1. Try creating a temporary table inside a user defined function. Did it work?
2. Try calling a stored procedure from a UDF. Did it work?
3. Create a table with the following script:

USE [AdventureWorks2012]

GO

/\*\*\*\*\*\* Object:  Table [dbo].[Circle]    Script Date: 8/17/2015 6:24:00 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Circle](

[CircleID] [int] NULL,

[Radius] [float] NULL,

[Area] [float] NULL

) ON [PRIMARY]

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (1, 3, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (2, 5, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (3, 3.2, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (4, 5.76, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (5, 3.22, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (6, 4, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (7, 3.66, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (8, 9.22332, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (9, 12, NULL)

GO

INSERT [dbo].[Circle] ([CircleID], [Radius], [Area]) VALUES (10, 10, NULL)

GO

Now, create a UDF that will accept 1 parameter i.e radius and return the area i.e 3.14 \* radius \* radius and will populate the value in the 3rd column.

i.e run an update statement and call the udf to return the area for the 3 rd column.