

CURSO DE HIBERNATE Y JPA

MAPEO DE ASOCIACIONES EN HIBERNATE Y JPA



Ing. Ubaldo Acosta

Por el experto: Ing. Ubaldo Acosta



CURSO DE HIBERNATE Y JPA

www.globalmentoring.com.mx

Hola, te saluda Ubaldo Acosta. Bienvenida o bienvenido nuevamente. Espero que estés listo para comenzar con esta lección.

Vamos a estudiar el tema de Asociaciones utilizando Hibernate y JPA.

¿Estás listo? Ok, ¡Vamos!

MAPEO DE ASOCIACIONES

- Las asociaciones en Hibernate y JPA son las mismas que se manejan en la teoría de bases de datos relacionales.
 - ✓ 1 a 1
 - ✓ 1 a Muchos o Muchos a 1
 - ✓ Muchos a Muchos
- Hibernate y JPA soportan las relaciones mencionadas en los archivos de mapeo de cada clase de Entidad o por medio de anotaciones Java.

CURSO DE HIBERNATE Y JPA

www.globalmentoring.com.mx

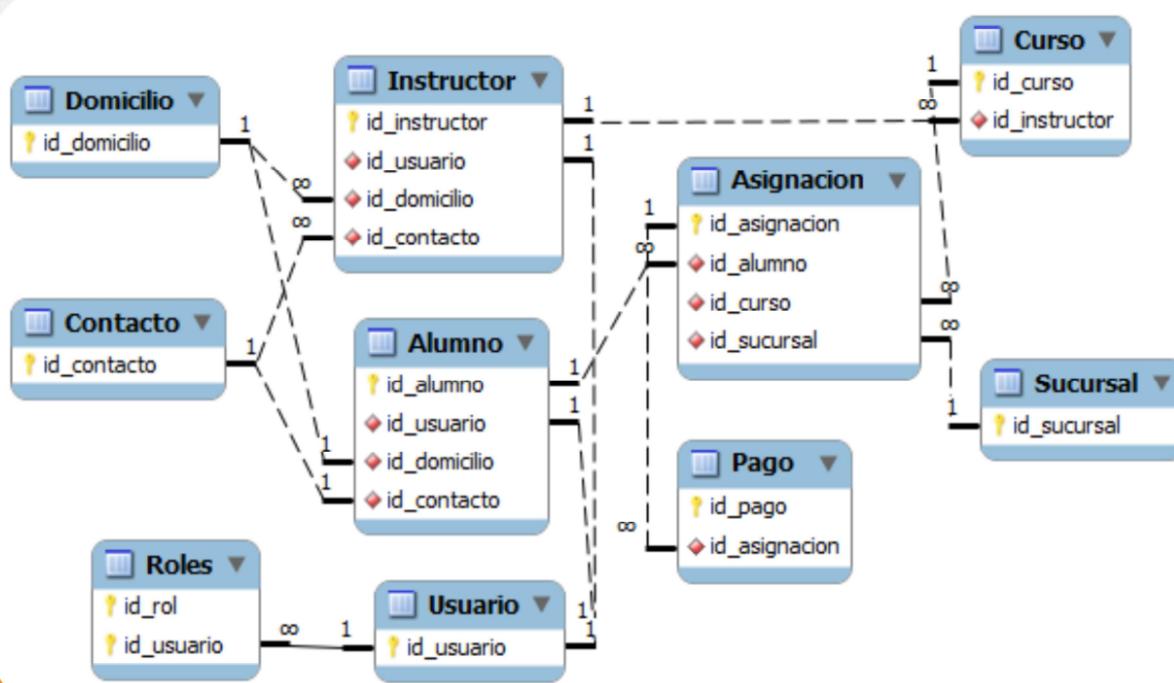
En esta lección vamos a revisar el tema de mapeo.

Las asociaciones en Hibernate y JPA son las mismas que se manejan en la teoría de base de datos relacionales, por ejemplo vamos a tener la relación de 1 a 1, la relación de 1 a muchos o muchos a 1 y la relación de muchos a muchos.

Hibernate y JPA soporta todas las relaciones mencionadas y esto lo vamos a configurar en nuestro archivo de mapeo de cada una de las clases de entidad que estemos manejando o más comúnmente por medio de anotaciones Java.

Vamos a revisar a más detalle los tipos de asociaciones que vamos a estar revisando en este curso.

DIAGRAMA ENTIDAD RELACIÓN “SGA”

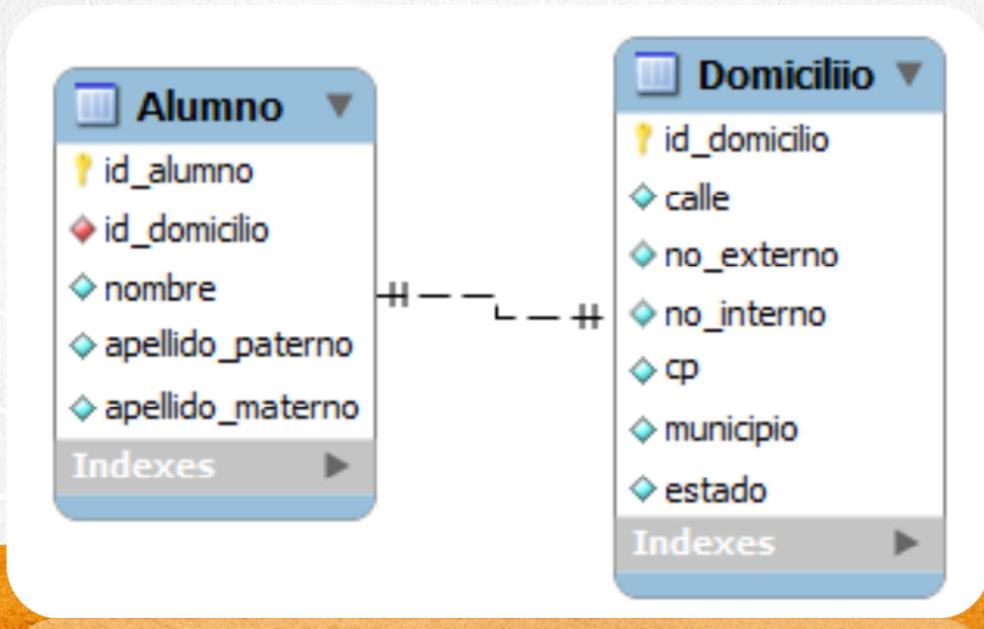


A lo largo del curso utilizaremos el siguiente diagrama Entidad – Relación, el cual utilizamos solo como ejemplo para poder revisar las relaciones y mapeos mencionados anteriormente.

Iremos revisando a más detalle cada una de estas relaciones, pero básicamente tenemos un diagrama de Entidad – Relación que presenta un sistema de Gestión de Alumnos de una Escuela, por ejemplo los alumnos de Global Mentoring. Al sistema Web que utiliza esta base de datos le llamaremos Sistema de Gestión de Alumnos o por sus siglas SGA y este es el esquema Entidad-Relación que utilizaremos a lo largo del curso.

EJEMPLO DE RELACIÓN 1 A 1

- Veamos un ejemplo de relación 1 a 1. Un alumno tiene un Domicilio.



www.globalmentoring.com.mx

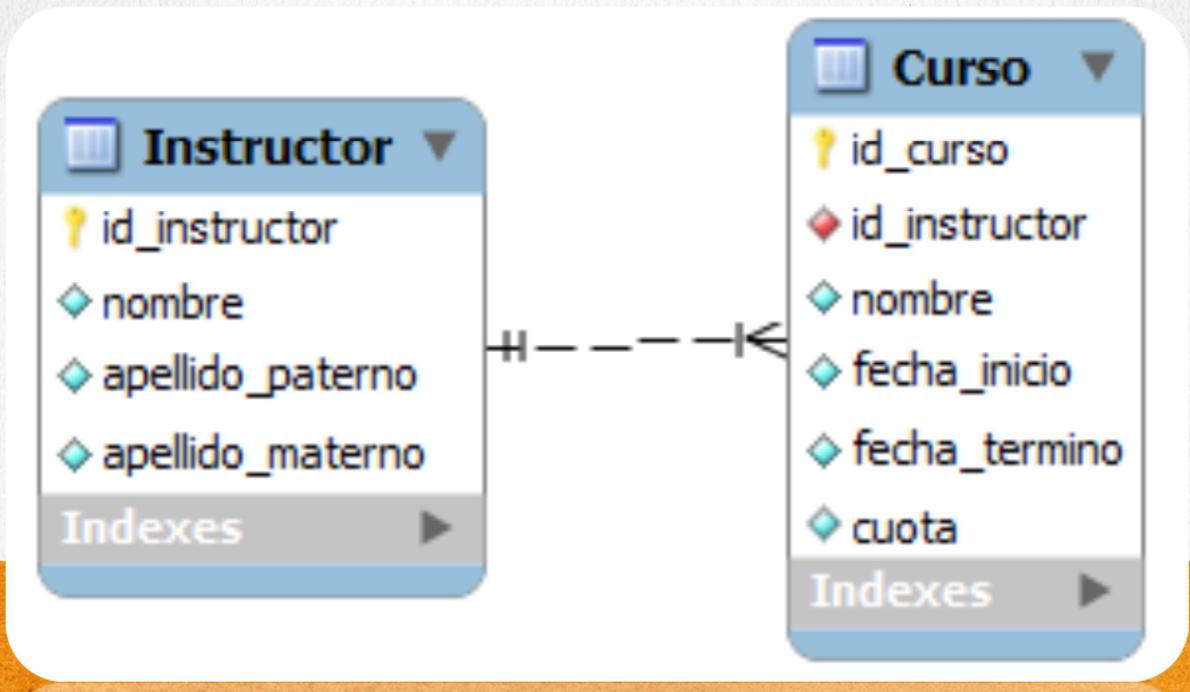
En este diagrama podemos observar una relación de 1 a 1. Como ejemplo estamos poniendo una identidad Alumno tiene un Domicilio relacionado.

En este tipo de relaciones uno a uno la llave foránea (hablando en términos de base de datos) puede quedar en cualquiera de las 2 tablas, pero en este caso debido a que a partir de la entidad Alumno recuperaremos la información asociada de su Domicilio, lo más conveniente es colocar la llave foránea del Domicilio en la tabla del Alumno.

Este es un ejemplo que utilizaremos más adelante para mapear esta asociación utilizando Hibernate o JPA.

EJEMPLO DE RELACIÓN DE 1 A MUCHOS

Ejemplo de relación 1 a Muchos. Un Instructor tiene muchos Cursos.



Vamos a revisar ahora una relación 1 a Muchos (1 a *).

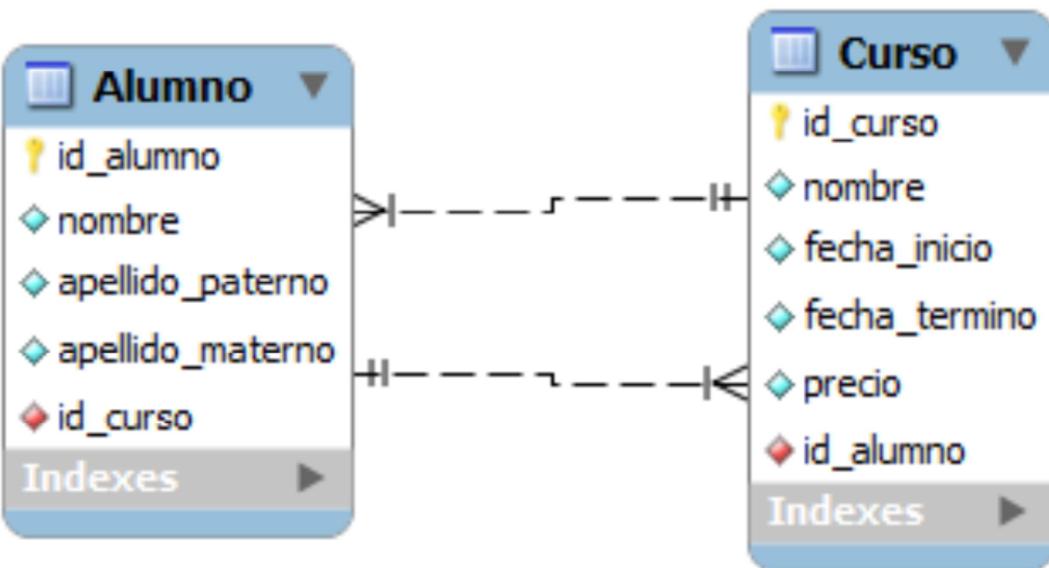
En este caso, estamos ejemplificando que un instructor puede impartir muchos cursos, entonces debido a que el instructor puede impartir uno o más cursos, agregamos el campo de **id_instructor** como llave foránea en la tabla de Cursos.

Entonces, a partir de la llave foránea del Instructor vamos a poder solicitar todos los cursos que este impartiendo dicho instructor.

Este es una de las relaciones más utilizadas en el diseño de relaciones en bases de datos, por ello es de vital importancia aplicar correctamente los conceptos de mapeo que veremos más adelante.

EJEMPLO DE RELACIÓN DE MUCHOS A MUCHOS

Ejemplo de relación Muchos a Muchos. Un Alumno tiene muchos Cursos, y un Curso tiene muchos Alumnos.



A continuación observamos un ejemplo de muchos a muchos (* a *).

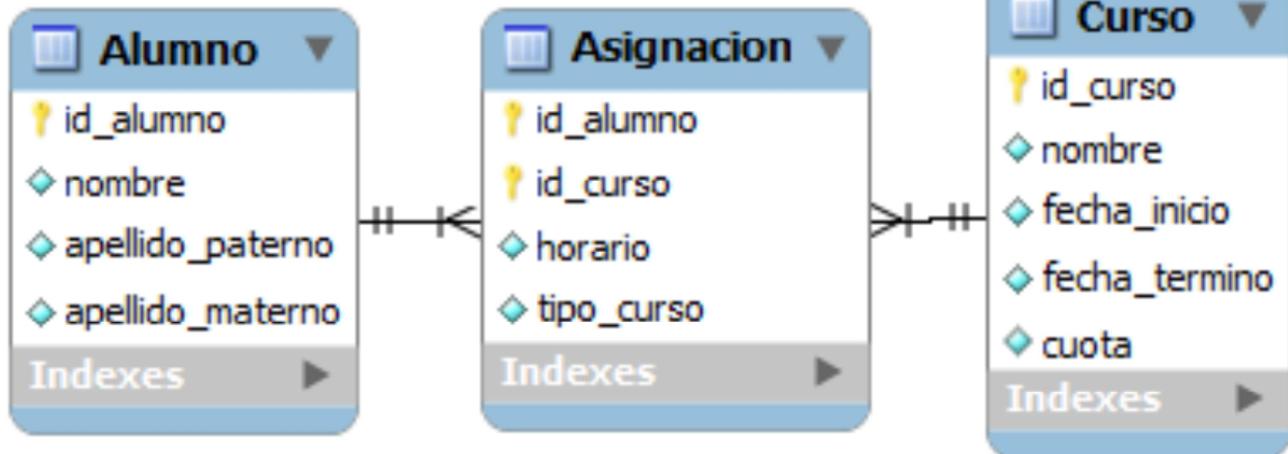
Aquí podemos observar que un Alumno puede estar relacionado a un Curso, pero a su vez un Curso puede tener muchos Alumnos.

Este tipo de relaciones Muchos a Muchos se pueden representar también con Hibernate o JPA, sin embargo este tipo de relaciones es mejor aplicar el concepto de normalización de base de datos para simplificar este tipo de relaciones y en lugar de tener relaciones muchos a muchos, tengamos relaciones Uno a Muchos o Uno a Uno.

Si no normalizamos este tipo de relaciones podemos caer en referencias circulares, debido a que un Alumno puede tener un Curso, pero un Curso puede tener a su vez al mismo Alumno. Para corregir esto vamos a normalizar (simplificar) esta relación, para que quede como una relación de Uno a muchos o de Uno a Uno. Veamos como hacer esto.

NORMALIZACIÓN DE RELACIÓN DE MUCHOS A MUCHOS

Ejemplo de normalización de relación Muchos a Muchos a una relación de Uno a Muchos. Un Alumno tiene una asignación y un curso tiene a un alumno asignado. Asignación es la tabla transitiva utilizada para romper la relación de Muchos a Muchos y tener ahora solo relaciones de Uno a Muchos.



www.globalmentoring.com.mx

Podemos observar la normalización de la relación anterior, en este caso la tabla de Alumno que se relacionaba directamente con la tabla de Curso.

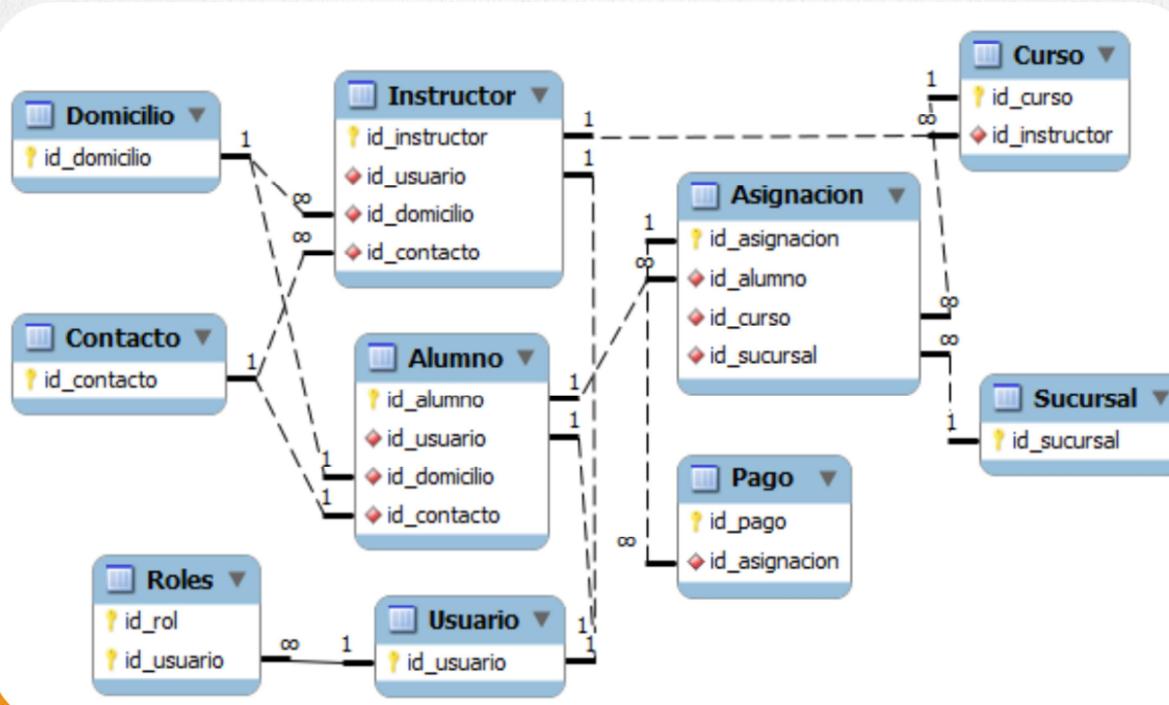
Ya no se relaciona directamente, si no que ahora hemos puesto una tabla intermedia, esta tabla intermedia se le conoce como tabla transitiva y lo que estamos haciendo es convertir la relación de Muchos a Muchos a una relación de Uno a Muchos de Alumnos hacia Asignación y una relación de Uno a Muchos de la tabla de Curso hacia la tabla Asignación.

Las llaves primarias de **id_alumno** y de **id_curso** las agregamos como llaves foráneas a la tabla de Asignación y a su vez combinándolas se convierte en una llave primaria compuesta para la tabla de Asignación, esas son algunas de las mejores prácticas que vamos a utilizar en Hibernate o JPA. Sin embargo también se puede agregar una llave primaria simple a la tabla de Asignación y así no manejar llaves primarias compuestas. Cualquiera de las dos opciones es factible.

A pesar de que pudimos haber manejado una relación de Muchos a Muchos directamente con Hibernate o JPA, normalizar este tipo de relaciones no permite administrar más fácilmente las relaciones de Uno a Muchos o Uno a Uno, y por lo tanto más sencillas de dar mantenimiento.

Además, una vez que hemos generado una tabla transitiva para el Alumno y el Curso, podemos agregar ciertos atributos de la nueva relación, por ejemplo un Alumno al estar tomando un Curso puede estar en cierto horario, entre otro tipo de atributos. En los ejercicios de mapeo de relaciones veremos cada uno de los casos estudiados hasta el momento.

REVISIÓN DEL ESQUEMA ENTIDAD – RELACIÓN COMPLETO



Observar el esquema de entidad relación del sistema SGA (Sistema de Gestión de Alumnos), que es la base de datos que vamos a estar trabajando a lo largo del curso. Este esquema entidad-relación tiene las relaciones más comunes que nos encontraremos al momento de mapear clases de Entidad utilizando Hibernate o JPA.

Tenemos una tabla de Alumno, la cual se relaciona con una tabla de Contacto, en el cual se van a guardar los datos de teléfono, celular, email o cualquier otro campo de para poder contactar al Alumno.

Podemos observar una tabla de Domicilio, la cual guardará los datos de domicilio del Alumno, esta relación es de tipo Uno a Uno. Un Alumno puede tener únicamente un Domicilio y un Alumno puede tener únicamente un registro de Contacto. Este registro de Contacto según hemos comentado pueden ser teléfonos o correos electrónicos, por otro lado un Alumno también puede tener asignados ciertos Cursos y un Curso tiene varios Alumnos, para ello hemos creado la tabla de Asignación, según comentamos anteriormente cuando desnormalizamos la relación de Muchos a Muchos entre Alumno y Curso, podemos observar que en esta tabla transitiva de Asignación tenemos la relación de un Alumno hacia muchas Asignaciones y de un Curso también hacia muchas Asignaciones y a su vez la tabla de Asignación tiene la ventaja de tener más atributos que representen a la relación entre Alumno y Curso, como puede ser que una asignación puede estar siendo impartida en cierta sucursal o campus distinto, entre otro tipo de atributos.

Este esquema de Entidad-Relación únicamente tiene las llaves primarias, no tiene los campos de cada una de estas tablas debido a que sería muy extenso mostrarlo en esta figura, pero se agrega el diagrama de entidad completo como parte de los ejercicios a resolver en esta lección.

Continuando con las relaciones, observamos que un Curso también es impartido por un Instructor y en este caso este Instructor también tiene asociado un Usuario. La tabla de Usuario y de Rol nos van a permitir tener los datos para el acceso al sistema SGA, por ello en esta tabla de Usuarios vamos a tener el username y el password y además le vamos asignar un Rol para establecer los permisos y las páginas o recursos que puede administrar este usuario.

A su vez una Asignación a un curso puede tener como un atributo el costo del curso en específico, pero también este pago podría ser en parcialidades y por lo tanto una Asignación puede tener muchos Pagos asociados, hay que recordar que una Asignación se hace única para un Alumno, para un Curso y para una Sucursal.

Este es el esquema entidad relación que vamos a estar trabajando a lo largo de este curso y es importante analizarlo y estudiarlo a detalle debido a que las relaciones de Hibernate o JPA que estemos estudiando van a estar basadas en la relaciones de este esquema entidad relación.

Descarga el esquema de Entidad-Relación completo aquí:

<http://icursos.net/cursos/Hibernate/leccion04/EsquemaEntidadRelacionSGAFull.png>

EJERCICIOS CURSO HIBERNATE Y JPA

- **ABRIR LOS ARCHIVOS DE EJERCICIOS EN PDF.**
- **EJERCICIO:** Ejercicio Mapeo de Asociaciones en Hibernate o JPA.

CURSO DE HIBERNATE Y JPA

www.globalmentoring.com.mx

CURSO ONLINE

HIBERNATE & JPA

Por: Ing. Ubaldo Acosta



CURSO DE HIBERNATE Y JPA

www.globalmentoring.com.mx

En Global Mentoring promovemos la Pasión por la Tecnología Java. Te invitamos a visitar nuestro sitio Web donde encontrarás cursos Java Online desde Niveles Básicos, Intermedios y Avanzados, y así te conviertas en un experto programador Java.

Además agregamos nuevos cursos para que continúes con tu preparación como programador Java profesional. A continuación te presentamos nuestro listado de cursos:

- | | |
|--------------------------|-------------------------------------|
| ✓ Lógica de Programación | ✓ Hibernate Framework |
| ✓ Fundamentos de Java | ✓ Spring Framework |
| ✓ Programación con Java | ✓ JavaServer Faces |
| ✓ Java con JDBC | ✓ Java EE (EJB, JPA y Web Services) |
| ✓ HTML, CSS y JavaScript | ✓ JBoss Administration |
| ✓ Servlets y JSP's | ✓ Android con Java |
| ✓ Struts Framework | ✓ HTML5 y CSS3 |

Datos de Contacto:

Sitio Web: www.globalmentoring.com.mx

Email: informes@globalmentoring.com.mx

