## Q1: Insert into many to many table (multiple rows)

You are given a databse:

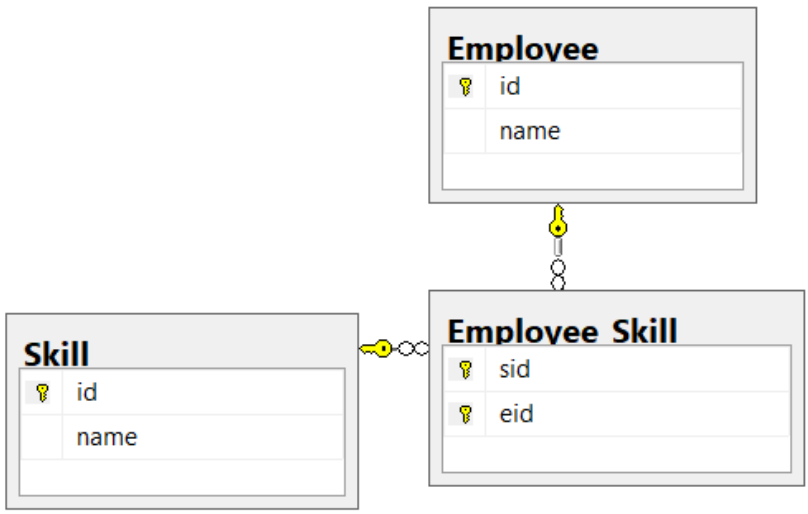
```
USE [EmpSkiDB]
GO
/****** Object:  Table [dbo].[Employee]    Script Date: 3/4/2020 12:51:24 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Employee](
	[id] [int] NOT NULL,
	[name] [varchar](150) NOT NULL,
 CONSTRAINT [PK_Employee] PRIMARY KEY CLUSTERED
(
	[id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[Employee_Skill]    Script Date: 3/4/2020 12:51:24 PM
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Skill](
	[id] [int] NOT NULL,
	[name] [varchar](150) NOT NULL,
 CONSTRAINT [PK_Skills] 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

CREATE TABLE [dbo].[Employee_Skill](
	[sid] [int] NOT NULL,
	[eid] [int] NOT NULL,
 CONSTRAINT [PK_Employee_Skill] PRIMARY KEY CLUSTERED
(
	[sid] ASC,
	[eid] 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

/****** Object:  Table [dbo].[Skill]    Script Date: 3/4/2020 12:51:24 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
INSERT [dbo].[Employee] ([id], [name]) VALUES (1, N'Mr A')
INSERT [dbo].[Employee] ([id], [name]) VALUES (2, N'XXXX')
```

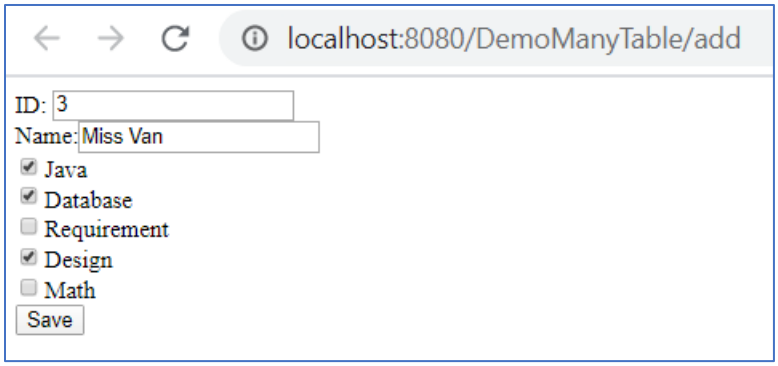
```
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (1, 1)
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (1, 2)
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (2, 1)
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (2, 2)
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (3, 1)
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (3, 2)
INSERT [dbo].[Employee_Skill] ([sid], [eid]) VALUES (4, 2)
INSERT [dbo].[Skill] ([id], [name]) VALUES (1, N'Java')
INSERT [dbo].[Skill] ([id], [name]) VALUES (2, N'Database')
INSERT [dbo].[Skill] ([id], [name]) VALUES (3, N'Requirement')
INSERT [dbo].[Skill] ([id], [name]) VALUES (4, N'Design')
INSERT [dbo].[Skill] ([id], [name]) VALUES (5, N'Math')
ALTER TABLE [dbo].[Employee_Skill]  WITH CHECK ADD  CONSTRAINT
[FK_Employee_Skill_Employee] FOREIGN KEY([eid])
REFERENCES [dbo].[Employee] ([id])
GO
ALTER TABLE [dbo].[Employee_Skill] CHECK CONSTRAINT [FK_Employee_Skill_Employee]
GO
ALTER TABLE [dbo].[Employee_Skill]  WITH CHECK ADD  CONSTRAINT
[FK_Employee_Skill_Skills] FOREIGN KEY([sid])
REFERENCES [dbo].[Skill] ([id])
GO
ALTER TABLE [dbo].[Employee_Skill] CHECK CONSTRAINT [FK_Employee_Skill_Skills]
GO
```

As shown in figure:




To build a java web application to create an employee to its skills, know that an employee can be assigned to several skills, and a skill may contains many of different employees.

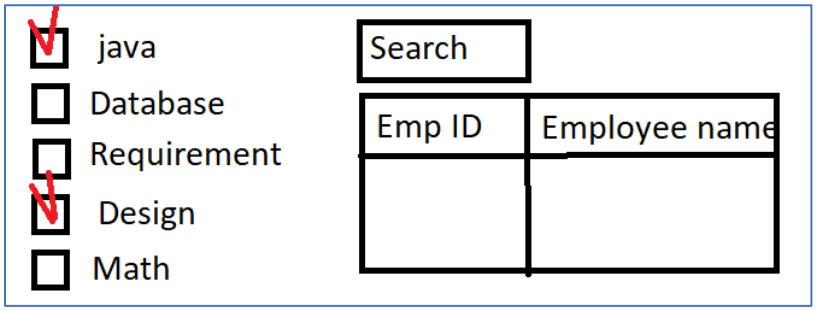
User can access /add to visit add new employee page as shown in figure below




For employees who can load all skills from [Skill] table, to contruct the page.

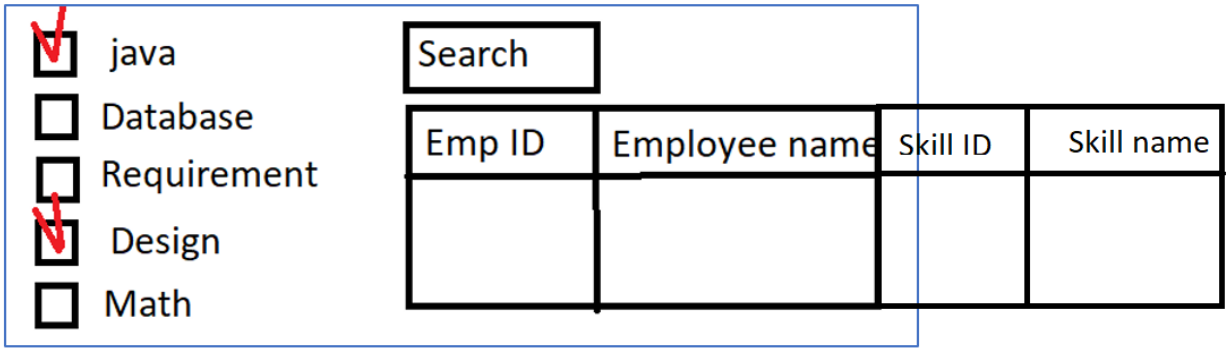
When user clicks [Save] button, the system save new employee to [Employee] table and assign the skill to its selected skills

## Q2: /search




```
select distinct e.id,e.name
     from Employee_Skill es join Employee e on(es.eid=e.id)
        where es.sid in(1,4)
```

## Q3: /search




```
select  e.id,e.name,es.sid,s.name
```

```
     from (Employee_Skill es join Employee e on(es.eid=e.id)) join Skill s
on(es.sid=s.id)
        where es.sid in(1,2)
```