**Applicants Project Database Tables:**

**Current Implementation:**

Create database resume:

create database resume

Create Table ApplicantsProfile:

USE [resume]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ApplicantsProfile] Script Date: 30-08-2015 14:14:50 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ApplicantsProfile](

[ApplicantID] [int] IDENTITY(1001,1) NOT NULL,

[FirstName] [nvarchar](50) NULL,

[LastName] [nvarchar](50) NULL,

[PhoneNo] [nvarchar](20) NOT NULL,

[EmailID] [nvarchar](50) NULL,

[EducationalQualification] [nvarchar](50) NULL,

[PreviousCompany] [nvarchar](50) NULL,

[YearsOfExperience] [int] NULL,

[ApplicantAddress] [nvarchar](150) NULL,

[ApplicantResume] [nvarchar](400) NULL,

[Comments] [nvarchar](100) NULL,

CONSTRAINT [PK\_ApplicantsProfile] PRIMARY KEY CLUSTERED

(

[ApplicantID] 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

**Scope of Implementation :**

Create a master table called “Applicants” setting ApplicantID as primary key and child table as “ApplicantDetails” where ID is primary key and Applicant ID is foreign key.

Write a Stored Proecedure “InsertApplicant” to insert the records in both tables on basis of Id and Applicant ID relation.

**Advantages of using Sp :**

Creating this master- child relation between the two tables will allow user to enter multiple Companies and Education Qualifications in the profile.

By creating the sp for inserting values we can resuse the same relationship in already executed format.

USE [resume]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Applicants](

[ApplicantID] [int] IDENTITY(100,1) NOT NULL,

[PhoneNo] [numeric](18, 0) NOT NULL,

[EmailID] [nvarchar](50) NOT NULL,

CONSTRAINT [PK\_Applicants] PRIMARY KEY CLUSTERED

(

[ApplicantID] 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

Create Child Table :Applicant Details:

USE [resume]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ApplicantDetails](

[ID] [int] IDENTITY(200,1) NOT NULL,

[ApplicantID] [int] NOT NULL,

[FirstName] [nvarchar](50) NULL,

[LastName] [nvarchar](50) NULL,

[EducationalQualification] [nvarchar](50) NULL,

[PreviousCompany] [nvarchar](50) NULL,

[YearsOfExperience] [int] NULL,

[ApplicantAddress] [nvarchar](150) NULL,

[Resume] [nvarchar](400) NULL,

[Comments] [nvarchar](100) NULL,

CONSTRAINT [PK\_ApplicantDetails] PRIMARY KEY CLUSTERED

(

[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

ALTER TABLE [dbo].[ApplicantDetails] WITH CHECK ADD CONSTRAINT [FK\_ApplicantDetails\_Applicants] FOREIGN KEY([ApplicantID])

REFERENCES [dbo].[Applicants] ([ApplicantID])

GO

ALTER TABLE [dbo].[ApplicantDetails] CHECK CONSTRAINT [FK\_ApplicantDetails\_Applicants]

GO

**Stored Procedure to Add an Applicant:**

CREATE PROCEDURE InsertApplicant

@ApplicantID int,

@PhoneNo numeric(18,0),

@EmailID nvarchar(50),

@FirstName nvarchar(50),

@LastName nvarchar(50),

@EducationalQualification nvarchar(50),

@PreviousCompany nvarchar(50),

@YearsOfExperience int,

@ApplicantAddress nvarchar(150),

@Resume nvarchar(400),

@Comments nvarchar(100)

AS

Begin

Set NoCount on

DECLARE @CandidateID int

Insert Applicants(PhoneNo,EmailID)

Values

(@PhoneNo,@EmailID)

Select @CandidateID=@@Identity

Insert into ApplicantDetails

(ApplicantID,FirstName,LastName,EducationalQualification,PreviousCompany,YearsOfExperience,ApplicantAddress,Resume,Comments)

Values

(@CandidateID,@FirstName,@LastName,@EducationalQualification,@PreviousCompany,@YearsOfExperience,@ApplicantAddress,@Resume,@Comments)

End