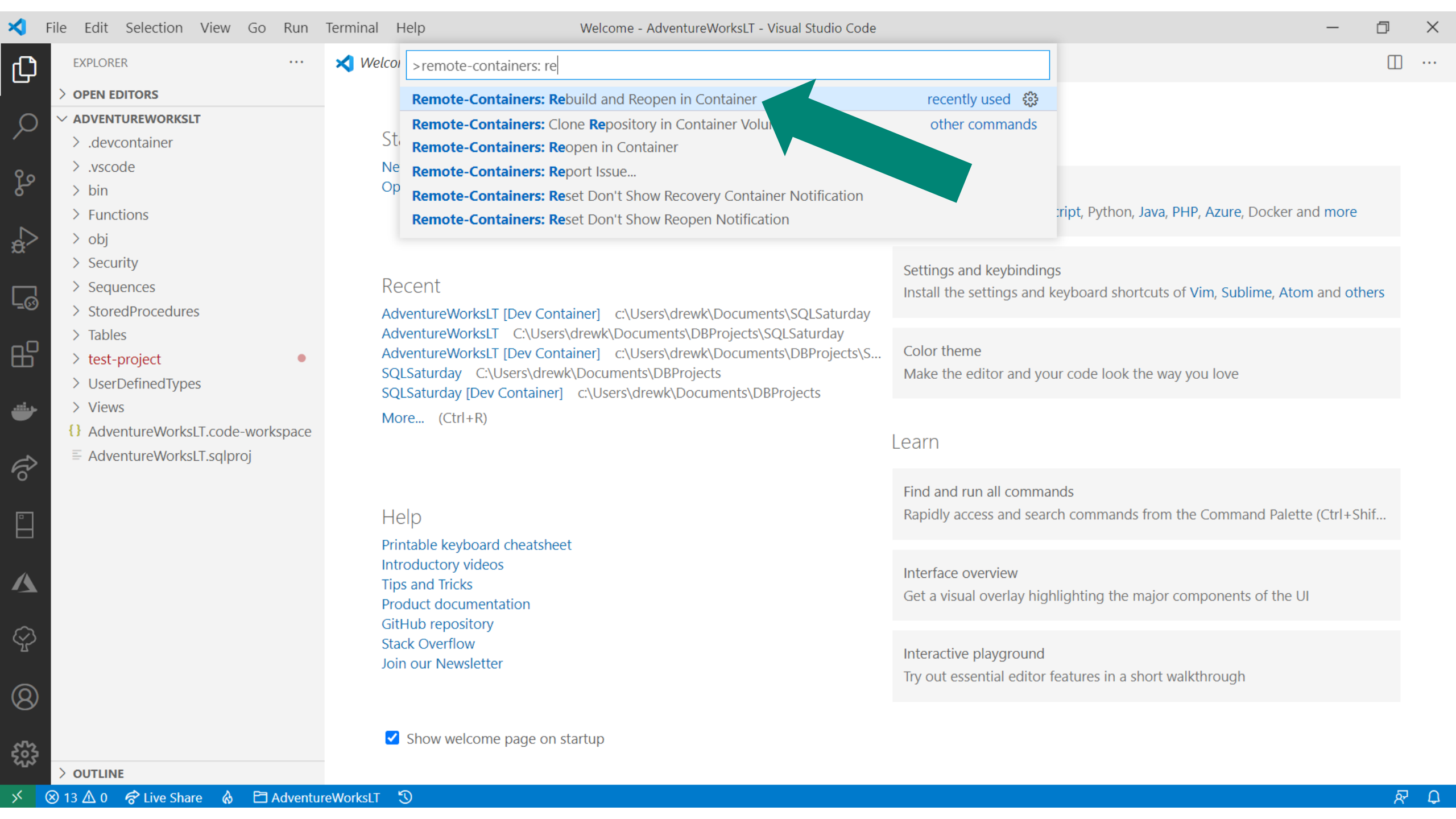
Remote Development in VS Code with mssql

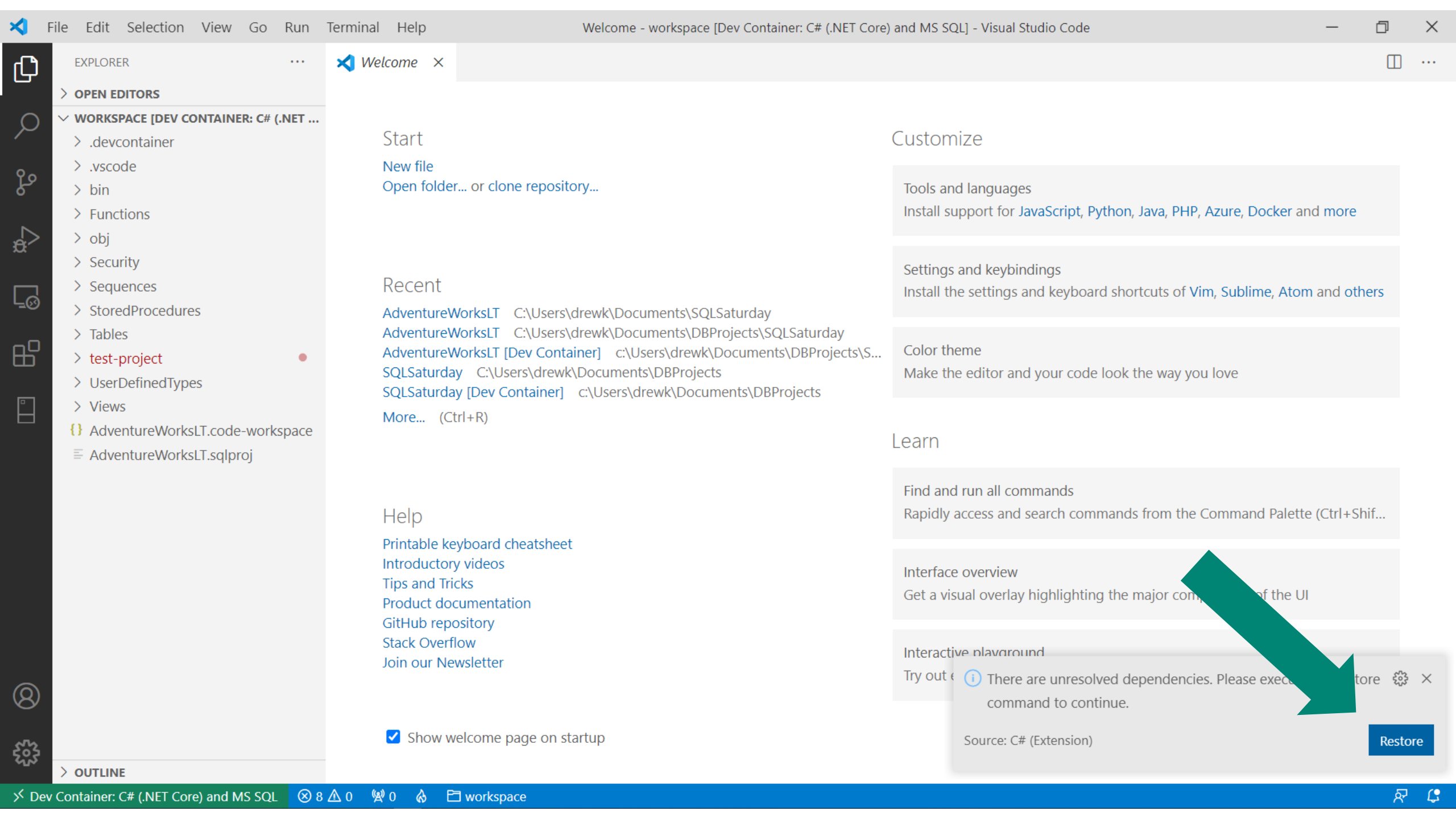
- Load mssql extension
- Connect to SQL instance from mssql container
- Use SqlPackage, sqlcmd, or other command line tools to deploy data-tier application components

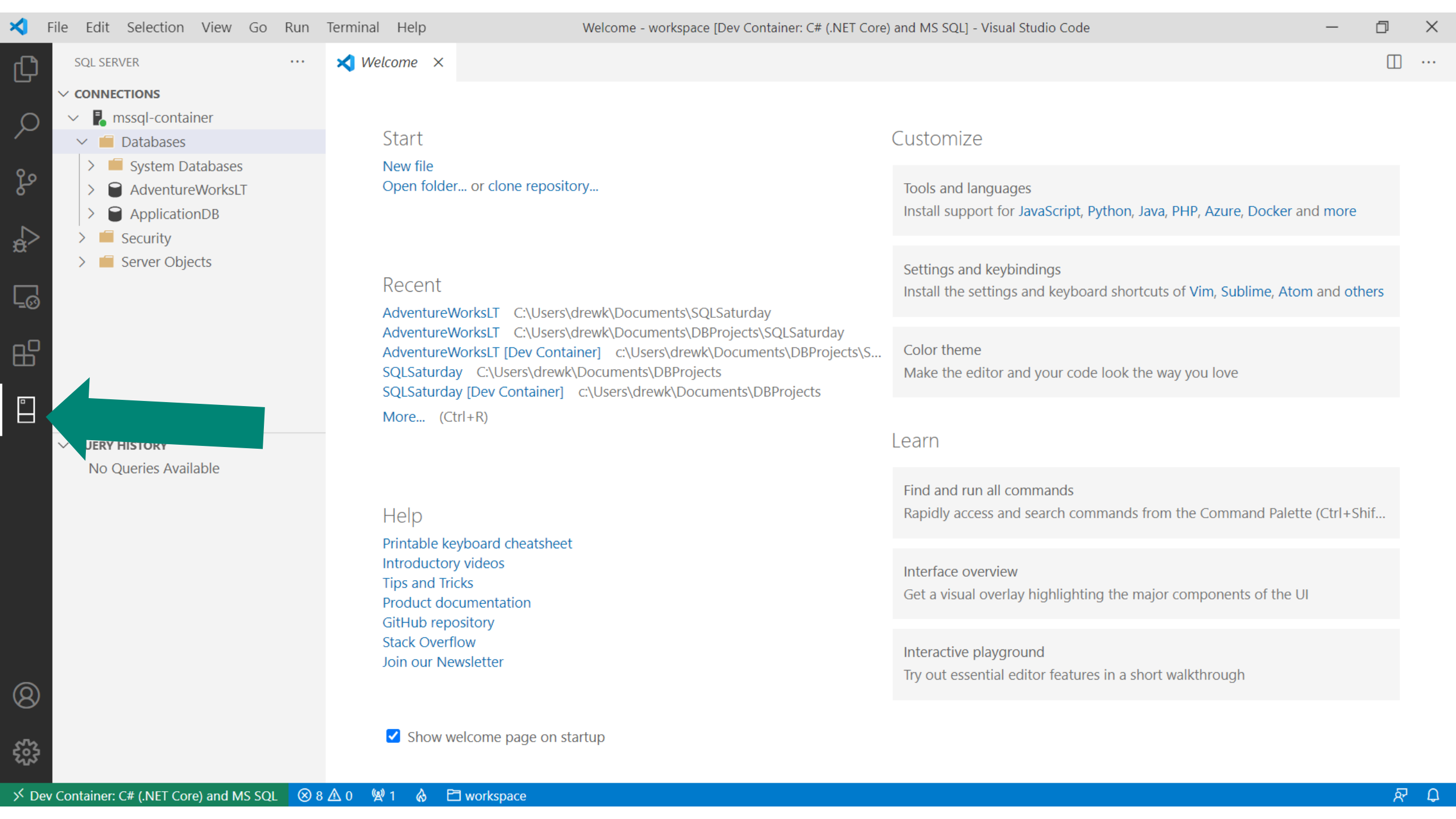


OR









SQL SERVER

CONNECTIONS

- mssql-container
 - Databases
 - System Databases
 - AdventureWorksLT
 - ApplicationDB
 - Security
 - Server Objects

QUERY HISTORY

No Queries Available

Welcome

Start

- New file
- Open folder... or clone repository...

Recent

- AdventureWorksLT C:\Users\drewk\Documents\SQLSaturday
- AdventureWorksLT C:\Users\drewk\Documents\DBProjects\SQLSaturday
- AdventureWorksLT [Dev Container] c:\Users\drewk\Documents\DBProjects\S...
- SQLSaturday C:\Users\drewk\Documents\DBProjects
- SQLSaturday [Dev Container] c:\Users\drewk\Documents\DBProjects
- More... (Ctrl+R)

Help

- Printable keyboard cheatsheet
- Introductory videos
- Tips and Tricks
- Product documentation
- GitHub repository
- Stack Overflow
- Join our Newsletter

☒ Show welcome page on startup

Customize

- Tools and languages
Install support for [JavaScript](#), [Python](#), [Java](#), [PHP](#), [Azure](#), [Docker](#) and more
- Settings and keybindings
Install the settings and keyboard shortcuts of [Vim](#), [Sublime](#), [Atom](#) and others
- Color theme
Make the editor and your code look the way you love

Learn

- Find and run all commands
Rapidly access and search commands from the Command Palette (Ctrl+Shif...
- Interface overview
Get a visual overlay highlighting the major components of the UI
- Interactive playground
Try out essential editor features in a short walkthrough



SQL SERVER ...

CONNECTIONS

- mssql-container
 - Databases
 - System Databases
 - AdventureWorksLT
 - Tables

QUERY HISTORY

- ✓ SELECT TOP (1000) [C...[SalesLT].[C...

```
1 SELECT TOP (1000) [CustomerID]
2      , [NameStyle]
3      , [Title]
4      , [FirstName]
5      , [MiddleName]
6      , [LastName]
7      , [Suffix1]
8      , [CompanyName]
9      , [SalesPersonID]
10     , [Email]
11     , [RewardPoints]
12     , [SecondaryPhoneNumbers]
13     , [Phone]
14     , [PasswordHash]
15     , [PasswordSalt]
16     , [rowguid]
17     , [ModifiedDate]
18 FROM [AdventureWorksLT].[SalesLT].[Customer]
```



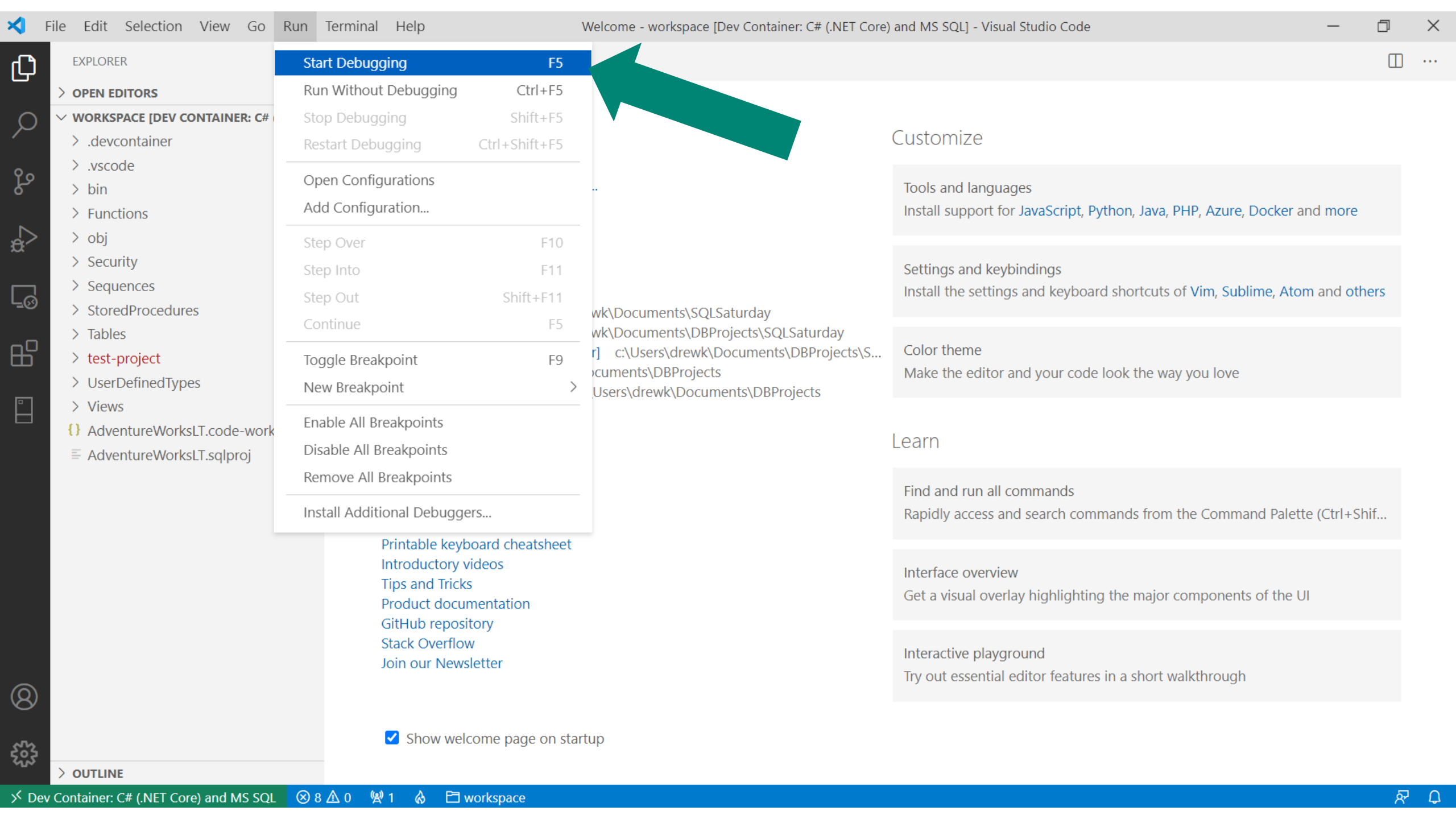
SalesLT.Customer X

RESULTS CTRL+ALT+R

	CustomerID	NameStyle	Title	FirstName
1	1	1	NULL	Drew

MESSAGES CTRL+ALT+Y

[9:26:28 PM] Started executing query at [Line 1](#)
(1 row affected)
Total execution time: 00:00:00.077



Start Debugging F5

Run Without Debugging Ctrl+F5

Stop Debugging Shift+F5

Restart Debugging Ctrl+Shift+F5

Open Configurations

Add Configuration...

Step Over F10

Step Into F11

Step Out Shift+F11

Continue F5

Toggle Breakpoint F9

New Breakpoint

Enable All Breakpoints

Disable All Breakpoints

Remove All Breakpoints

Install Additional Debuggers...

[Printable keyboard cheatsheet](#)
[Introductory videos](#)
[Tips and Tricks](#)
[Product documentation](#)
[GitHub repository](#)
[Stack Overflow](#)
[Join our Newsletter](#)

☒ Show welcome page on startup

Customize

Tools and languages

Install support for [JavaScript](#), [Python](#), [Java](#), [PHP](#), [Azure](#), [Docker](#) and more

Settings and keybindings

Install the settings and keyboard shortcuts of [Vim](#), [Sublime](#), [Atom](#) and others

Color theme

Make the editor and your code look the way you love

Learn

Find and run all commands

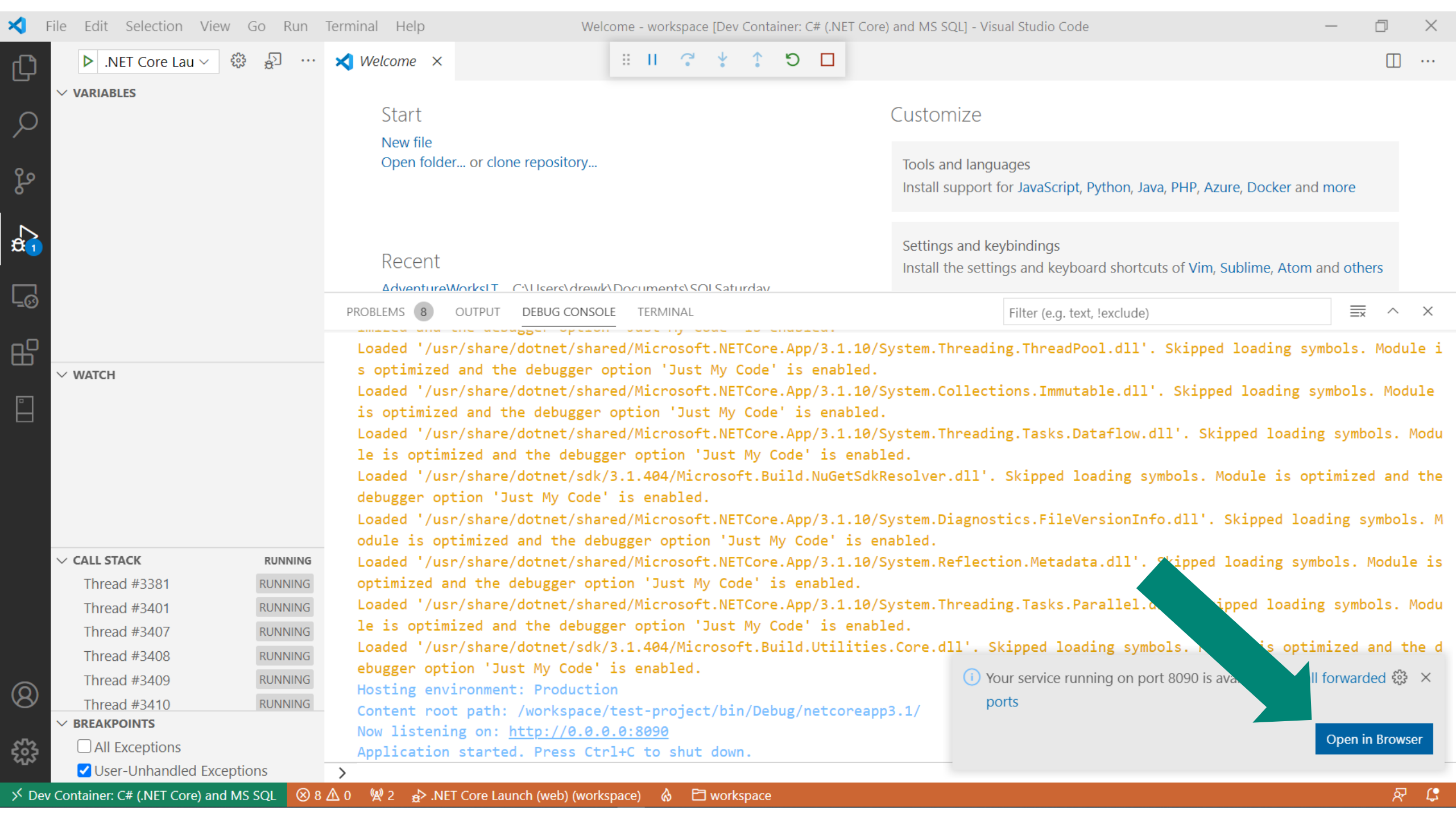
Rapidly access and search commands from the Command Palette (Ctrl+Shif...

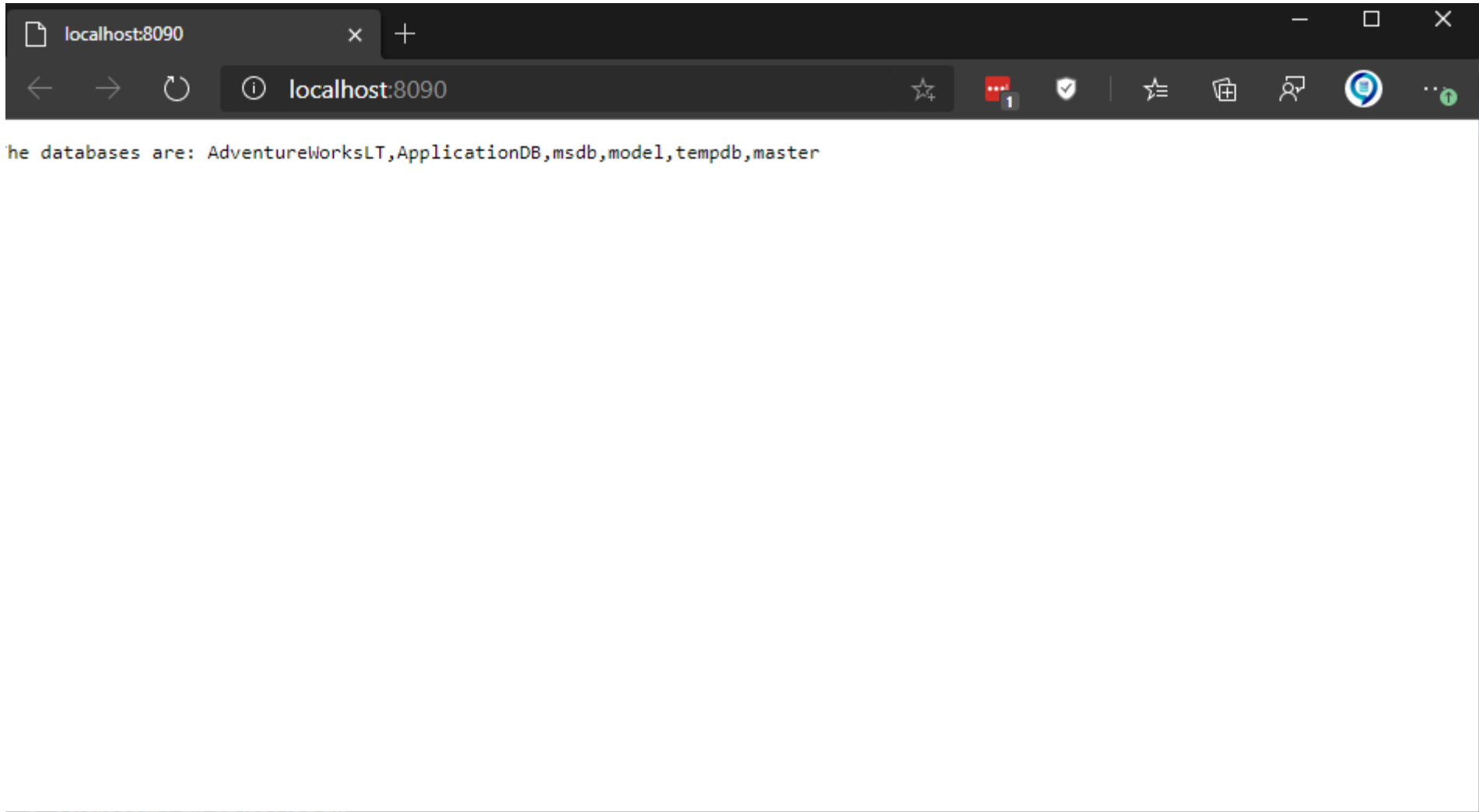
Interface overview

Get a visual overlay highlighting the major components of the UI

Interactive playground

Try out essential editor features in a short walkthrough





Resources for Remote Development

- **Getting started:**

<https://code.visualstudio.com/docs/remote/remote-overview>

- **Our example code:** <https://github.com/dzsquared/sql-development-tools>

- **.NET Core + mssql template:** <https://github.com/microsoft/vscode-dev-containers/tree/master/containers/dotnet-mssql>

Remote development in Codespaces

The screenshot shows the GitHub repository page for `dzsquared/sql-development-tools`. The repository is at the `main` branch with 1 branch and 0 tags. The file list shows various folders and files, all marked as "initial commit" and "2 minutes ago".

The "Code" button is highlighted, and a modal is open showing options to clone or open the repository. The "Open with Codespaces" option is selected, indicated by a blue arrow.

The "About" section shows the repository has no description, website, or topics provided. The "Languages" section shows the following distribution:

Language	Percentage
TSQL	74.2%
Shell	13.9%
C#	9.0%
Dockerfile	2.9%

The "No active codespaces" dialog is open, showing the message: "Codespaces let you develop on the cloud, as if you're running a local environment." The "+ New codespace" button is highlighted with a blue arrow.

Same dev environment from VS Code... in the browser.

The screenshot shows a web browser window displaying a SQL development interface. The address bar shows the URL `https://dzsquared-sql-development-tools-rv9c.github.io`. The interface is divided into several panels:

- Left Panel (SQL SERVER):** A tree view showing the database structure. Under "Databases", "AdventureWorksLT" is expanded, showing "Tables". The "SalesLT.Customer" table is selected.
- Center Panel (Query Editor):** Displays a SQL query:

```
1 SELECT TOP (1000) [CustomerID]
2     , [NameStyle]
3     , [Title]
4     , [FirstName]
5     , [MiddleName]
6     , [LastName]
7     , [Suffix]
8     , [CompanyName]
9     , [SalesPerson]
10    , [EmailAddress]
11    , [RewardsAcctId]
12    , [SecondaryPhoneNumber]
13    , [Phone]
14    , [PasswordHash]
15    , [PasswordSalt]
16    , [rowguid]
17    , [ModifiedDate]
18 FROM [AdventureWorksLT].[SalesLT].[Customer]
```
- Right Panel (RESULTS):** Displays the query results in a table. The first row is highlighted. The table has columns: CustomerID, NameStyle, Title, FirstName, MiddleName, and LastName.
- Bottom Panel (MESSAGES):** Shows execution details:

```
[9:48:12 PM] Started executing query at Line 1
(1 row affected)
Total execution time: 00:00:00.058
```

The status bar at the bottom indicates the current file is "main*", the editor is in "UTF-8" encoding, and the query is executed against "MSSQL localhost,1433 : AdventureWorksLT : sa".

CustomerID	NameStyle	Title	FirstName	MiddleName	LastName
1	1	NULL	Drew	NULL	Skwiers-Koball

Recap

- Database development can leverage a *similar* development cycle as general applications
- Database projects extension in Azure Data Studio (and SSDT) enables developing databases with data-tier applications
- Container-based development can place a reusable data layer with application code



Drew Skwiers-Koballa

Program Manager

drew.skwierskoballa@microsoft.com