Microsoft Sql 2016 16.0.30 version

* Create

ALTER Proc [dbo].[usp\_AddPlan]

@PlanName nvarchar(100),

@MessageId int output,

@Message varchar(100) output

As

BEGIN

BEGIN try

if not exists(select 1 from [PostPayPlan] where PlanDescription=@PlanName)

BEGIN

BEGIN TRAN

insert into [PostPayPlan]

select @PlanName,getDate(),NULL,1

set @Message='Record saved'

set @MessageId=1

--print @Message

COMMIT tran

END

ELSE

BEGIN

set @Message='Plan Name already exists'

set @MessageId=2

--print @Message

END

END try

BEGIN Catch

rollback tran

set @Message=ERROR\_MESSAGE();

set @MessageId=2

--print @Message

ENd Catch

END

* C#

public MessageDetails AddSubPlan(int PlanId, string SubPlanName)

{

MessageDetails \_MessageDetails = new MessageDetails();

try

{

SqlConnection con = new SqlConnection(SQLConnectionString);

SqlCommand cmd = new SqlCommand("usp\_AddSubPlan", con);

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@PlanId", PlanId);

cmd.Parameters.AddWithValue("@SubPlanName", SubPlanName);

cmd.Parameters.Add("@MessageId", SqlDbType.Int).Direction = ParameterDirection.Output;

cmd.Parameters.Add("@Message", SqlDbType.NVarChar, 500).Direction = ParameterDirection.Output;

if (con.State == ConnectionState.Closed)

con.Open();

SqlDataReader rdr = null;

rdr = cmd.ExecuteReader();

\_MessageDetails.MessageId = Convert.ToInt16(cmd.Parameters["@MessageId"].Value);

\_MessageDetails.Message = Convert.ToString(cmd.Parameters["@Message"].Value);

con.Close();

}

catch (Exception ex)

{

}

return \_MessageDetails;

}

* EDIT

ALTER Proc [dbo].[usp\_UpdatePlan]--usp\_UpdatePlan 1002,'Iridium - Postpaid SIM card','',''

@PlanId int,

@PlanName nvarchar(100),

@MessageId int output,

@Message varchar(100) output

As

BEGIN

BEGIN try

if not exists(select 1 from [PostPayPlan] where PlanDescription=@PlanName and PlanId<>@PlanId and IsActive=1)

BEGIN

BEGIN TRAN

update [PostPayPlan]

set

PlanDescription=@PlanName,

UpdateDate=GETDATE()

where PlanId=@PlanId

set @Message='Record updated successfully'

set @MessageId=1

print @Message

COMMIT TRAN

END

ELSE

BEGIN

set @Message='Plan name already exists'

set @MessageId=2

print @Message

END

END try

BEGIN Catch

rollback tran

set @Message=ERROR\_MESSAGE();

set @MessageId=2

--print @Message

ENd Catch

END

* C#

public MessageDetails UpdateSubPlan(int SubPlanId, int PlanId, string SubPlanName)

{

MessageDetails \_MessageDetails = new MessageDetails();

try

{

SqlConnection con = new SqlConnection(SQLConnectionString);

SqlCommand cmd = new SqlCommand("usp\_UpdateSubPlan", con);

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@SubPlanId", SubPlanId);

cmd.Parameters.AddWithValue("@PlanId", PlanId);

cmd.Parameters.AddWithValue("@SubPlanName", SubPlanName);

cmd.Parameters.Add("@MessageId", SqlDbType.Int).Direction = ParameterDirection.Output;

cmd.Parameters.Add("@Message", SqlDbType.NVarChar, 500).Direction = ParameterDirection.Output;

if (con.State == ConnectionState.Closed)

con.Open();

SqlDataReader rdr = null;

rdr = cmd.ExecuteReader();

\_MessageDetails.MessageId = Convert.ToInt16(cmd.Parameters["@MessageId"].Value);

\_MessageDetails.Message = Convert.ToString(cmd.Parameters["@Message"].Value);

con.Close();

}

catch (Exception ex)

{

}

return \_MessageDetails;

}

* Select By Primary

ALTER Proc [dbo].[usp\_PlanSelectById]--usp\_PlanSelectById 1009,''

@PlanId int,

@Output varchar(max) output

As

BEGIN

set @Output =(select \* from PostPayPlan where PlanId=@PlanId for json Path )

print @Output

END

* C#

public string SubPlanSelectById(int SubPanId)

{

string OutPut = string.Empty;

try

{

using (SqlConnection conn = new SqlConnection(SQLConnectionString))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("usp\_SubPlanSelectById", conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@SubPanId", SubPanId);

cmd.Parameters.Add("@Output", SqlDbType.NVarChar, -1).Direction = ParameterDirection.Output;

cmd.ExecuteReader();

OutPut = cmd.Parameters["@Output"].Value.ToString();

}

}

}

catch (Exception ex)

{

}

return OutPut;

}

* List

ALTER Proc [dbo].[usp\_GetPlanList]--usp\_GetPlanList ''

@Output varchar(max) output

As

BEGIN

set @Output =(select PlanId,PlanDescription as [Plan Name] from PostPayPlan where IsActive=1 order by isnull(UpdateDate,CreatedDate) desc for json Path)

--print @Output

END

* C#

public string GetSubPlanList()

{

string OutPut = string.Empty;

try

{

using (SqlConnection conn = new SqlConnection(SQLConnectionString))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("usp\_GetSubPlanList", conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.Add("@Output", SqlDbType.NVarChar, -1).Direction = ParameterDirection.Output;

cmd.ExecuteReader();

OutPut = cmd.Parameters["@Output"].Value.ToString();

}

}

}

catch (Exception ex)

{

}

return OutPut;

}

* Delete

ALTER Proc [dbo].[usp\_DeletePlan]

@PlanId int,

@MessageId int output,

@Message varchar(100) output

As

BEGIN

BEGIN try

BEGIN TRAN

update PostPayPlan

set

IsActive=0 where PlanId=@PlanId

update PostPaySubPlan

set

IsActive=0 where PlanId=@PlanId

set @Message='Record updated successfully'

set @MessageId=1

--print @Message

COMMIT tran

END try

BEGIN Catch

rollback tran

set @Message=ERROR\_MESSAGE();

set @MessageId=2

--print @Message

ENd Catch

END

* C#

public MessageDetails DeletePlan(int PlanId)

{

MessageDetails \_MessageDetails = new MessageDetails();

try

{

SqlConnection con = new SqlConnection(SQLConnectionString);

SqlCommand cmd = new SqlCommand("usp\_DeletePlan", con);

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@PlanId", PlanId);

cmd.Parameters.Add("@MessageId", SqlDbType.Int).Direction = ParameterDirection.Output;

cmd.Parameters.Add("@Message", SqlDbType.NVarChar, 500).Direction = ParameterDirection.Output;

if (con.State == ConnectionState.Closed)

con.Open();

SqlDataReader rdr = null;

rdr = cmd.ExecuteReader();

\_MessageDetails.MessageId = Convert.ToInt16(cmd.Parameters["@MessageId"].Value);

\_MessageDetails.Message = Convert.ToString(cmd.Parameters["@Message"].Value);

con.Close();

}

catch (Exception ex)

{

}

return \_MessageDetails;

}

public class MessageDetails

{

public int MessageId { get; set; }

public string Message { get; set; }

public string ListMessage { get; set; }

}