Lesson 2: basic CRUD using UWP/WPF

UWP Form List (Data grid View)

ListView and GridView both derive from the ListViewBase class, so they have the same functionality, but display data differently. In this article, when we talk about ListView, the info applies to both the ListView and GridView controls unless otherwise specified. We may refer to classes like ListView or ListViewItem, but the "List" prefix can be replaced with "Grid" for the corresponding grid equivalent (GridView or GridViewItem).

Using the code

Create Database and Table

We will create a StudentDetails table to be used for the Student Profile CRUD Operations. The following is the script to create a database and Table query. Run this script in your SQL Server. I have used SQL Server 2014.

Hide Shrink ▲ Copy Code

```
--Script to create DB,Table and sample Insert data
USE MASTER;
-- 1) Check for the Database Exists .If the database is exist then drop and create new DB
IF EXISTS (SELECT [name] FROM sys.databases WHERE [name] = 'StudentsDB' )
BEGIN
ALTER DATABASE StudentsDB SET SINGLE_USER WITH ROLLBACK IMMEDIATE
DROP DATABASE StudentsDB ;
END
```

```
CREATE DATABASE StudentsDB
GO

USE StudentsDB
GO

-- 1) ////////// ToysDetails table

-- Create Table  ToysDetails ,This table will be used to store the details like Toys Information

IF EXISTS ( SELECT [name] FROM sys.tables WHERE [name] = 'StudentDetails' )
DROP TABLE StudentDetails
GO

CREATE TABLE StudentDetails
(
   std_ID int  identity(1,1),
   StudentName VARCHAR(100)  NOT NULL,
   Email VARCHAR(100)  NOT NULL,
   Phone VARCHAR(100)  NOT NULL,
   Address VARCHAR(100)  NOT NULL,
   IMAGEs varbinary(MAX)
   CONSTRAINT [PK_StudentDetails] PRIMARY KEY CLUSTERED
(
  [std_ID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]

GO

select * from StudentDetails
```

After creating our Table we will create a Stored Procedure for our CRUD Operations.

Hide   Shrink ▲   Copy Code

```
-- 1) Stored procedure to Select Student Details
-- Author      : Shanu
-- Create date : 2015-12-01
-- Description : Student Details
-- Tables used :  Student Details
-- Modifier    : Shanu
-- Modify date : 2015-12-01
-- =============================================
-- exec USP_Student_Select ''
-- =============================================
CREATE PROCEDURE [dbo].[USP_Student_Select]
  (
     @StudentName          VARCHAR(100)     = ''
      )
AS
BEGIN
        select   std_ID as StdNO,
```

```sql
                 StudentName as StdName,
                 Email as Email,
                 Phone as Phone,
                 Address as Address,
                 IMAGEs as StdImage

        FROM StudentDetails
         Where
               StudentName like @StudentName +'%'
         ORDER BY
             StudentName

END

-- to Select by Student ID
CREATE PROCEDURE [dbo].[USP_StudentID_Select]
   (
     @std_ID         int
      )
AS
BEGIN
        select   std_ID as StdNO,
                 StudentName as StdName,
                 Email as Email,
                 Phone as Phone,
                 Address as Address,
                 IMAGEs as StdImage

        FROM StudentDetails
        Where
            std_ID = @std_ID

END



-- To Insert Student Detail
CREATE PROCEDURE [dbo].[USP_Student_Insert]
  (
   @StudentName     VARCHAR(100),
   @Email           VARCHAR(100)     = '',
   @Phone           VARCHAR(100)     = '',
   @Address         VARCHAR(100)     = '',
   @IMAGEs          varbinary(MAX)
    )
AS
BEGIN
     IF   NOT   EXISTS   (SELECT   StudentName   FROM   StudentDetails   WHERE
StudentName=@StudentName)
        BEGIN

            INSERT INTO StudentDetails
       (StudentName   ,Email     ,Phone      ,Address  ,IMAGEs)
   VALUES
       (@StudentName    ,@Email       ,@Phone       ,@Address       ,@IMAGEs)

        Select 'Inserted' as results

        END
      ELSE
         BEGIN
               Select 'Exists' as results
          END
END

-- To Update Student Detail
CREATE PROCEDURE [dbo].[USP_Student_Update]
  (   @std_ID             Int=0,
   @StudentName     VARCHAR(100),
   @Email          VARCHAR(100)     = '',
   @Phone          VARCHAR(100)     = '',
   @Address        VARCHAR(100)     = '',
   @IMAGEs         varbinary(MAX)
    )
AS
BEGIN


            UPDATE  StudentDetails SET
                    StudentName = @StudentName  ,
                    Email       =@Email,
                    Phone       =@Phone,
                    Address     =@Address,
                    IMAGEs      =@IMAGEs
              WHERE
                  std_ID=@std_ID


        Select 'Updated' as results
```

```sql
END



-- to Delete
CREATE PROCEDURE [dbo].[USP_Student_Delete]
   (  @std_ID              Int=0 )
AS
BEGIN

       DELETE FROM StudentDetails WHERE  std_ID=@std_ID

        Select 'Deleted' as results

END
```