-- ================================================

-- Template generated from Template Explorer using:

-- Create Procedure (New Menu).SQL

--

-- Use the Specify Values for Template Parameters

-- command (Ctrl-Shift-M) to fill in the parameter

-- values below.

--

-- This block of comments will not be included in

-- the definition of the procedure.

-- ================================================

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Create new procedures/tests

-- =============================================

CREATE PROCEDURE [dbo].[uspProcedures\_Add] @name Varchar(max), @desc text, @type int, @fee decimal(10,2), @active bit

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Insert statements for procedure here

Declare @Ret int;

Set @Ret = -1;

If Exists (Select 1 From dbo.Procedures Where name = @name)

Begin

Set @Ret = 0;

return;

End

Else

Begin

Insert Into dbo.Procedures(name,description,type,fee, active)

Values (@name,@desc,@type,@fee, @active);

set @Ret = 1;

End

select @Ret

END

GO

-- ================================================

-- Template generated from Template Explorer using:

-- Create Procedure (New Menu).SQL

--

-- Use the Specify Values for Template Parameters

-- command (Ctrl-Shift-M) to fill in the parameter

-- values below.

--

-- This block of comments will not be included in

-- the definition of the procedure.

-- ================================================

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Create new procedures/tests

-- =============================================

CREATE PROCEDURE [dbo].[uspProcedures\_Edit] @id int, @name Varchar(max), @desc text, @type int, @fee decimal(10,2), @active bit

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @Ret int;

SET @Ret = -1;

IF Exists (SELECT 1 FROM dbo.[Procedures] WHERE [name] = @name and id <> @id)

BEGIN

SET @Ret = 0;

END

ELSE

BEGIN

UPDATE dbo.[Procedures] SET [name] = @name, [description]=@desc, type=@type, fee=@fee, active=@active WHERE [id]=@id

SET @Ret = 1;

END

SELECT @Ret

END

GO

-- ================================================

-- Template generated from Template Explorer using:

-- Create Procedure (New Menu).SQL

--

-- Use the Specify Values for Template Parameters

-- command (Ctrl-Shift-M) to fill in the parameter

-- values below.

--

-- This block of comments will not be included in

-- the definition of the procedure.

-- ================================================

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Create new procedures/tests

-- =============================================

CREATE PROCEDURE [dbo].[uspProcedures\_Get] @SearchBy varchar(50), @SearchValue varchar(100)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @sql nvarchar(max);

SET @sql = 'SELECT p.id, p.name, p.description, p.type, t.type\_name, p.fee, p.active from Procedures p JOIN Procedure\_Types t ON t.id=p.type ';

IF(@SearchBy <> 'All')

BEGIN

SET @sql = @sql + ' WHERE ' + @SearchBy + ' like ''' + @SearchValue + '%''';

END

EXEC(@sql);

END

GO

-- ================================================

-- Template generated from Template Explorer using:

-- Create Procedure (New Menu).SQL

--

-- Use the Specify Values for Template Parameters

-- command (Ctrl-Shift-M) to fill in the parameter

-- values below.

--

-- This block of comments will not be included in

-- the definition of the procedure.

-- ================================================

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspProcedureTypes\_Combo] @id int

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @sql nvarchar(max);

SET @sql = 'SELECT id, type\_name from Procedure\_Types WHERE Active = 1';

IF(@id > 0)

BEGIN

SET @sql = @sql + ' OR id=''' + CAST(@id AS varchar(30)) + '''';

END

SET @sql = @sql + ' ORDER BY Active, type\_name';

EXEC(@sql);

END

GO

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspProcedures\_Single] Script Date: 12/1/2017 9:37:41 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspProcedures\_Single] @id int

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

SELECT p.\*, t.type\_name FROM dbo.[PROCEDURES] p JOIN dbo.Procedure\_Types t ON t.id=p.type WHERE p.id=@id

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspOptions\_Get] Script Date: 12/2/2017 11:30:52 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE procedure [dbo].[uspOptions\_Get] @SearchBy varchar(50), @SearchValue varchar(100) AS

BEGIN

DECLARE @sql nvarchar(max);

SET @sql = 'SELECT id, op\_name, op\_description, op\_value FROM Options';

IF(@SearchBy <> 'All')

BEGIN

SET @sql = @sql + ' WHERE ' + @SearchBy + ' LIKE ''' + @SearchValue + '%''';

END

EXEC(@sql);

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspProcedureTypes\_Add] Script Date: 12/2/2017 11:17:36 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[uspOptions\_Add] @name Varchar(max), @description text, @val Varchar(max) AS

DECLARE @Ret int;

SET @Ret = -1;

IF Exists (SELECT 1 From dbo.[Options] WHERE op\_name = @name)

BEGIN

SET @Ret = 0;

return;

END

ELSE

BEGIN

INSERT INTO dbo.[Options] (op\_name, op\_description, op\_value)

Values (@name, @description, @val);

SET @Ret = 1;

End

SELECT @Ret;

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspProcedures\_Edit] Script Date: 12/2/2017 11:26:50 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Modify options

-- =============================================

CREATE PROCEDURE [dbo].[uspOptions\_Edit] @id int, @name Varchar(max), @desc text, @val Varchar(max)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @Ret int;

SET @Ret = -1;

IF Exists (SELECT 1 FROM dbo.[Options] WHERE [op\_name] = @name and id <> @id)

BEGIN

SET @Ret = 0;

END

ELSE

BEGIN

UPDATE dbo.[Options] SET op\_name = @name, op\_description=@desc, op\_value=@val WHERE [id]=@id

SET @Ret = 1;

END

SELECT @Ret

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspOptions\_Single]

Script Date: 12/2/2017 11:33:02 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[uspOptions\_Single] @name varchar(100) AS

BEGIN

SELECT \* FROM dbo.[Options] WHERE op\_name=@name

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspOptions\_Single]

Script Date: 12/2/2017 11:33:02 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[uspOptions\_Many] @name varchar(100) AS

BEGIN

SELECT op\_name, op\_value FROM dbo.[Options] WHERE op\_name IN (@name)

END

CREATE TABLE dbo.Menu\_Items

(

id int NOT NULL IDENTITY (1, 1),

parent\_id int NOT NULL,

menu\_name varchar(50) NOT NULL,

menu\_text varchar(50) NULL,

active bit NOT NULL,

locked bit NOT NULL ) ON [PRIMARY]

GO

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspProcedureTypes\_Combo] Script Date: 12/4/2017 1:58:49 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspUsers\_ValidateLogin] @empid varchar(20), @pass varchar(50)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @salt1 VARBINARY(4);

DECLARE @hash VARBINARY(MAX);

DECLARE @hash1 VARBINARY(MAX);

DECLARE @Ret int;

SET @Ret = 0;

SELECT @salt1=salt,@hash1=[password] FROM dbo.Users WHERE emp\_id=@empid AND active=1;

if(@salt1 <> '' AND @hash1 <> '')

BEGIN

SET @hash = 0x0200 + @salt1 + HASHBYTES('SHA2\_512', CAST(@pass AS VARBINARY(MAX)) + @salt1);

IF(@hash = @hash1)

BEGIN

SET @Ret = 1; ;

END

ELSE

BEGIN

SET @Ret = -1;

END

END

SELECT @Ret;

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspProcedureTypes\_Combo] Script Date: 12/4/2017 1:58:49 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspUsers\_SetPwdHash] @empid varchar(20), @pass varchar(50), @utype int

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @salt1 VARBINARY(4) = CRYPT\_GEN\_RANDOM(4);

DECLARE @hash1 VARBINARY(MAX);

DECLARE @id VARCHAR(20);

SET @hash1 = 0x0200 + @salt1 + HASHBYTES('SHA2\_512', CAST(@pass AS VARBINARY(MAX)) + @salt1);

SELECT @id=emp\_id FROM dbo.Users WHERE emp\_id=@empid;

if(@id != '')

BEGIN

UPDATE dbo.Users SET password=@hash1,salt=@salt1 WHERE emp\_id=@empid;

END

ELSE

BEGIN

INSERT INTO dbo.Users (emp\_id,password,salt,user\_type\_id,active) VALUES(@empid,@hash1,@salt1,@utype,1);

END

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspUsers\_SetPwdHash] Script Date: 12/4/2017 3:20:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspUsers\_LoggedUser] @empid varchar(20)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

SELECT u.\*, t.type\_name,s.staff\_id, s.staff\_name, s.staff\_phone FROM dbo.Users u JOIN dbo.User\_Types t ON u.user\_type\_id=t.id

LEFT JOIN

(SELECT d.id as staff\_id, d.user\_id as staff\_user\_id, CONCAT(d.first\_name, ' ', d.last\_name) as staff\_name, d.phone as staff\_phone FROM dbo.Users u JOIN dbo.Doctors d ON u.id=d.user\_id WHERE u.emp\_id=@empid

UNION

SELECT d.id as staff\_id, d.user\_id as staff\_user\_id, CONCAT(d.first\_name, ' ', d.last\_name) as staff\_name, d.phone as staff\_phone FROM dbo.Users u JOIN dbo.Staffs d ON u.id=d.user\_id WHERE u.emp\_id=@empid)

s ON u.id=s.staff\_user\_id

WHERE u.emp\_id=@empid;

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspUsers\_SetPwdHash] Script Date: 12/4/2017 3:20:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspUsers\_SetLogDate] @empid varchar(20)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DECLARE @Ret int;

SET @Ret = 0;

UPDATE dbo.Users SET log\_date=GETDATE() WHERE emp\_id=@empid;

SET @Ret = 1;

SELECT @Ret;

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspUsers\_SetPwdHash] Script Date: 12/4/2017 3:20:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE PROCEDURE [dbo].[uspMenus\_GetActive] @active bit

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

SELECT \* FROM dbo.Menu\_Items WHERE active=@active ORDER BY parent\_id, menu\_order, menu\_text;

END

USE [hospitalERP\_db]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[uspUsers\_SetPwdHash] Script Date: 12/4/2017 3:20:56 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Insync Tech Solutions

-- Create date: 2017-12-01

-- Description: Datatable for Procedure Types Combobox

-- =============================================

CREATE TYPE dbo.MenuList

AS TABLE

(

id INT

);

GO

CREATE PROCEDURE uspMenuUserTypes\_Add

@List AS dbo.MenuList READONLY, @utype int

AS

BEGIN

SET NOCOUNT ON;

DELETE FROM dbo.Menu\_User\_Types WHERE user\_type\_id=@utype;

INSERT INTO dbo.Menu\_User\_Types SELECT id, @utype FROM @List;

END

GO