

The logo for Oracle Academy. The word "ORACLE" is in a bold, orange, sans-serif font. Below it, the word "Academy" is in a smaller, dark gray, sans-serif font. The entire logo is centered on a light gray background, which is framed by dark gray horizontal bars at the top and bottom.

ORACLE

Academy

Database Design

6-4

Third Normal Form

ORACLE
Academy



Copyright © 2020, Oracle and/or its affiliates. All rights reserved.

Objectives

- This lesson covers the following objectives:
 - Identify transitive dependencies in a data model
 - Define the rule of Third Normal Form in the normalization process
 - Examine a non-normalized entity and determine which rule, or rules of normalization are being violated
 - Apply the rule of Third Normal Form to resolve a violation in the model

Purpose

- Your goal as a database designer is to "store information in one place only and in the best possible place"
- Following the rules of normalization helps you achieve this goal
- You may want to enter different kinds of information for a friend in your personal address book: phone number, address, name of school or place of work

Purpose

- If you have several friends who go to the same school, and you enter the school's street address along with each of them, you would not only be duplicating data but causing potential problems - for instance, if the school moved and changed its address, you would have to go back and change it everywhere!
- Normalization is a process to eliminate these kinds of problems



Third Normal Form Rule

- The rule of Third Normal Form (3NF) states that no non-UID attribute can be dependent on another non-UID attribute
- Third Normal Form prohibits transitive dependencies
- A transitive dependency exists when any attribute in an entity is dependent on any other non-UID attribute in that entity

Third Normal
Form Violation

CD
Id
* Title
* Producer
* Year
o Store name
o Store address

Third normal form: The form of database normalization where all non-key fields are dependent on the key, the whole key, and nothing but the key

Third Normal Form Rule

- Think of the kind of information you'd like to store about your CD collection
- Does information about the store where you bought the CD belong in the same entity?
- If the store address changed, you would have to change the information on all the CDs that were bought at that store

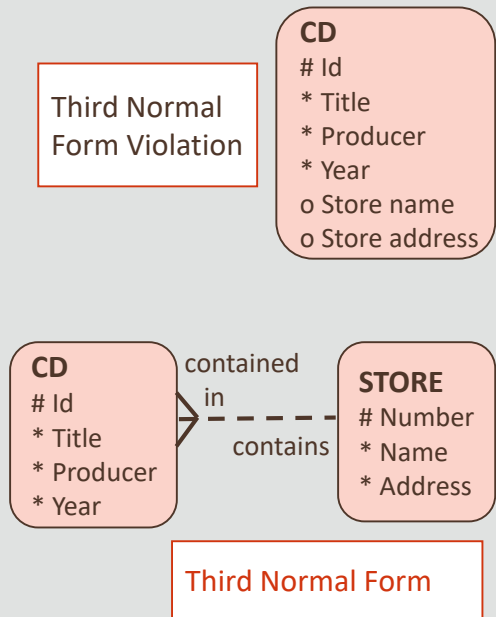
Third Normal
Form Violation

CD
Id
* Title
* Producer
* Year
o Store name
o Store address

To check if an entity is in Third Normal Form, examine each non-UID attribute, and check for Transitive Dependency with other non-UID attributes.

Third Normal Form Transitive Dependency

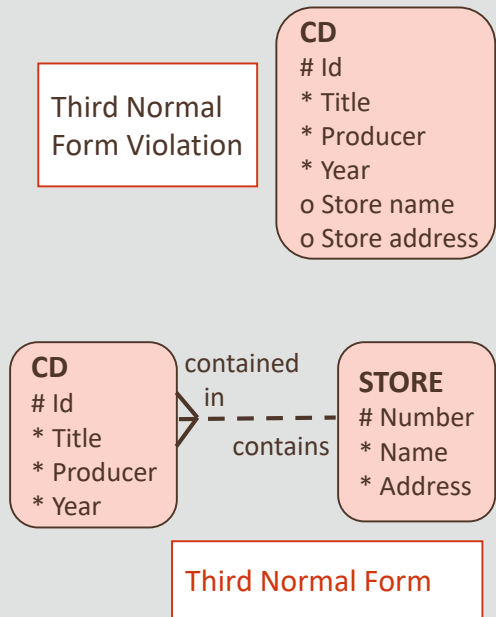
- The store address is dependent on the CD number, which is the UID of the CD entity, so this entity is in 1NF and 2NF
- But store address is also dependent on store name, which is a non-UID attribute
- This is an example of a transitive dependency and a violation of Third Normal Form



Third Normal Form Violations are hiding another entity.
For an entity to be in Third Normal Form, it must also be in Second Normal Form.

Third Normal Form Transitive Dependency

- The correctly normalized model is shown here: create a second entity **STORE**, with a relationship to **CD**



To resolve a Third Normal Form violation, you create a new Entity, move the attributes that were causing the violation to the new entity (adding a UID) , and draw a relationship to the original entity. In the second model, if the store address changed, you would have to change the information in only one instance of the STORE entity.

Third Normal Form Example

- Consider a system that tracks information about cities - size, population, mayor, and so on
- The first model shows an entity that includes state information
- Although state is an attribute of city, state flower is really an attribute of state

Third Normal
Form Violation

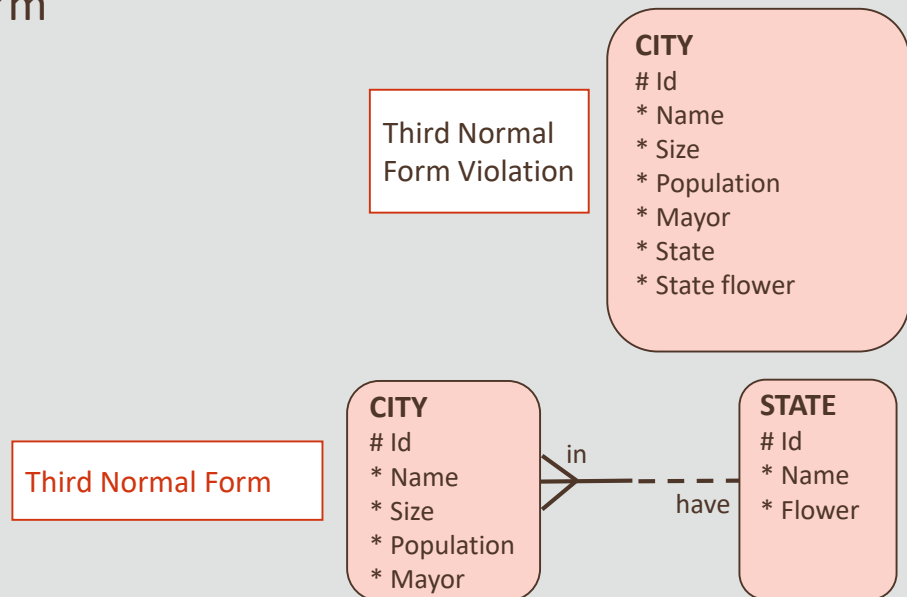
CITY

- # Id
- * Name
- * Size
- * Population
- * Mayor
- * State
- * State flower

Another way to look at the rule of third Normal Form is: Attributes can't have attributes of their own. In the first model, the attribute state has an attribute – state flower. This is a violation of Third Normal Form.

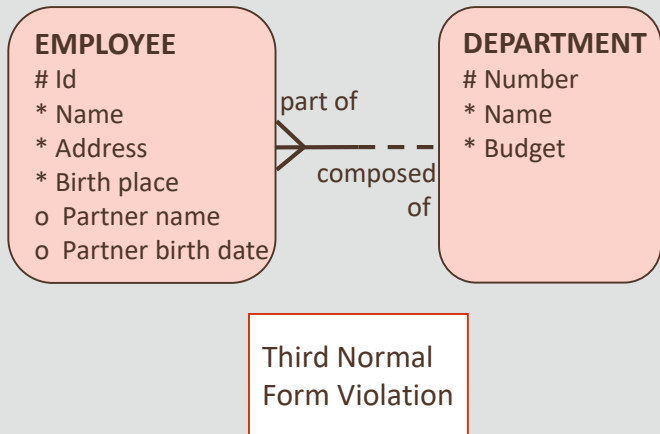
Third Normal Form Example

- The second model, with a new entity STATE, is in Third Normal Form



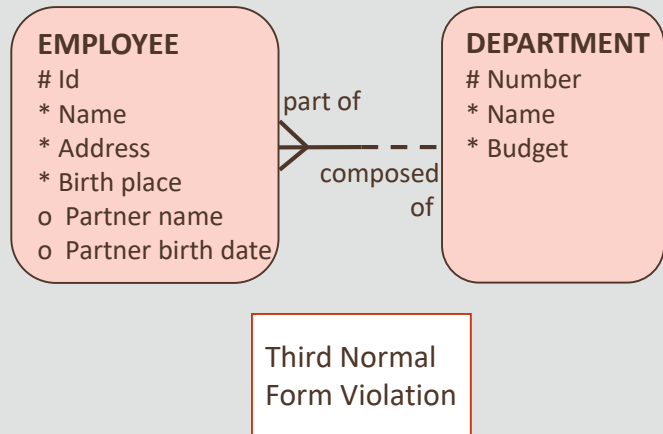
Third Normal Form Second Example

- In this example, assume the following business rule: each employee can have one partner
- This model violates Third Normal Form because partner birth date is an attribute of partner, not of EMPLOYEE



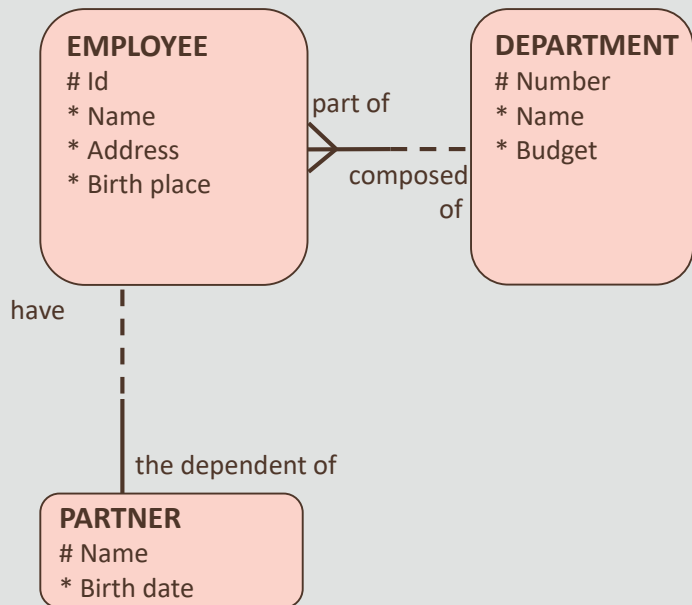
Third Normal Form Second Example

- Another way of stating Third Normal Form: non-UID attributes can't have attributes of their own



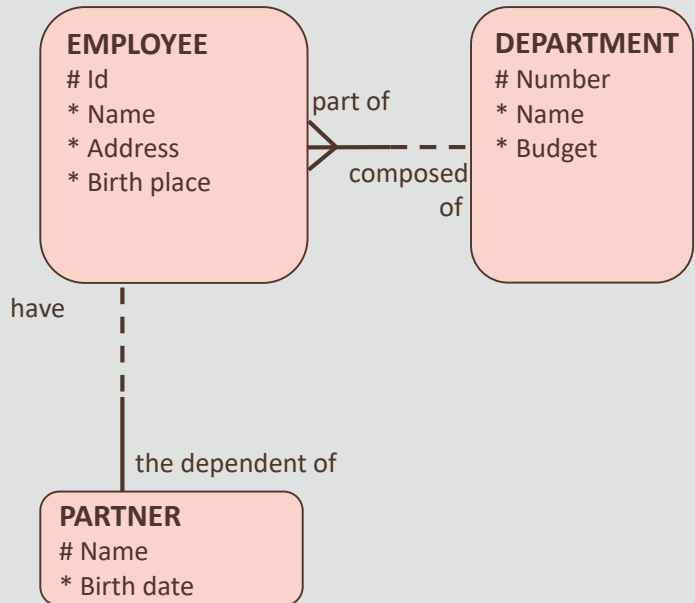
3NF Second Example Solution

- This model supports Third Normal Form because partner birth date is an attribute of partner, not of EMPLOYEE



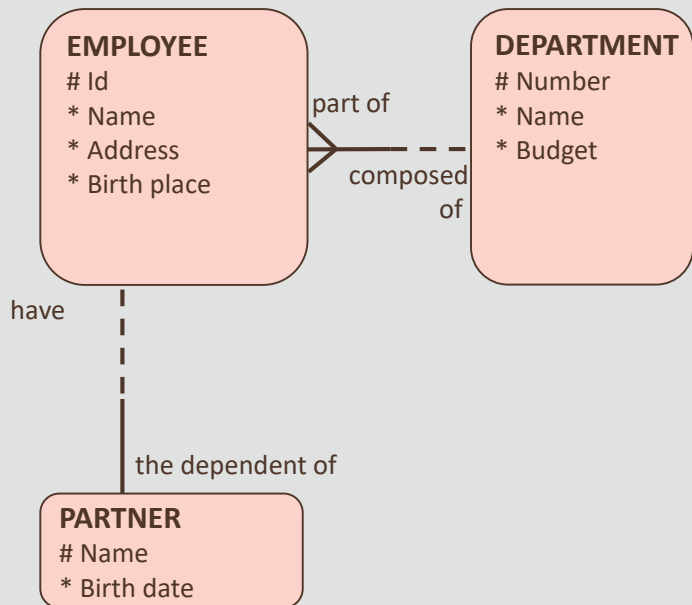
3NF Second Example Solution

- The 1:1 relationship is optional on the EMPLOYEE end because some employees will not have partners



3NF Second Example Solution

- It is mandatory on the PARTNER end because information about a partner is tracked only if that person is a partner of one and only one EMPLOYEE



Terminology

- Key terms used in this lesson included:
 - Third Normal Form (3NF)
 - Transitive dependency

Summary

- In this lesson, you should have learned how to:
 - Identify transitive dependencies in a data model
 - Define the rule of Third Normal Form in the normalization process
 - Examine a non-normalized entity and determine which rule, or rules of normalization are being violated
 - Apply the rule of Third Normal Form to resolve a violation in the model

The Oracle Academy logo is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by two horizontal dark gray bars, one at the top and one at the bottom.

ORACLE

Academy