# Visual Studio SQL Projects

# Intro

- Juan Tarquino
- Software Developer with about 20 years of experience
- https://www.linkedin.com/in/juantarquino/
- Sample repo for this session:
  - https://github.com/TheJuanitoLearnsShow/PodcastsManager

# Today’s talk

- Level: mostly beginner, with some deeper dives
- Outline
  - Why SQL Projects
  - How to Create SQL Projects
  - How to Deploy (including CI/CD integration)
  - How to do Automated Testing
- Conference sessions are great! They allow interactions with speakers and other attendees.

# Once upon a time in Pizzas’R’Us™…

- Pizzas’R’Us™ is a small but profitable company, trying to be as nimble as possible, providing pizza services to restaurants and super-markets.

# Once upon a time in Pizzas'R'Us™...

- Alexandria
  - Junior Dev always seeking the most efficient and safe way to accomplish the business goals
- Michael J
  - Alexandria's co-worker, also a junior dev, working hard with Alexandria, wearing multiple hats at the company.
- David
  - Alexandria's and Michael's boss. A rocker by night, a manager by day.

# The Business Goal

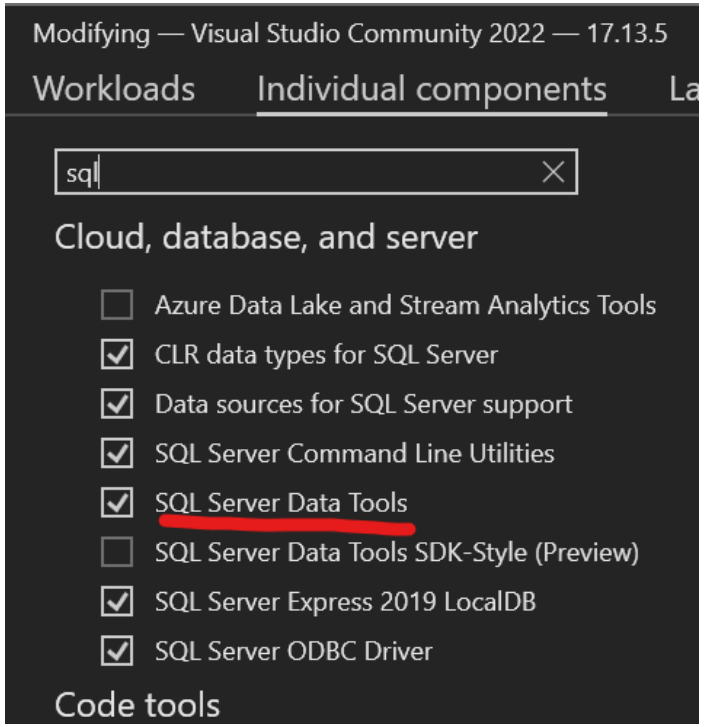
- Develop an app to help the Business Units manage and publish the weekly podcasts. The podcasts are mostly interviews with Pizza Chefs all over the world.
- The app must maintain state in a database, MS SQL Server is what the company uses because it already paid a hefty license fee for it due to other 3rd party software that uses it.
- The data needs to be able to be queried from other tools/apps to help the BU audit and ask questions about how well the podcast is progressing.

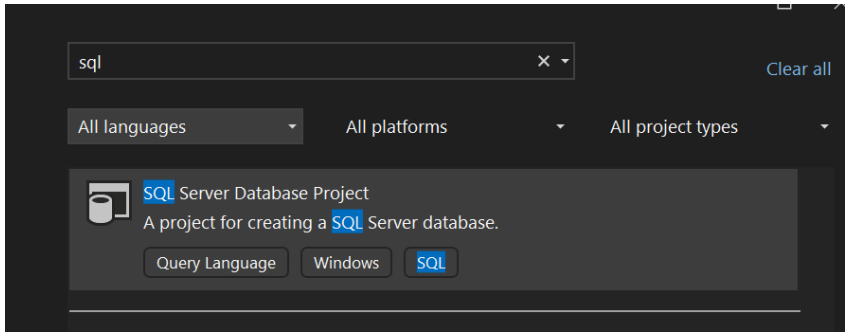
# Initial attempts

- SSMS
  - Need source control.
- .NET ORM, (e.g.: Entity Framework)
  - Cannot define views and Stored Procedures in C#, still relies on SQL code.
  - Migration mechanism to update the actual db does not look right (not declarative, delta operations vs desired state).

# Visual Studio SQL Projects to the rescue

- Two types
  - “Legacy” MSBuild based proj file format
  - SDK-style proj file (Preview)
- Similar to other VS projects
  - IntelliSense and CoPilot
  - Find references
- Folder organization
- Source control
- “New Item” menu, examples:
  - Roles
  - SQL types



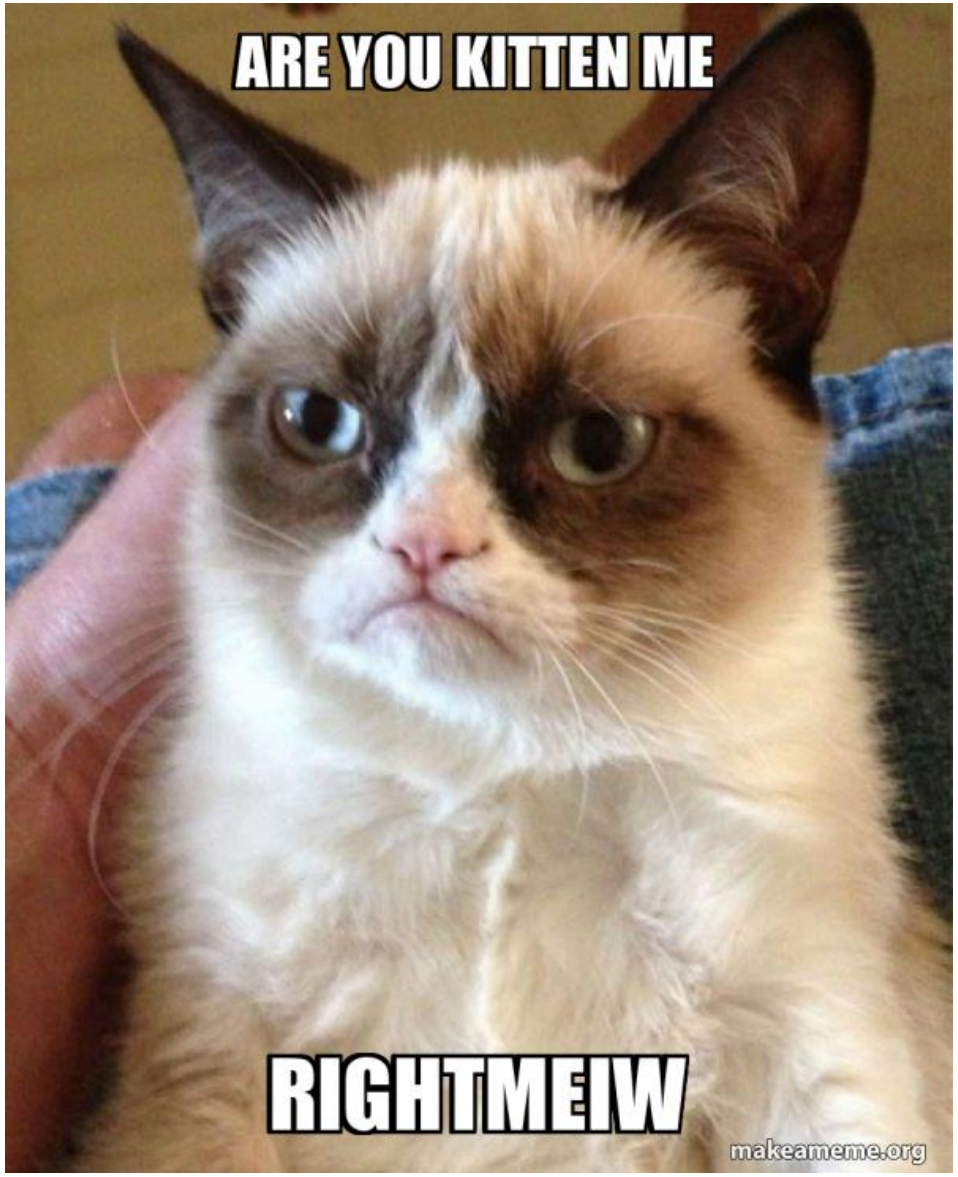



# Modeling the Database

- Match SQL objects to Business concepts
- Events/State
- Outbox pattern

# Deployment

- Right-click publish
- What happens in a deployment
  - “Compilation” to DACPAC, catches syntax and reference errors.
  - “Publishing” of sql objects to destination
- CI/CD
- Credentials
- JIT Access suggestions


ARE YOU KITTEN ME
RIGHTMEIW
makeameme.org

# What happens in a Deployment?

1. VS compares the objects defined in the SQL project vs what is on the server.
2. Anything that is in the SQL project but not in the server is scripted as an "add" (e.g. CREATE VIEW).
3. Anything that is in the SQL project and the SQL server, but differs in definition (e.g. a view has a different order of columns or an extra column), gets changed. Sometimes via an alter statement and sometimes via a drop/create. The deployment engine decides that for you.
    1. It is important to note that, by default, the deployment engine will NOT do any changes that have the possibility of losing data in the server. For example, a column being removed from a table or a column changing to a smaller type.
    2. Removing a column from a View or Stored Proc does not constitute data loss. You might be thinking, if the view is missing the column would not that affect anything else that depends on the column? The answer is yes and there are two ways tio prevent issues:
        1. Any sql object that refers to that column within the db, will fail to compile in the project so you would know that you are referencing something that is now missing so you would not even be able to get to the deployment step because compilation would fail.
        2. If your external code references that missing column, then you would do the same thing you would do if you .net public API has a breaking change. We will talk more about it later when we talk of the db as a service.
4. Anything that is not in this SQL project but it is in the db, will NOT be touched by default.
5. Database settings (such as Read Committed/Snapshot)are applied by default.

# Command line publishing

msbuild "nameofsqlproject.sqlproj" /p:Configuration=Release

dotnet tool install -g microsoft.sqlpackage

SqlPackage /Action:Publish /SourceFile:".\bin\release\nameofsqlproject.dacpac" /TargetConnectionString:"ConnectionString to the db you want to publish to” /p:ScriptDatabaseCompatibility=true

**For example**:

msbuild "PodcastsManagerDb.sqlproj" /p:Configuration=Release

dotnet tool install -g microsoft.sqlpackage

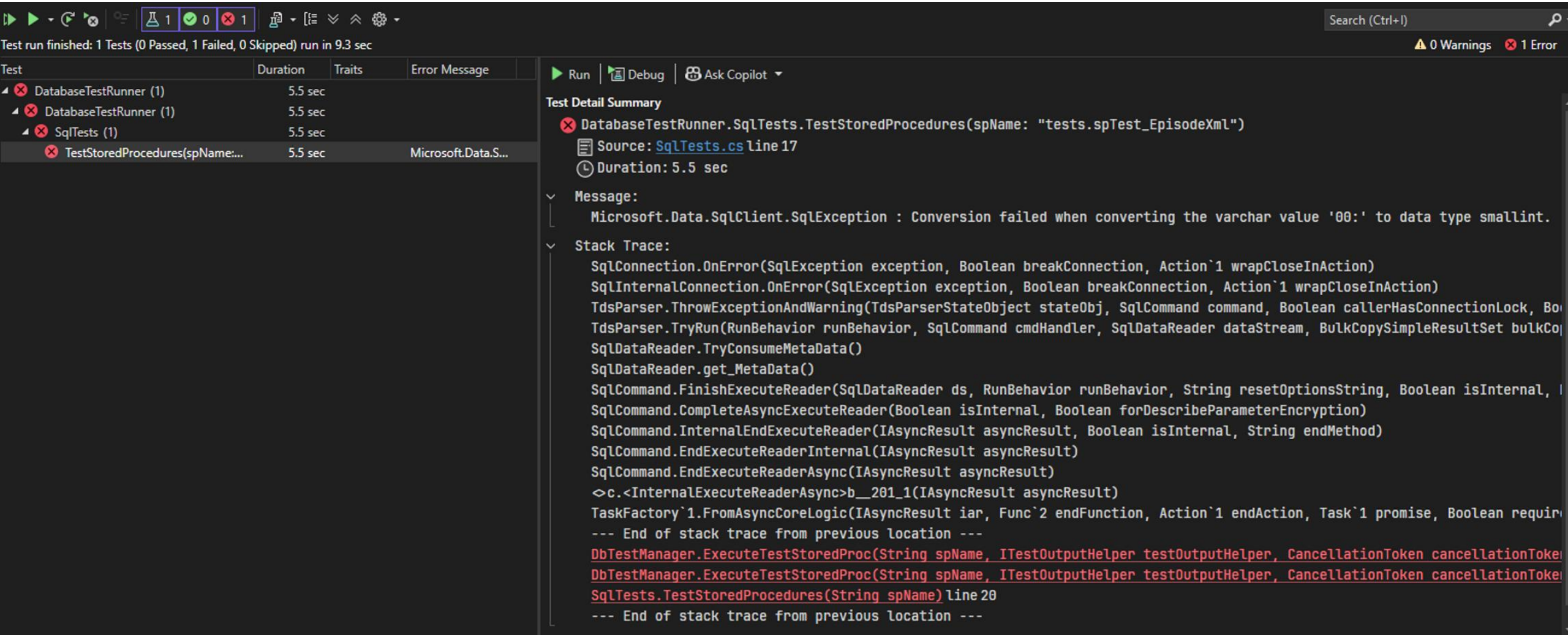
SqlPackage /Action:Publish /SourceFile:".\bin\release\PodcastsManagerDb.dacpac" /TargetConnectionString:"Data Source=.\sqlExpress;Database=PodcastsManager;Integrated Security=True;" /p:DropObjectsNotInSource=True /p:ScriptDatabaseCompatibility=True /p:BlockOnPossibleDataLoss=False

# Automated testing

- Many ways to implement automated testing
  - Custom tests from .NET.
  - SQL-Specific Unit Test frameworks (e.g.: tUnit).
  - Custom, generic, convention-based automated testing using xUnit template.
- Demo of Reusable xUnit project
- CI/CD integration

# Custom generic xUnit Project

- Sample in repo
  - https://github.com/TheJuanitoLearnsShow/PodcastsManager/tree/main/src/PodcastsManager/7-Tests/DatabaseTestRunner
- Uses a convention approach to discovering test stored procedures from the SQL db
- Each test stored procedure follows the arrange, act, assert pattern.
  - Uses transaction that are rolled back after the test
  - The results produced by the SP will be examined by the xUnit code to determine if the follow the PASS/FAIL convention
- The C# code is minimal and easily ported to another solution


1
0
1
Search (Ctrl+I)
Test run finished: 1 Tests (0 Passed, 1 Failed, 0 Skipped) run in 9.3 sec
0 Warnings
1 Error
Test
Duration
Traits
Error Message
DatabaseTestRunner (1)
5.5 sec
DatabaseTestRunner (1)
5.5 sec
SqlTests (1)
5.5 sec
TestStoredProcedures(spName:...
5.5 sec
Microsoft.Data.S...
Run
Debug
Ask Copilot
Test Detail Summary
DatabaseTestRunner.SqlTests.TestStoredProcedures(spName: "tests.spTest_EpisodeXml")
Source: SqlTests.cs line 17
Duration: 5.5 sec
Message:
Microsoft.Data.SqlClient.SqlException : Conversion failed when converting the varchar value '00:' to data type smallint.
Stack Trace:
SqlConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
SqlInternalConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj, SqlCommand command, Boolean callerHasConnectionLock, Bo
TdsParser.TryRun(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader dataStream, BulkCopySimpleResultSet bulkCo
SqlDataReader.TryConsumeMetaData()
SqlDataReader.get_MetaData()
SqlCommand.FinishExecuteReader(SqlDataReader ds, RunBehavior runBehavior, String resetOptionsString, Boolean isInternal,
SqlCommand.CompleteAsyncExecuteReader(Boolean isInternal, Boolean forDescribeParameterEncryption)
SqlCommand.InternalEndExecuteReader(IAsyncResult asyncResult, Boolean isInternal, String endMethod)
SqlCommand.EndExecuteReaderInternal(IAsyncResult asyncResult)
SqlCommand.EndExecuteReaderAsync(IAsyncResult asyncResult)
<>c.<InternalExecuteReaderAsync>b__201_1(IAsyncResult asyncResult)
TaskFactory`1.FromAsyncCoreLogic(IAsyncResult iar, Func`2 endFunction, Action`1 endAction, Task`1 promise, Boolean requir
--- End of stack trace from previous location ---
DbTestManager.ExecuteTestStoredProc(String spName, ITestOutputHelper testOutputHelper, CancellationToken cancellationToke
DbTestManager.ExecuteTestStoredProc(String spName, ITestOutputHelper testOutputHelper, CancellationToken cancellationToke
SqlTests.TestStoredProcedures(String spName) line 20
--- End of stack trace from previous location ---

# Additional Resources

### Formatter

Poor Man's T-Sql Formatter is a VS extension that allows you to format sql code in Visual Studio. It is my preferred formatter for T-SQL

https://marketplace.visualstudio.com/items?itemName=mick9956.PoorMansTSqlFormatter

### Resources

- Get started with SQL database projects - SQL Server | Microsoft Learn ( https://learn.microsoft.com/en-us/sql/tools/sql-database-projects/get-started )
- 9 Tips for Faster SQL Server Applications - Brent Ozar Unlimited® (https://www.brentozar.com/archive/2019/05/9-tips-for-faster-sql-server-applications )
- https://jptarqu.blogspot.com/2023/04/add-sql-server-database-projects-to-ci.html
- SqlPackage Publish - SQL Server | Microsoft Learn (https://learn.microsoft.com/en-us/sql/tools/sqlpackage/sqlpackage-publish )

# SQL Projects for the Win!!!

- Declarative way to define your databases.
- Source control friendly.
- Automated deployments
- Automated tests
- Model your databases as close to the business problem as possible
  - Your database is a service not just a repository of CRUD tables.

# Questions, Questions, Questions???

- Best part of sessions like this one ☺

Thank you! Win something at the raffle!