# IST769 Homework Submission #3

## Basic Information

Your Name: Thulasi Ram Ruppa Krishnan   
Your SUID: truppakr  
Your Email: truppakr@syr.edu  
Date Due: 460746269   
Homework #: HW3

## Instructions

For each answer, please include your answer as text, and any screenshot(s) which demonstrate your answer was executed. Most importantly, make sure to include evidence your answer is correct. This will most likely be a screenshot. If you had issues, problems, or had to make assumptions include them in your answer.

## Answers:

Complete each of the following exercises. If you are unsure how to accomplish the task, please consult the coursework videos where there are explanations and demos.

Complete each of the following exercises. If you are unsure how to accomplish the task, please consult the coursework videos where there are explanations and demos.

In this example we will create a series of database objects in the **Demo** database to track Basketball player shooting statistics across a 5-minute period.

1. In the demo database, create two tables:
   1. The first table **players** should have columns player id (int pk), player name (varchar), shots attempted (int) shots made (int)
   2. The second table **shots** should have columns shot id (int pk), player id (int fk to players), clock time (datetime) shot made (bit)
   3. Add two players to the players table. Mary and Sue initialize the players with 0 shots attempted and made.

**Solution:**

USE [demo]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

DROP TABLE IF EXISTS [dbo].[shots]

DROP TABLE IF EXISTS [dbo].[players]

GO

CREATE TABLE [dbo].[players](

[player\_id] [int] IDENTITY NOT NULL ,

[player\_name] [varchar](max) NULL,

[shots\_attempted] [int] NOT NULL,

[shots\_made] [int] NOT NULL,

CONSTRAINT [CK\_shots\_attempted\_not\_less\_than\_zero] CHECK([shots\_attempted]>=0),

CONSTRAINT [CK\_shots\_made\_not\_less\_than\_zero] CHECK([shots\_made]>=0),

CONSTRAINT [PK\_players] PRIMARY KEY CLUSTERED

(

[player\_id] DESC

)

) ON [PRIMARY]

GO

CREATE TABLE [dbo].[shots](

[shot\_id] [int] IDENTITY NOT NULL ,

[player\_id] [int] NOT NULL,

[clock\_time] [datetime] NOT NULL,

[shot\_made] [bit] NOT NULL,

CONSTRAINT [PK\_shots] PRIMARY KEY CLUSTERED

(

[shot\_id] DESC

)

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[shots] WITH CHECK ADD CONSTRAINT [FK\_shots\_players] FOREIGN KEY([player\_id])

REFERENCES [dbo].[players] ([player\_id])

GO

ALTER TABLE [dbo].[shots] CHECK CONSTRAINT [FK\_shots\_players]

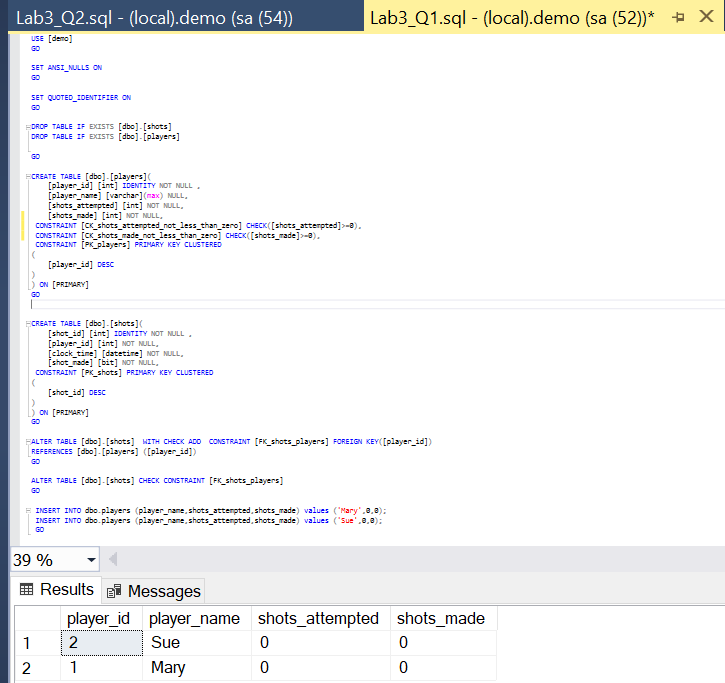
GO

INSERT INTO dbo.players (player\_name,shots\_attempted,shots\_made) values ('Mary',0,0);

INSERT INTO dbo.players (player\_name,shots\_attempted,shots\_made) values ('Sue',0,0);

GO

SELECT \* FROM dbo.players;



1. Write transaction safe code as a stored procedure which when given a player id, clock time, and whether the shot was made (bit value) will add the record to the **shots** table and update the player record in the **players** table. For example, If Mary takes a shot and makes it, then misses the next one, there would be two records in the **shots** table and her row in the **players** table should have 2 attempt and 1 shot made. Execute the stored procedure to demonstrate the transaction is ACID compliant.

**Solution:**

IF EXISTS(SELECT \* FROM sys.objects WHERE NAME ='p\_write\_shot')

BEGIN

DROP PROCEDURE dbo.p\_write\_shot;

END

GO

CREATE PROCEDURE p\_write\_shot(

@player\_id INT, @clock\_time DATETIME, @shot\_made BIT

)

AS

BEGIN TRY

BEGIN TRANSACTION

-- Insert shots and update players here

INSERT INTO dbo.shots(player\_id,clock\_time,shot\_made) VALUES(@player\_id,@clock\_time,@shot\_made);

IF @@ROWCOUNT<>1 THROW 50001,'Insert to shots affected 0 rows, expecting 1',0

UPDATE dbo.players SET shots\_attempted= shots\_attempted+1, shots\_made = shots\_made+@shot\_made WHERE player\_id=@player\_id;

IF @@ROWCOUNT<>1 THROW 50002, 'Update to players affected 0 rows, expecting 1',0

PRINT 'Committing'

COMMIT -- save

END TRY

BEGIN CATCH

SELECT ERROR\_NUMBER() AS error, ERROR\_MESSAGE() AS message

PRINT 'Rolling Back'

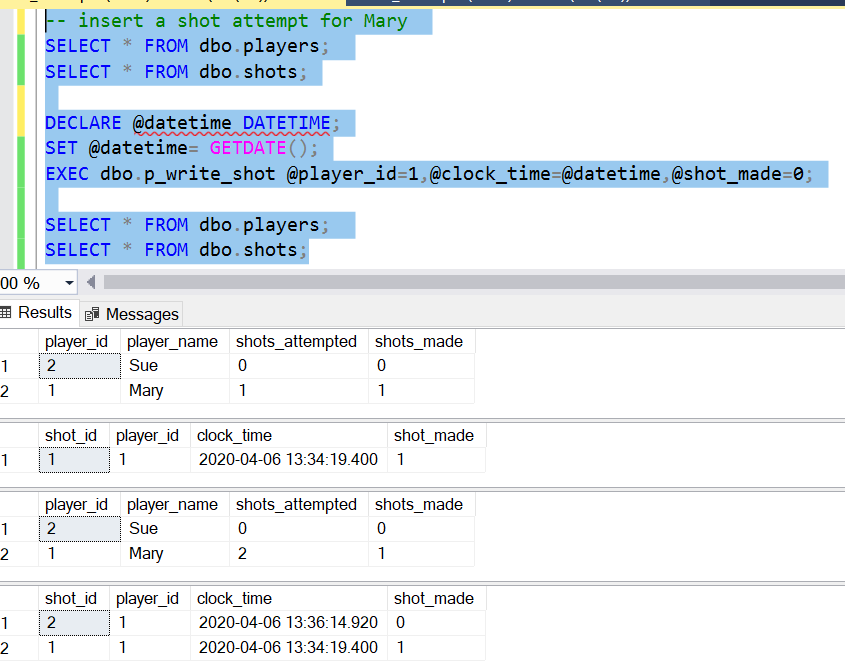
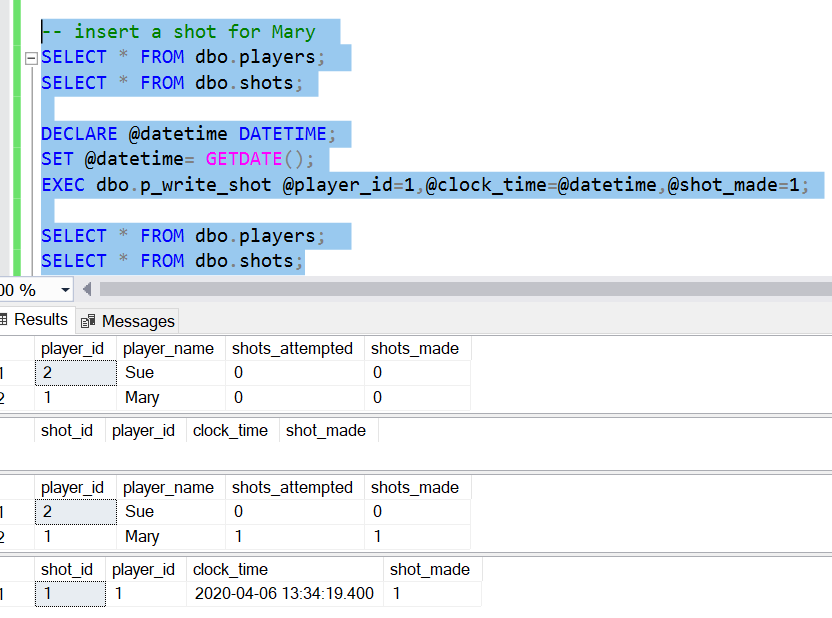
ROLLBACK

END CATCH

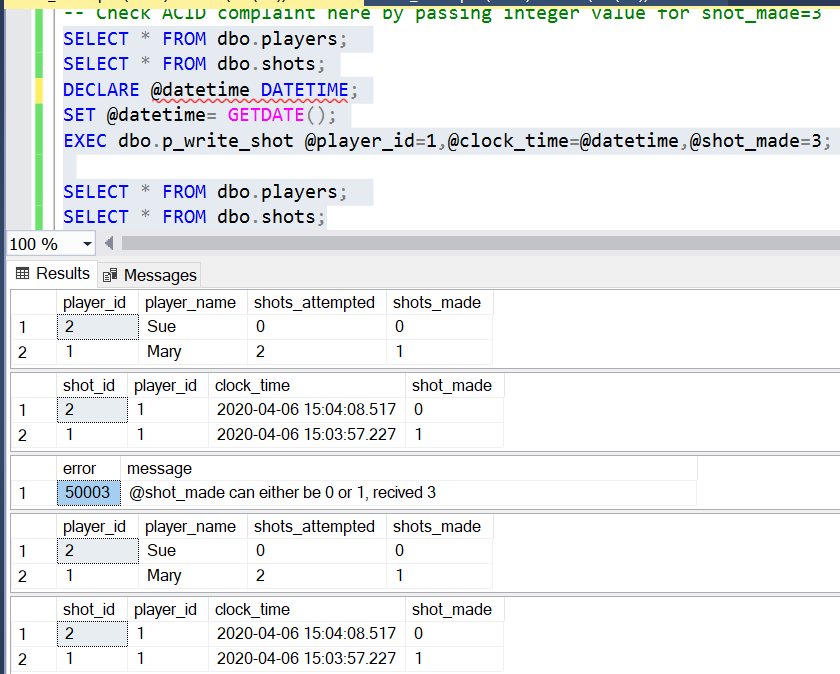
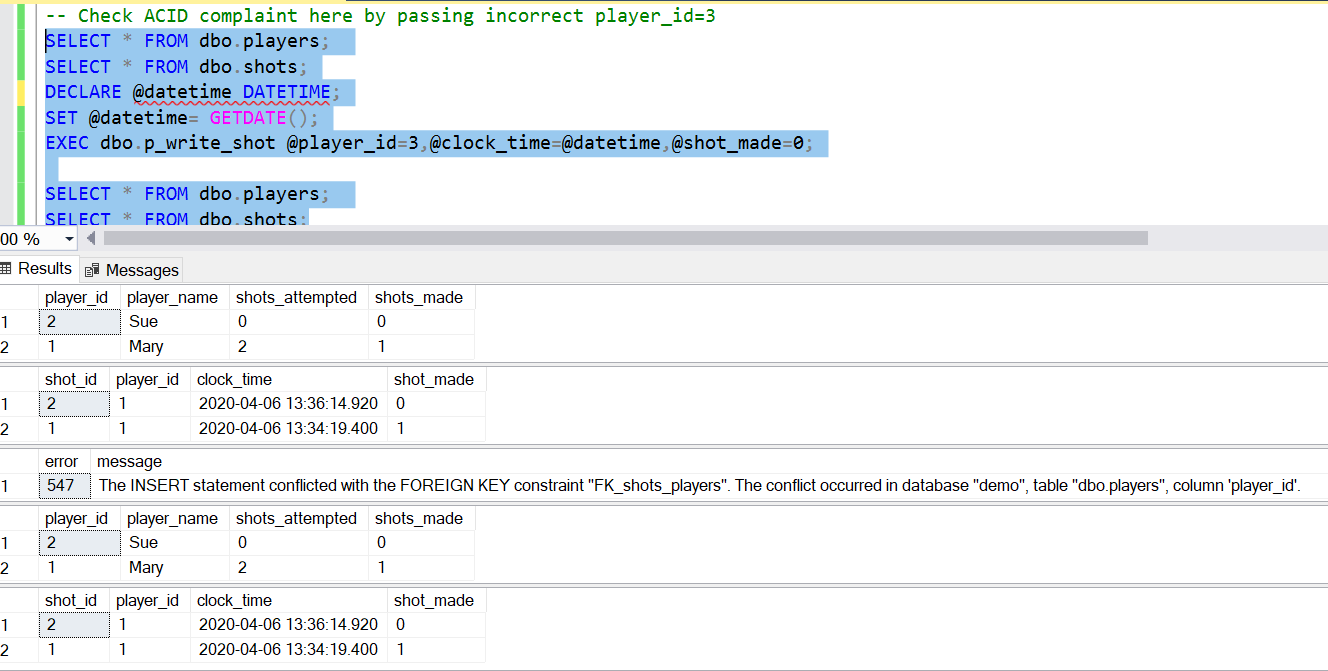
;

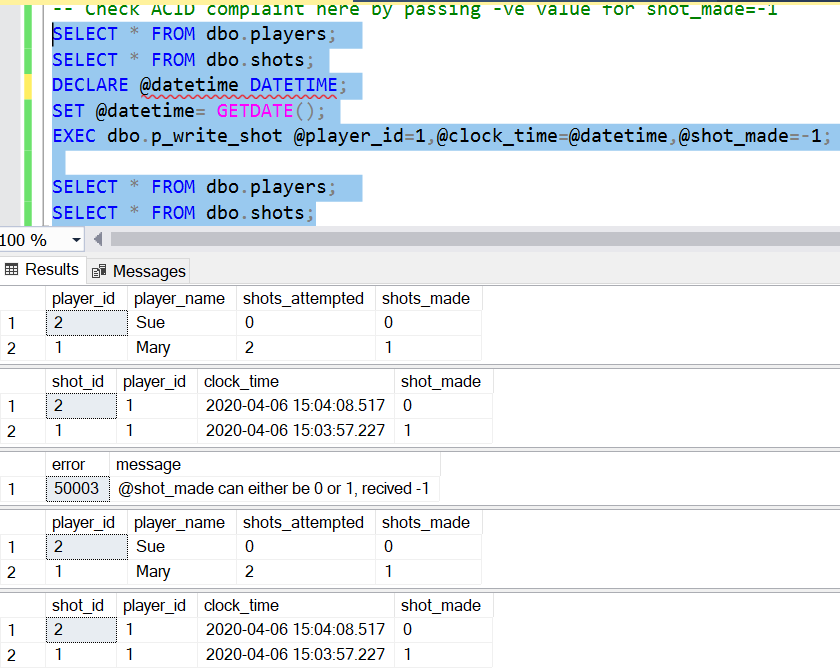
GO

**Valid Transactions**



**Invalid Transactions**





1. Alter the **players** table to be a system-versioned temporal table.

**Solution:**

ALTER TABLE [dbo].[players]

ADD

valid\_from DATETIME2(2) GENERATED ALWAYS AS ROW START HIDDEN

CONSTRAINT df\_valid\_from DEFAULT DATEADD(SECOND, -1, SYSUTCDATETIME())

, valid\_to DATETIME2(2) GENERATED ALWAYS AS ROW END HIDDEN

CONSTRAINT df\_valid\_to DEFAULT '9999.12.31 23:59:59.99'

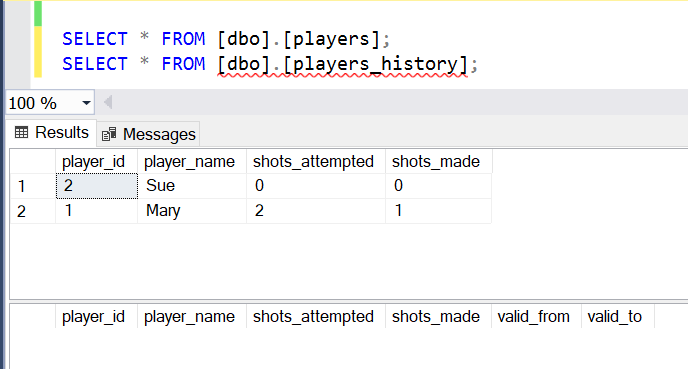
, PERIOD FOR SYSTEM\_TIME (valid\_from,valid\_to);

GO

ALTER TABLE [dbo].[players]

SET (SYSTEM\_VERSIONING = ON (HISTORY\_TABLE=dbo.players\_history));

GO



1. Execute your stored procedure from part 2 to create at least 15 shot records over a 5-minute period. Make sure there are records in the first ½ of the 5-minute period and at few in the last minute of the 5-minute period.

Solution:

-- Create Transactions for 5 minutes at an interval of '00:00:05's

DECLARE @datetime DATETIME;

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=1,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'

SET @datetime= GETDATE();

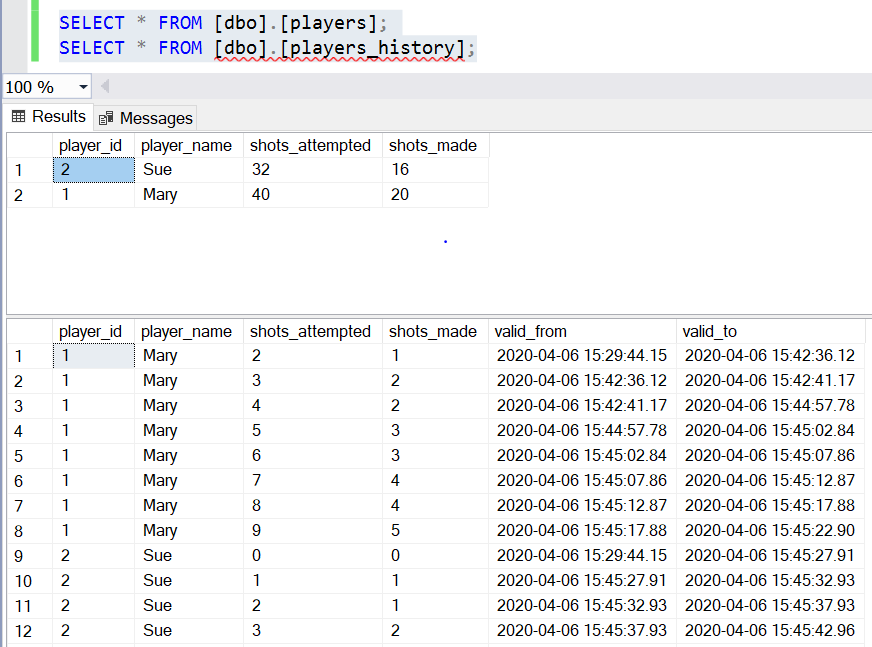
EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=1;

WAITFOR DELAY '00:00:05'

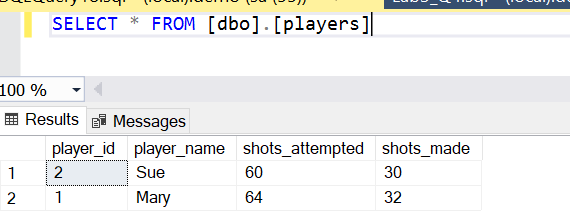
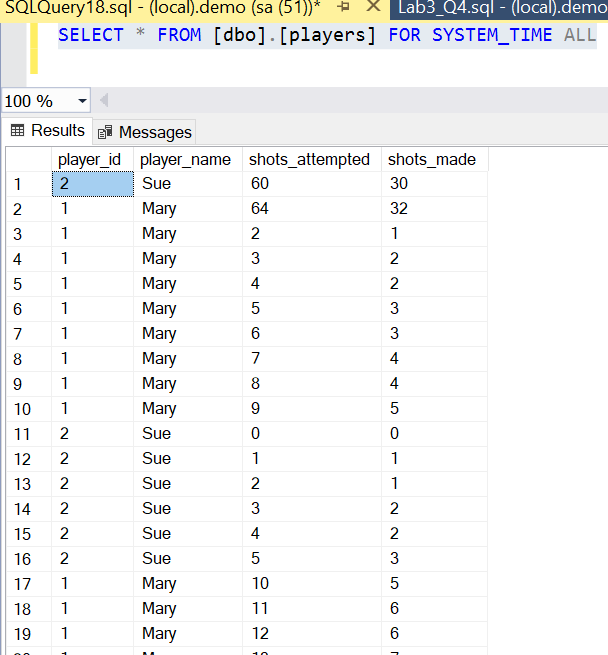
SET @datetime= GETDATE();

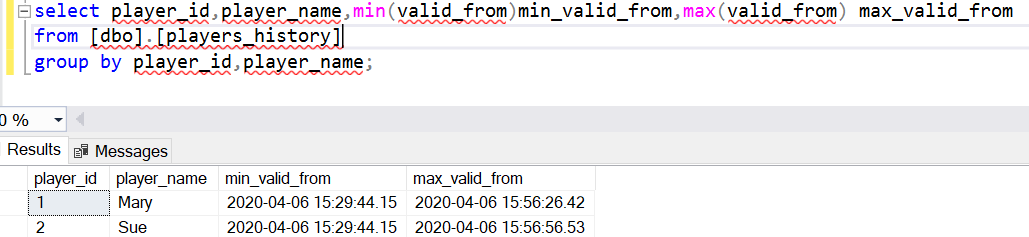
EXEC dbo.p\_write\_shot @player\_id=2,@clock\_time=@datetime,@shot\_made=0;

WAITFOR DELAY '00:00:05'



1. Write SQL queries to show:
   1. The player statistics at the end of the 5-minute period (current statistics).



* 1. The player statistics exactly 2 minutes and 30 seconds into the period.

**Solution**

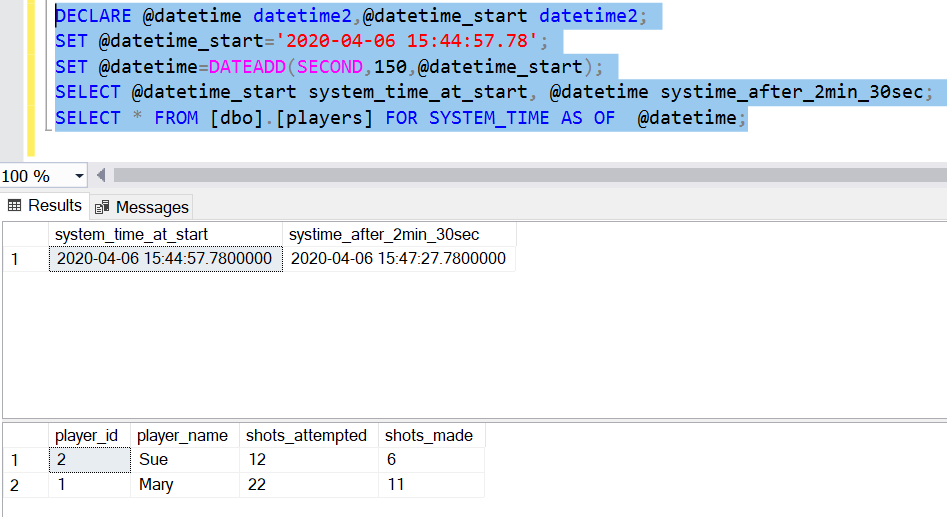
DECLARE @datetime datetime2,@datetime\_start datetime2;

SET @datetime\_start='2020-04-06 15:44:57.78';

SET @datetime=DATEADD(SECOND,150,@datetime\_start);

SELECT @datetime\_start system\_time\_at\_start, @datetime systime\_after\_2min\_30sec;

SELECT \* FROM [dbo].[players] FOR SYSTEM\_TIME AS OF @datetime;



* 1. The player statistics in the last minute of the period.

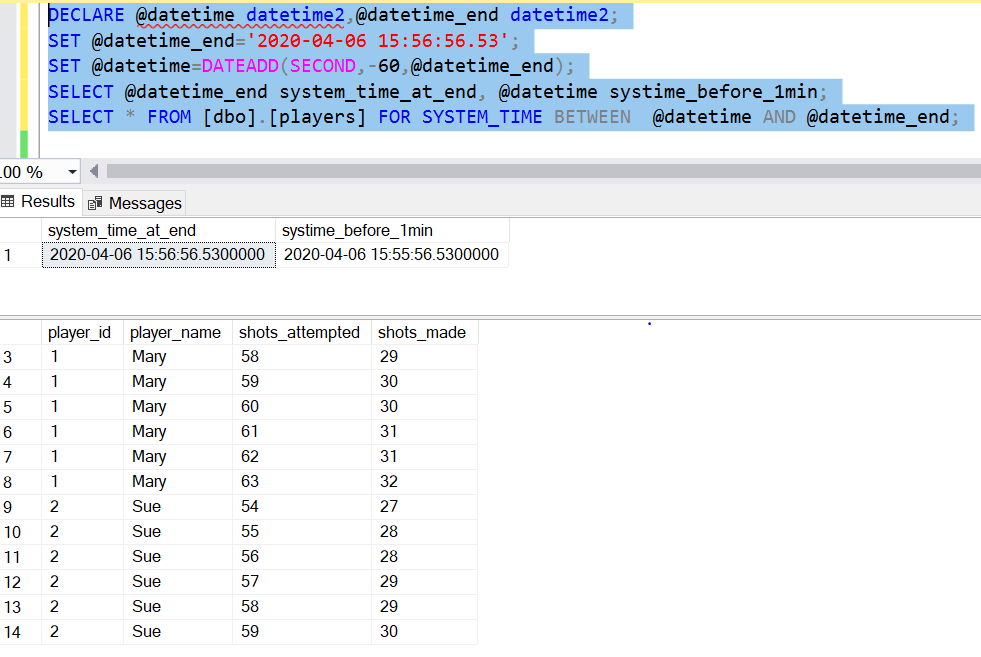
DECLARE @datetime datetime2,@datetime\_end datetime2;

SET @datetime\_end='2020-04-06 15:56:56.53';

SET @datetime=DATEADD(SECOND,-60,@datetime\_end);

SELECT @datetime\_end system\_time\_at\_end, @datetime systime\_before\_1min;

SELECT \* FROM [dbo].[players] FOR SYSTEM\_TIME BETWEEN @datetime AND @datetime\_end;



# Tear-Down

When you are finished with the homework you should stop the environment:

1. From the terminal window where you typed docker-compose up -d type in the following:  
   docker-compose stop

