

1. Agregar una clase Curso al proyecto, que incluya una lista de estudiantes inscritos:

```
public class Curso
{
    public int CursoId { get; set; }
    public string Nombre { get; set; }
    public virtual List<Estudiante> MisInscritos { get; set; }
}
```

2. Crear el DbSet correspondiente en el DbContext :

```
public class UniversidadContext:DbContext
{
    public DbSet<Estudiante> Estudiantes { get; set; }
    public DbSet<Curso> Cursos { get; set; }
}
```

Ahora agregaremos un curso que contenga algunos de los estudiantes ya ingresados en la base de datos, y verificaremos que se han creado listándolos.

En la aplicación de consola, en el main() escribir el código siguiente (comentar o eliminar el código mediante el que se crearon los estudiantes):

```
UniversidadContext db = new UniversidadContext();
Curso c1 = new Curso();
c1.Nombre = "Programación";
if (c1.MisInscritos == null) c1.MisInscritos = new List<Estudiante>();
c1.MisInscritos.Add(db.Estudiantes.Find(3));
c1.MisInscritos.Add(db.Estudiantes.Find(2));
db.Cursos.Add(c1);
db.SaveChanges();
var losCursos = from Curso c
                 in db.Cursos
                 select c;
foreach (Curso unC in losCursos)
{
    Console.WriteLine(unC.Nombre);
    if (unC.MisInscritos != null && unC.MisInscritos.Count > 0)
        foreach (Estudiante estud in unC.MisInscritos)
            Console.WriteLine(estud.Nombre);
}
Console.ReadKey();
```

Asumimos que existen los estudiantes con PK = 2 y 3

Ejecutar el programa.

3. Verificar la base de datos.

Desde el explorador de Servidores (CTRL-ALT-S) o desde el Management Studio, verificar que la base de datos se ha modificado. Observar que como no hemos incluido la lista de cursos en el estudiante, EF asumió que cada estudiante tiene un único curso, y le agregó una clave foránea.

4. Agregar la lista de cursos en Estudiante:

```
public class Estudiante
{
    public int EstudianteId { get; set; }
    public string Nombre { get; set; }
    public int edad { get; set; }
    public virtual List<Curso> MisCursos { get; set; }
}
```

5. Sincronización del Modelo con la BD a partir de modificaciones.

Ejecutar nuevamente la aplicación. Observar que se produce una excepción:

```
The model backing the 'UniversidadContext' context has changed since the
database was created. Consider using Code First Migrations to update the
database (http://go.microsoft.com/fwlink/?LinkId=238269).
```

Esto significa que se han realizado cambios de sincronización entre el modelo y la base de datos, y sugiere utilizar la herramienta de Migración para realizar la sincronización. Este tema se verá en el próximo práctico.

Observe también que no siempre la generación automática es apropiada (por ejemplo ¿queremos siempre asociaciones bidireccionales en el modelo?). Para solucionar estos problemas, podemos valernos de Data Annotations y/o Fluent API. En el próximo práctico también se hará uso de Data Annotations.