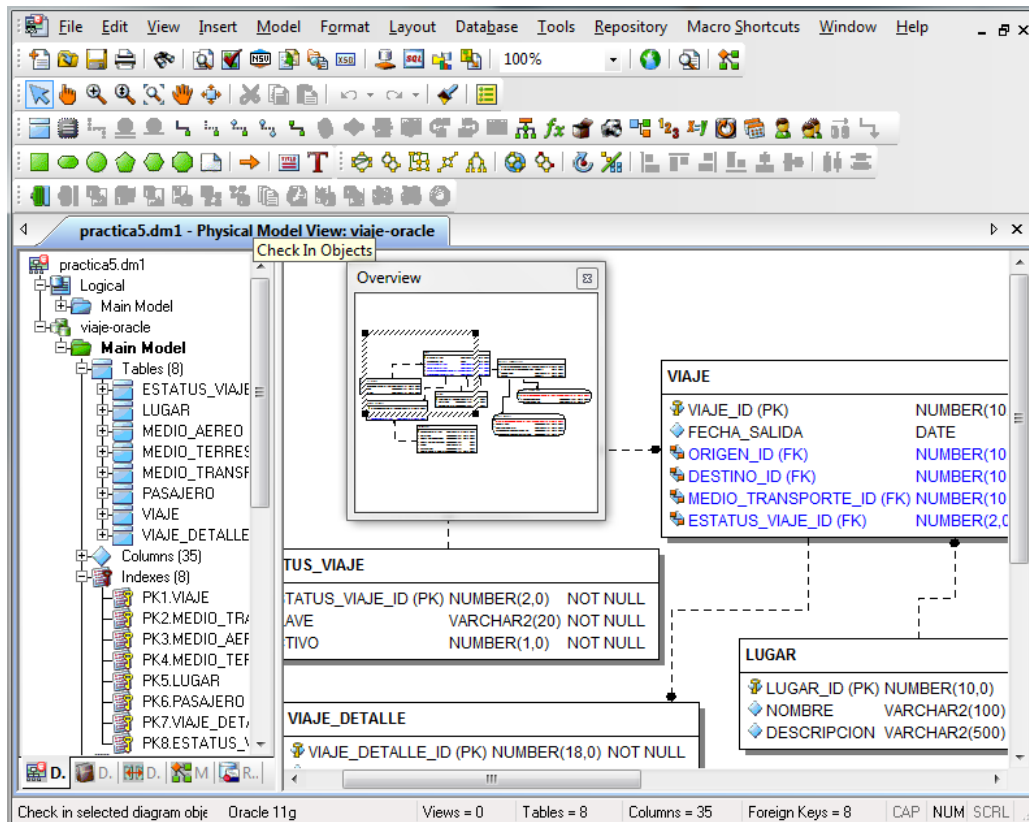


PRÁCTICA 4 - PREVIO
DISEÑO DE MODELOS RELACIONALES

1.1. GENERACIÓN DE MODELOS RELACIONALES CON ER STUDIO

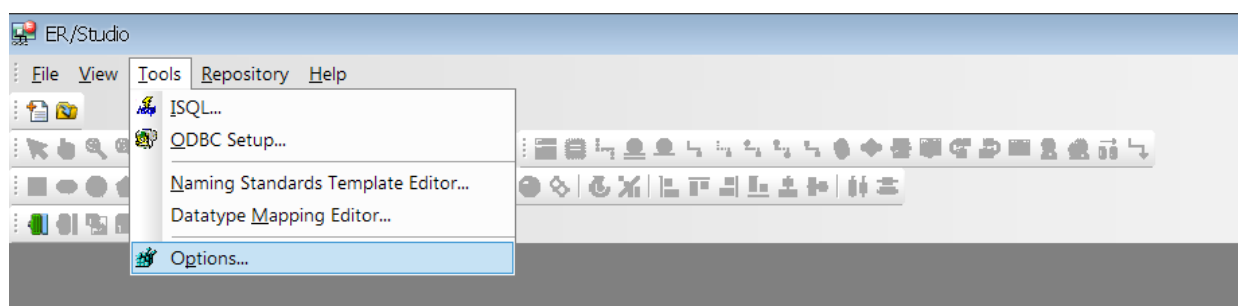
ER Studio es una herramienta que permite realizar el modelado de datos empleando varias de las notaciones vistas en clase. Su uso es sencillo y permite generar varios modelos físicos para diversos RDBMS como Oracle, DB2, PostgreSQL, etc., a partir de un único modelo lógico.



Para realizar el modelado de datos se hará uso de ER Studio V8. A continuación, se presenta una breve descripción del uso de la herramienta para generar modelos de datos.

1.1.1. Personalizar el editor

Antes de iniciar con el editor, es conveniente realizar algunos ajustes a la apariencia, así como algunos valores por default, principalmente para homologar las vistas de los diagramas generados. Para realizar esta personalización, Abrir ER Studio, (**no crear un nuevo modelo**), seleccionar Tools -> Options.

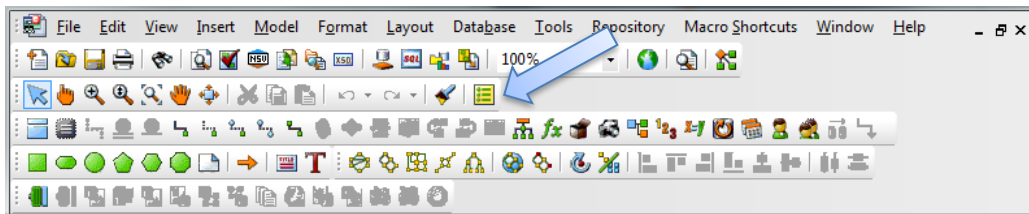


Aparecerá una ventana de opciones. Aplicar las configuraciones como se muestra en la imagen, pestaña Logical:

En la pestaña Physical, aplicar los siguientes cambios:

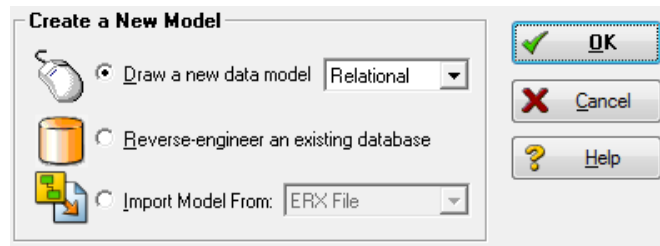
En la pestaña Display, seleccionar las opciones como se indica en la siguiente imagen:

Finalmente, en caso de requerir modificar la notación para un diagrama en particular y no a nivel global como se indicó en las imágenes anteriores, se deberá seleccionar la opción View -> Diagram and object display options y posteriormente seleccionar la pestaña Relationships. Aparecerá una pantalla en la que se podrá modificar la notación solo para el diagrama en turno. En su defecto, seleccionar el icono como se muestra en la siguiente figura:



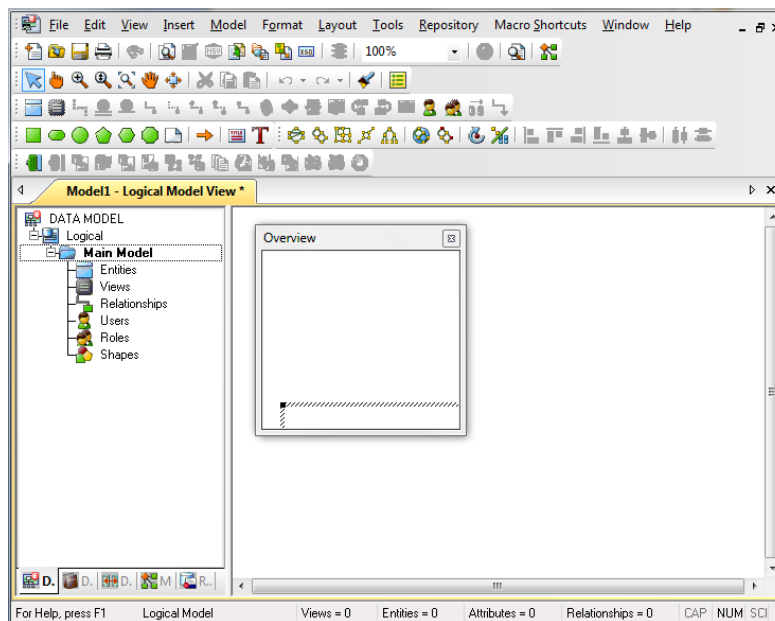
1.1.2. Creación de un nuevo modelo.

ER Studio permite la creación de un nuevo modelo de datos a partir de 3 fuentes distintas. Seleccionar `file -> new`, y seleccione la opción como se muestra en la figura.



Observar que es posible crear un modelo a partir de una base de datos existente, o realizar la importación de un modelo de ERWin.

1.1.3. Agregando objetos al modelo lógico.



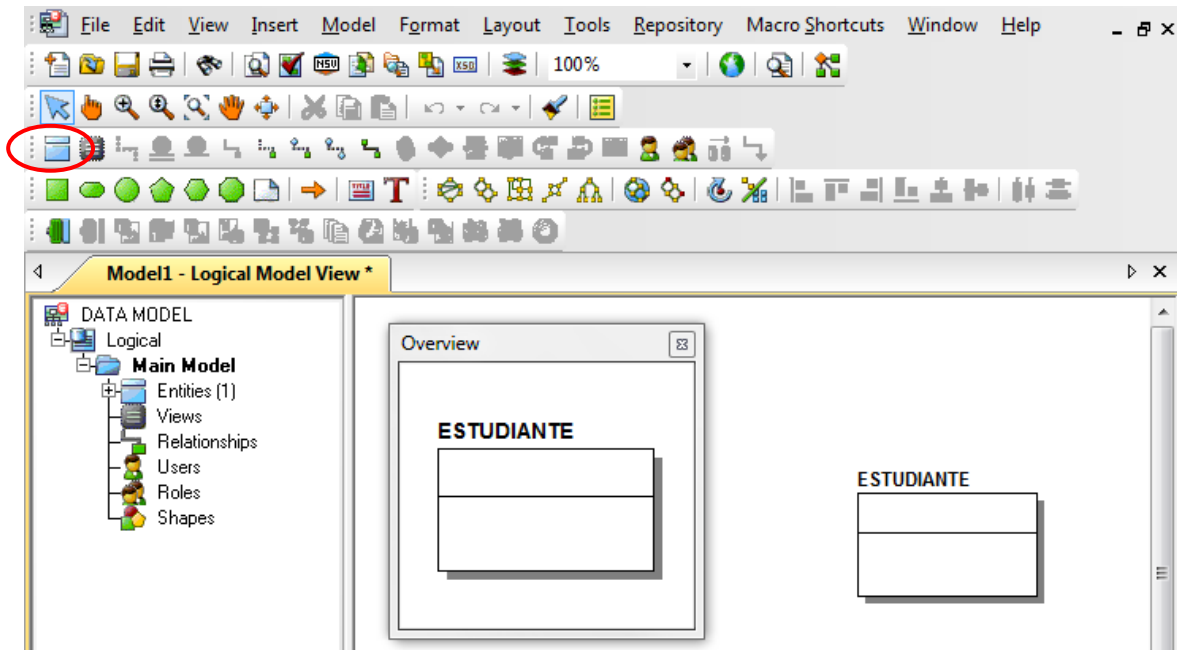
Al crear el nuevo modelo, por default se crea un modelo lógico (independiente del manejador). Se muestra del lado izquierdo los elementos del modelo: entidades, vistas, relaciones, usuarios.

1.1.4. Elementos del modelo lógico.

Para iniciar con la creación de los elementos del modelo de datos, se emplea la barra de herramientas

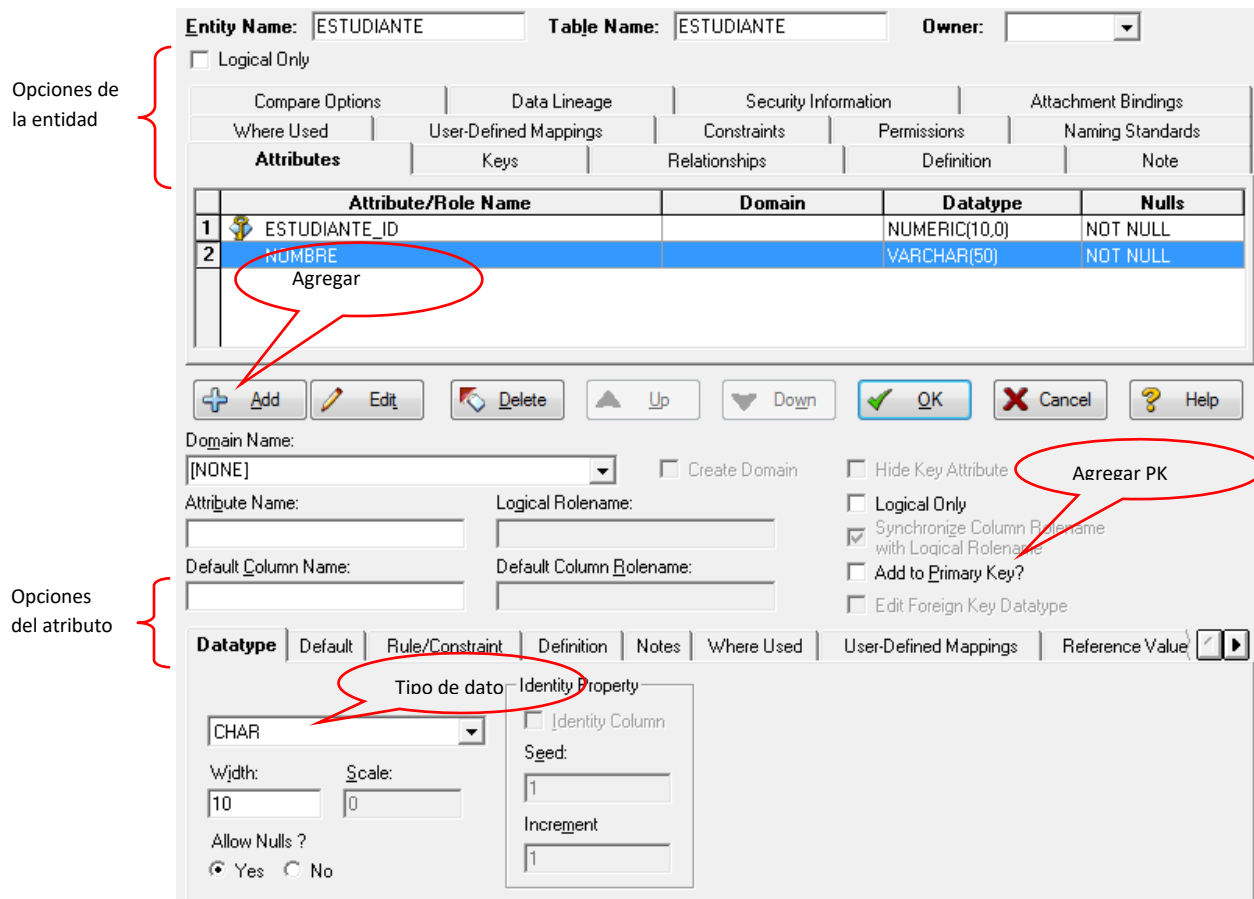
1.1.4.1. Creación de entidades.

Seleccionar `insert -> Entity`, o hacer clic en el icono que se muestra a continuación



1.1.4.2. Atributos de una entidad.

Para agregar los atributos a una entidad, seleccionar la entidad en el editor, dar doble clic, ó clic derecho -> edit entity. En la pantalla que aparece, seleccionar la opción Add.



Observe que el combo que muestra los tipos de datos, estos corresponden a los tipos de datos que define el estándar SQL, esto para un modelo lógico.

Al terminar de agregar los atributos de la entidad, el resultado es un diagrama como el siguiente:

Nombre de la tabla

Llave primaria

NULL o NOT NULL option

Tipo de dato

SOLICITUD		
SOLICITUD_ID	NUMERIC(10,0)	NOT NULL
FECHA_SOLICITUD	DATETIME	NOT NULL
NUM_CANDIDATOS	NUMERIC(5,0)	NOT NULL
FECHA_INICIO	DATE	NOT NULL
PUESTO	VARCHAR(40)	NOT NULL
COMPANIA_ID (FK)	NUMERIC(10,0)	NOT NULL
CERTIFICACION_ID (FK)	NUMERIC(5,0)	NOT NULL

1.1.4.3. Documentación de entidades y atributos.

Es muy importante que el modelo este bien documentado, tanto a nivel tabla como a nivel atributo. Esta documentación permitirá entre otras cosas transmitir el conocimiento y entendimiento del modelo. En ER studio se agrega la documentación seleccionando la pestaña **definition**.

Entity Name: Entity6 Table Name: PASAJERO Owner: [dropdown]

☐ Physical Only ☐ Do Not Generate

Permissions | PreSQL & PostSQL | Naming Standards | Compare Options | Data Lineage | Security Information | Attachment Bindings

User-Defined Mappings | Storage | Partitions | Overflow | Constraints | Dependencies | Capacity Planning

Columns | DDL | Indexes | Foreign Keys | **Definition** | Note | Where Used

You can optionally provide a definition for an entity. ER/Studio adds the definition as a table comment when generating SQL code, if the target database supports it.

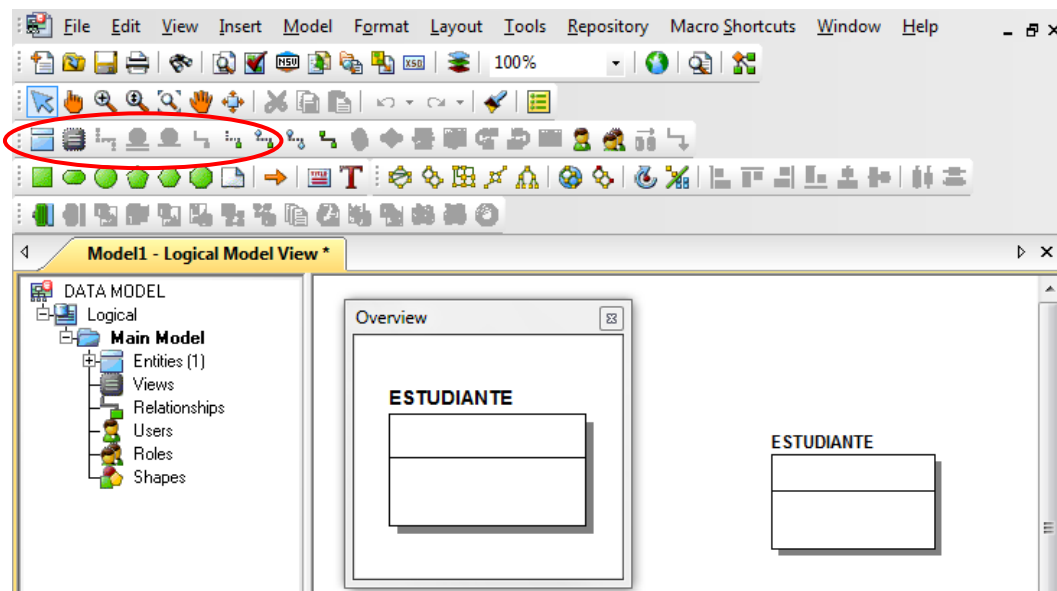
En esta entidad se almacenan los atributos de un Pasajero requeridos por la Agencia de viajes para programar su itinerario.

OK Cancel Help

1.1.4.4. Asociación de entidades empleando relaciones

La asociación se realiza a partir de los tipos de relaciones vistos en clase:

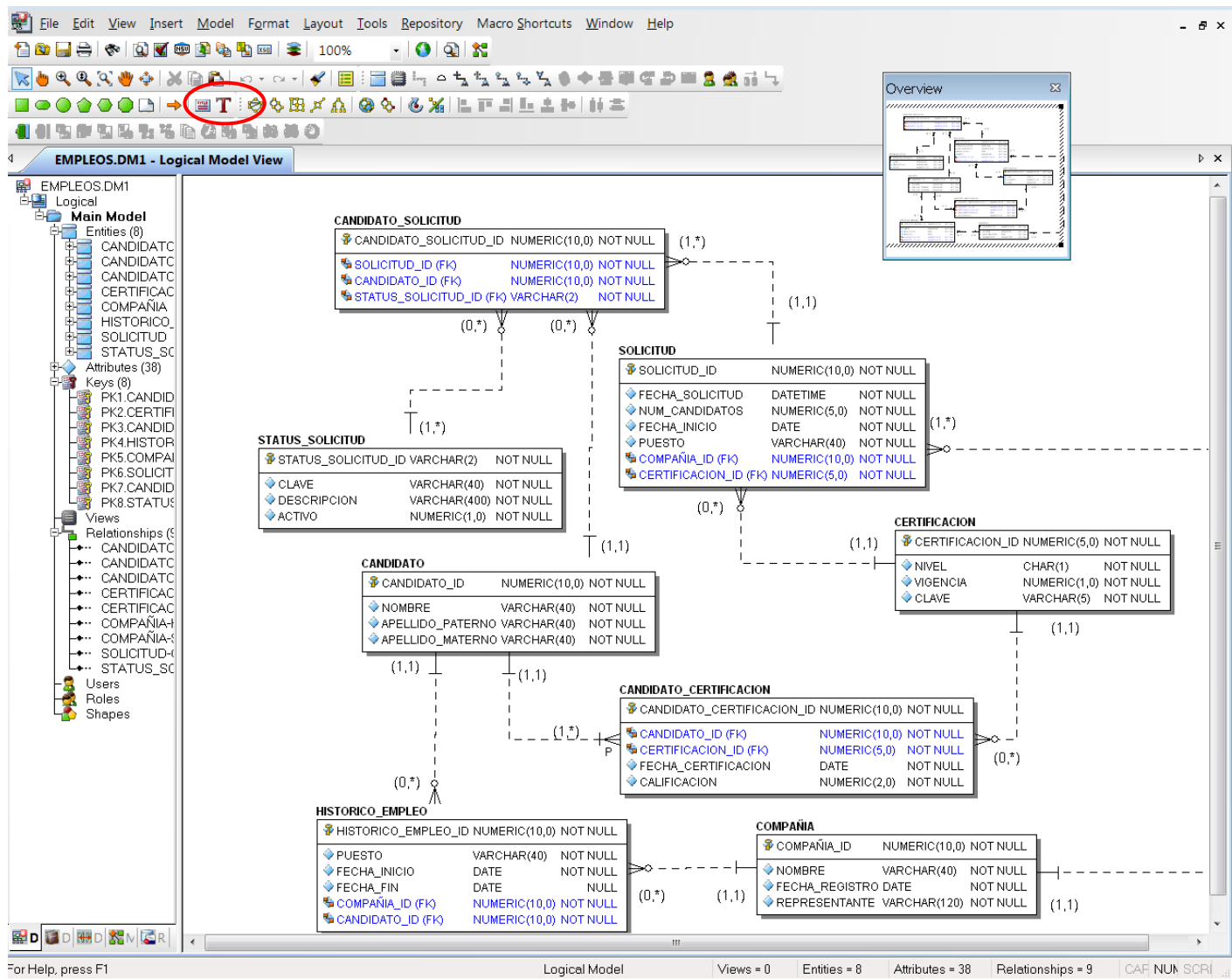
- Relaciones identificativas
- Relaciones no identificativas obligatorias
- Relaciones no identificativas opcionales
- Subtipos



Para asociar 2 entidades:

- I. se hace clic en el icono que representa el tipo de relación a usar.
- II. Se hace clic sobre la entidad origen
- III. Se hace clic sobre la entidad destino (la tabla que contendrá la FK).

Al final, se obtiene un modelo como el siguiente:



Observar que los valores de la cardinalidad se deben escribir de forma manual, empleando la herramienta de texto mostrada en la imagen.

1.1.4.5. Opciones para relaciones

Al hacer doble clic en la línea que representa la relación, aparecerá una ventana como la siguiente, en la que se puede personalizar o modificar las propiedades de la relación.

Parent Entity: ESTATUS_VIAJE
Parent Key(s): ESTATUS_VIAJE_ID
Child Entity: VIAJE
Child Key(s): ESTATUS_VIAJE_ID

Properties | Phrases | Name | Trigger | Definition | Note | **RoleName** | Compare Options

Parent Key: Primary Key

Relationship Type
☐ Identifying
☒ Non-Identifying
☐ Non-Specific

Existence
☐ Optional
☒ Mandatory

Cardinality
☒ One to Zero or More
☐ One to One or More (P)
☐ One to Zero or One (Z)
☐ One to Exactly: 0

☐ Physical Only ☐ Do Not Generate
☐ Defferable ☐ Initially Deferred

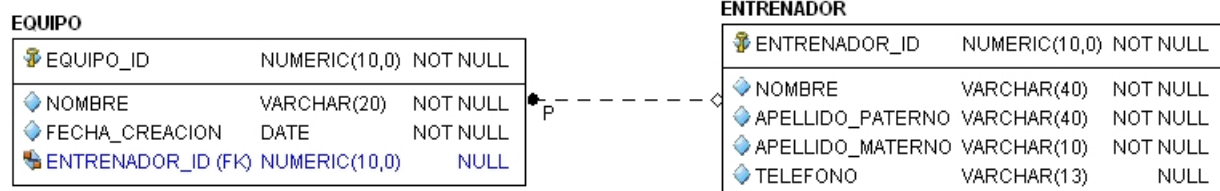
OK Cancel Help

Algunas de las opciones que se pueden modificar son:

- Tipo de relación
- Cardinalidad
- Rol name (cambio del nombre de la FK)

1.1.4.6. Cambio del nombre a la FK (ver flecha de la imagen anterior)

En algunas situaciones es conveniente cambiar el nombre a la FK para tener una mayor comprensión del modelo. Ejemplo:



Suponer que se desea cambiar el nombre de la FK: ENTRENADOR_ID A DIRECTOR_TECNICO_ID.

Para realizar el cambio, seleccionar la pestaña "Role Name" mostrada en la figura anterior, aparecerá una pantalla como la siguiente:

Parent Entity: ENTRENADOR
Parent Key(s): ENTRENADOR_ID
Child Entity: EQUIPO
Child Key(s): ENTRENADOR_ID

Properties | Phrases | Name | Trigger | Definition | Note | **RoleName** | Compare Options

Key Name	Logical Rolename	Default Column Name	Default Column Rolename
ENTRENAD...	ENTRENADOR_ID		

☐ Logical Only

OK Cancel Help

Para cambiar el nombre, dar doble clic al registro que aparece en la pantalla anterior, aparecerá una nueva pantalla en la que se captura el nombre de la FK:

Normally, a foreign key attribute assumes the name as its associated primary key. If you wish to name the foreign key differently, then you must specify a rolename below.

Attribute Name (Read-Only):
ENTRENADOR_ID

Logical Rolename:
DIRECTOR_TECNICO_ID

Default Column Name:
DIRECTOR_TECNICO_ID

Default Column Rolename:
DIRECTOR_TECNICO_ID

☒ Synchronize Column Rolename with Logical Rolename

OK Cancel Help

Para el caso de relaciones unarias, la herramienta por default, presenta una pantalla para cambiar el nombre al momento de establecer la FK hacia la misma tabla:

A recursive relationship occurs when a non-identifying relationship propagates the keys from the child entity back into itself.

If you wish to duplicate the keys in the entity you must specify a rolename for each attribute.

Parent Entity: JUGADOR

Child Entity: JUGADOR

Attribute Name (Read-Only):
JUGADOR_ID

Logical Rolename:
LIDER_GRO_ID

Default Column Name:
LIDER_GRO_ID

Default Column Rolename:
LIDER_GRO_ID

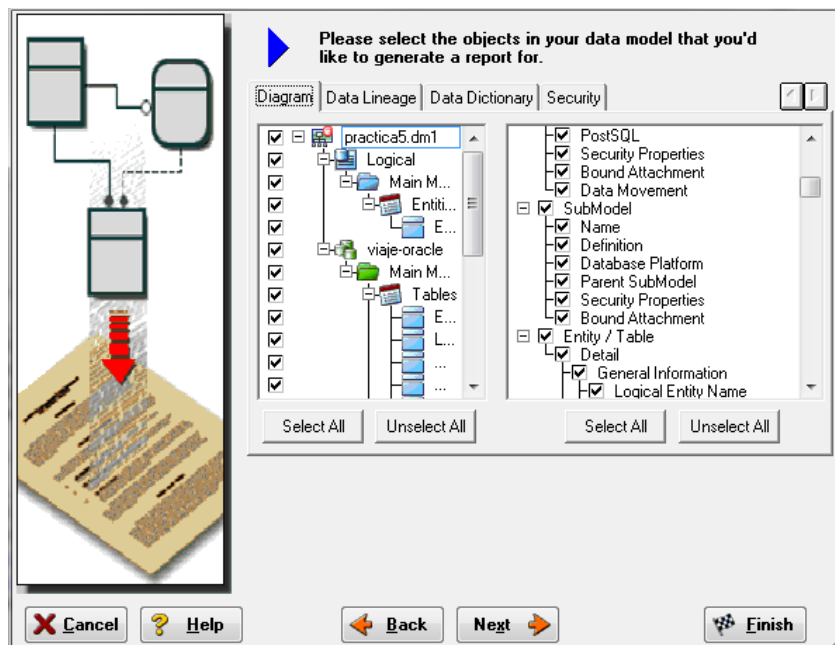
☒ Synchronize Column Rolename with Logical Rolename

OK Cancel Help

1.1.5. Generación de reportes.

Con ER estudio es posible crear reportes en formato RTF o en formato HTML que describen de forma detallada todas las opciones del modelo. Para generar el reporte, seleccionar Tools -> Generate Reports.

- Seleccionar la creación de un reporte HTML, seguir las indicaciones en pantalla.
- En el segundo paso, seleccionar las pestañas Diagram y Data Dictionary, seleccionar todas las opciones.



Al final del proceso, el reporte HTML que se genera es el siguiente:

The screenshot displays the Embarcadero ER/Studio Web Report Generation interface. On the left, a tree view shows the database structure, including entities like PASAJERO, VIAJE, and VIAJE_DETALLE. The main panel shows the details for the PASAJERO entity, including its logical name, default table name, and a list of columns with their domains and datatypes.

PASAJERO

Logical Entity Name	Entity6
Default Table Name	PASAJERO
Oracle Temporary Table	NO
Physical Only	NO
Do Not Generate	False
Owner	
Definition	En esta entidad se almacenan los atributos de un Pasajero requeridos por la Agencia de viajes para programar su itinerario.
Note	
PreSQL	
DDL Code	<pre>CREATE TABLE PASAJERO (PASAJERO_ID NUMBER(10, 0) NOT NULL, NOMBRE VARCHAR2(50) NOT NULL, APELLIDO_PATERNO VARCHAR2(50) NOT NULL, APELLIDO_MATERNO VARCHAR2(50), EDAD NUMBER(3, 0), TELEFONO VARCHAR2(20), NUM_IDENTIFICACION VARCHAR2(18) NOT NULL, CONSTRAINT PK6 PRIMARY KEY (PASAJERO_ID) USING INDEX LOGGING); LOGGING ;</pre>
PostSQL	

PASAJERO Columns

ColumnName	Domain	Datatype	NULL	Definition
PASAJERO_ID		NUMBER(10, 0)	NO	
NOMBRE		VARCHAR2(50)	NO	
APELLIDO_PATERNO		VARCHAR2(50)	NO	
APELLIDO_MATERNO		VARCHAR2(50)	YES	
EDAD		NUMBER(3, 0)	YES	

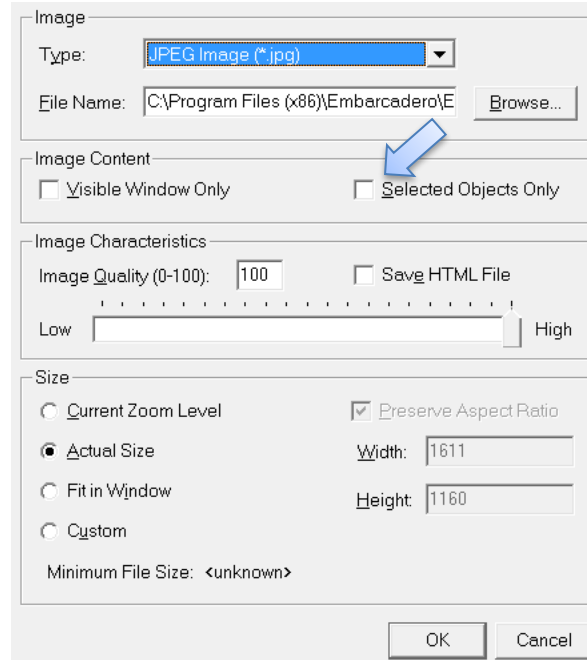
1.1.6. Exportando diagramas

ER studio permite exportar un diagrama en diferentes formatos. Uno de ellos es a través de una imagen. Esta última opción es útil sobre todo para efectos del reporte de la práctica. Para realizar este proceso seleccionar File->Export Image. Aparecerá una pantalla similar a la siguiente.

Observar que es posible seleccionar un conjunto de tablas (tal vez no se desea generar una imagen con todo el diagrama) y generar un diagrama solo para las tablas seleccionadas haciendo clic en la opción "Selected Objects Only".

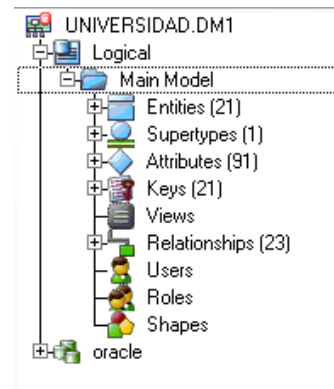
Para efectos del reporte se recomienda emplear esta funcionalidad de tal forma que los diagramas que se incluirán en el reporte sean claros y fáciles de leer.

Se recomienda adicionalmente, no dejar demasiado espacio entre tablas para que el tamaño del todos los objetos sea lo más grande posible.



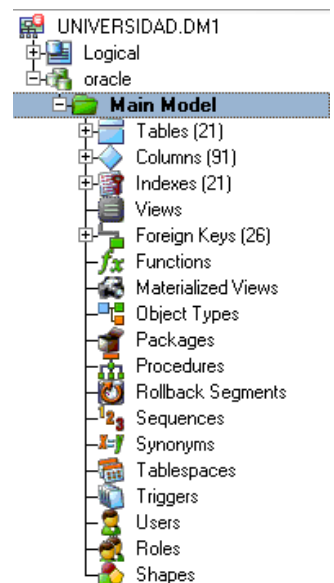
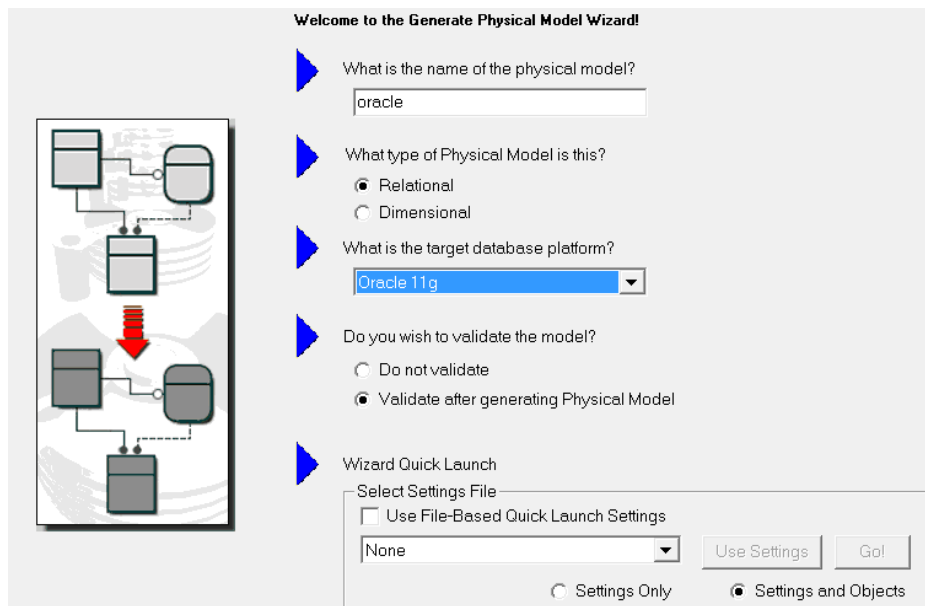
1.1.7. Generando modelos físicos

Cuando se inicia la construcción de un modelo de datos se prefiere empezar por la creación de un **modelo lógico**, es decir, un modelo que es totalmente independiente del manejador. Este diagrama emplea únicamente las definiciones del estándar SQL. En la siguiente imagen se muestra un ejemplo de un modelo lógico, observar el color azul de la carpeta.



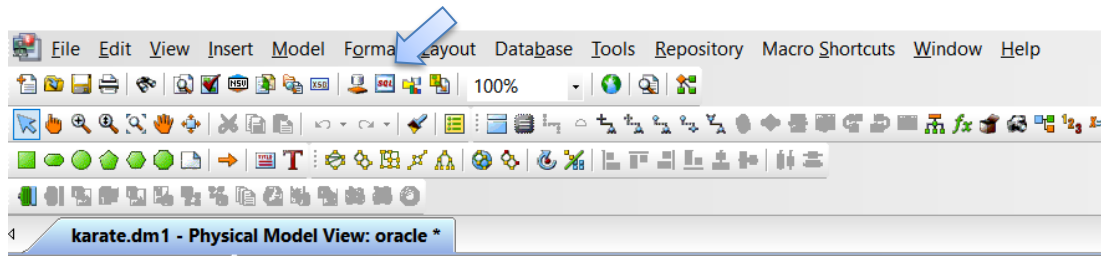
Una vez que se ha concluido con la creación, revisión y validación exitosa del modelo lógico, es posible crear un **modelo físico**.

El modelo físico se genera de manera automática para un RDBMS en particular. En ER studio se puede generar haciendo clic derecho en el ícono "Logical" -> "Generate physical Model". Aparecerá una pantalla similar a la siguiente en la que se configuran las opciones para generar el modelo. . Observar la creación del nuevo modelo empleado una estructura de color verde.

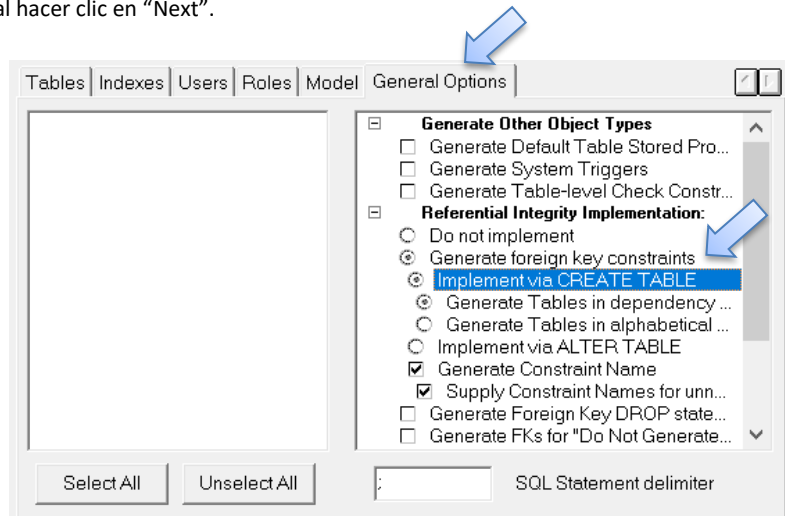


Uno de los principales usos de un modelo físico es que ER Studio puede generar el código SQL para crear la estructura de todos los objetos de una base de datos.

Para generar el código SQL seleccionar la opción que se muestra en la siguiente figura:



Al seleccionar la opción anterior (Generación de código SQL), aparecerá una serie de opciones empleadas para personalizar la generación de código SQL. En general, todas las opciones se pueden dejar con sus valores por default, excepto las opciones indicadas en la siguiente figura que corresponden a la segunda pantalla que se muestra al hacer clic en “Next”.



En la imagen anterior se han modificado 2 opciones:

- Generate foreign key Constraints:
 - I. Implement via CREATE TABLE. Permite crear las restricciones de referencia dentro de la definición de la tabla en lugar de crearlas empleando la instrucción ALTER TABLE.
 - II. Generate tables in dependency Order. Permite crear las tablas en el orden correcto de tal forma que primero sean creadas las tablas “padre” y después las tablas dependientes.

Finalmente, para mostrar el código SQL, hacer clic en “SQL Preview”.