

## TABLE INSTANCE CHART

Table Name:

Column Name							
Key Type							
Nulls/Unique							
Sample Data							

## TRANSLATING ENTITIES TO TABLES

### Data Modeling and Database Design

STEPS	DETAILS
Map Entity to Table	<ul style="list-style-type: none"> <li>Create a table for each entity in the ERD</li> <li>The plural of the entity name is used as the table name. For example, EMPLOYEE becomes EMPLOYEES</li> </ul>
Map attributes to columns	<ul style="list-style-type: none"> <li>Attribute names become column names. Within the same table no two columns can have the same name</li> <li>Remove the spaces in attribute names and replace them with an underscore. For example Start Date becomes Start_date</li> <li>Determine the datatype and correct size for each data attribute. For example First_name VARCHAR2 (15)</li> <li>An attribute with a # sign becomes a primary key ( PK) or unique key ( U1)</li> <li>Determine any other constraints that apply to this column. For example Salary must t be &gt;0, email must be Not Null - NN</li> </ul>
Map relationship to foreign keys	<ul style="list-style-type: none"> <li>Identify the foreign key (FK) column(s) by examining the relationships between entities. A relationship relates one or more foreign key columns in the table at the "many" side of the relationship. If the relationship is required, the FK is mandatory otherwise it is optional.</li> <li><b>1 : M relationships</b> - place the parent column as a foreign key column in the child (many side) table. If the "parent - child" relationship is mandatory, the child FK column must be NN. If the relationship is optional, the child FK column can be null. (pdf 7-14) All of the following are rare: <ul style="list-style-type: none"> <li><b>All 1 : 1 relationships</b> - create a foreign key and a unique key. All columns of this foreign key are also part of a unique key or create a single table or intersection table instead of two separate tables. ( pdf 7-18)</li> <li><b>1 : 1 relationships optional one side/mandatory one side</b> - place the foreign key in the mandatory side table (pdf 7-18)</li> <li><b>1 : 1 relationships optional both sides or mandatory both sides</b> - place the foreign key in either table, if optional on both sides you can place the foreign key in the table with fewer numbers of rows to save space (pdf 7-18)</li> </ul> </li> <li><b>1 : M Barred relationship</b> is mapped into a foreign key in the child "many side" table. There may be many different child instances but each parent-child combination must be unique. (pdf 7-15 to 7-16)</li> <li><b>M : M relationship</b> - create and name an intersection table. The intersection table will have two foreign key columns created from the primary key columns of the two tables it intersects. These columns together form the primary key for the intersection table. The two intersection table columns are always mandatory</li> <li><b>Supertype</b> - create a table with columns for each attribute of the supertype (pdf 7-21)</li> <li><b>Subtypes</b> - create one table for each first level subtype; each subtype table will have all the columns from the supertype plus a column for each of its own attributes; all columns have the original optionality; the primary UID of supertype creates a PK in each of the subtype tables. If the subtypes have their own UID, it can be used as the basis for the subtype PK; supertype secondary identifiers become unique keys within each subtype table; for relationships at the subtype level, the FK is implemented in the table it is mapped to. (pdf 7-20)</li> <li><b>Arc-</b> the primary key of each parent is placed in the child table as a foreign key. A check constraint is used to verify that if one of the foreign keys is populated, the other must not be populated. (pdf 7-19)</li> </ul>

Map UIDs	<ul style="list-style-type: none"> <li>Identify entity attributes that have the # sign. These attributes will become a PK, U1</li> <li>Identify any entity that has relationships that have a UID bar. Those FK columns will also become PK, U1.</li> </ul>
A foreign key column will be unique only when:	<ul style="list-style-type: none"> <li>It's a PK also, i.e. the relationship has a UID bar (U1)</li> <li>It came from a 1:1 relationship (U2).</li> </ul>