Hibernate & Spring Data JPA

Beginner to Guru

Relational Database Principles

Relational Databases

- Relational Databases store data in tables which have relations to other tables.
- The Relational Model was first proposed by Edgar F. Codd in 1969.
- Nearly all modern Relational Databases use SQL for data definition and manipulation.
  - SQL - Structured Query Language
- Relational Databases are the most widely used database in the world.
  - Used in smart phones, desktops, automobiles, and highly popular in businesses

Database Table

- A database table is a lot like a spreadsheet.
- Data is kept in Columns and Rows.
- Each Column is assigned:
  - A Unique Name, identifying a human readable name of the column. (ie FIRST_NAME, LAST_NAME)
  - A Data Type (ie - String, Date, Time, Number, etc)
  - Optionally, constraints (ie - Is a value required?, Length of String, etc)
- Each Row is a distinct database Record.

Example Database Table

	A	B	C	D	E	F
1	**FIRST_NAME**	**LAST_NAME**	**ADDRESS**	**CITY**	**STATE**	**ZIP_CODE**
2	Michael	Weston	123 Brickel	Miami	FL	33135
3	Fiona	Glenanne	123 Brickel	Miami	FL	33135
4	Sam	Axe	222 Taimiami	Miami	FL	33109
5	Madeline	Weston	945 Sunset Lane	Miami	FL	33114
6	Jesse	Porter	9973 A1A	Miami	FL	33132
7	Barry	Burkowski	3838 Orange Grove St	Miami	FL	33314

Primary Key

- A Primary Key is an optional special database column or columns used to identify a database record. Note: Optional - but not recommended!
- Unique - There can be only one! Like Highlander!
- Could be FIRST_NAME, LAST_NAME, or FIRST_NAME and LAST_NAME.

	A	B	C	D	E	F
1	**FIRST_NAME**	**LAST_NAME**	**ADDRESS**	**CITY**	**STATE**	**ZIP_CODE**
2	Michael	Weston	123 Brickel	Miami	FL	33135
3	Fiona	Glenanne	123 Brickel	Miami	FL	33135
4	Sam	Axe	222 Taimiami	Miami	FL	33109
5	Madeline	Weston	945 Sunset Lane	Miami	FL	33114
6	Jesse	Porter	9973 A1A	Miami	FL	33132
7	Barry	Burkowski	3838 Orange Grove St	Miami	FL	33314

Surrogate Key

- A Surrogate Key is a type of Primary Key which used a unique generated value.
- Should have no business value, and should never change.
- Typically a system generated self incrementing number. Can be a unique string (UUID).
- Considered a best practice in Relational Database Design.
- This will typically map to the @Id value of your entity

ID	FIRST_NAME	LAST_NAME	ADDRESS	CITY	STATE	ZIP_CODE
1234	Michael	Weston	123 Brickel	Miami	FL	33135
1235	Fiona	Glenanne	123 Brickel	Miami	FL	33135
1236	Sam	Axe	222 Taimiami	Miami	FL	33109
1237	Madeline	Weston	945 Sunset Lane	Miami	FL	33114
1238	Jesse	Porter	9973 A1A	Miami	FL	33132
1239	Barry	Burkowski	3838 Orange Grove St	Miami	FL	33314

Natural Key

- A Natural Key is a type of Primary Key which uses one or more data columns
- Is not a best practice because tied to business data, which can change
- Typically encountered with legacy databases
- Supported by JPA, will be covered later in the course
- Few reasons to use natural keys, one is a 'codes' table for configuration values

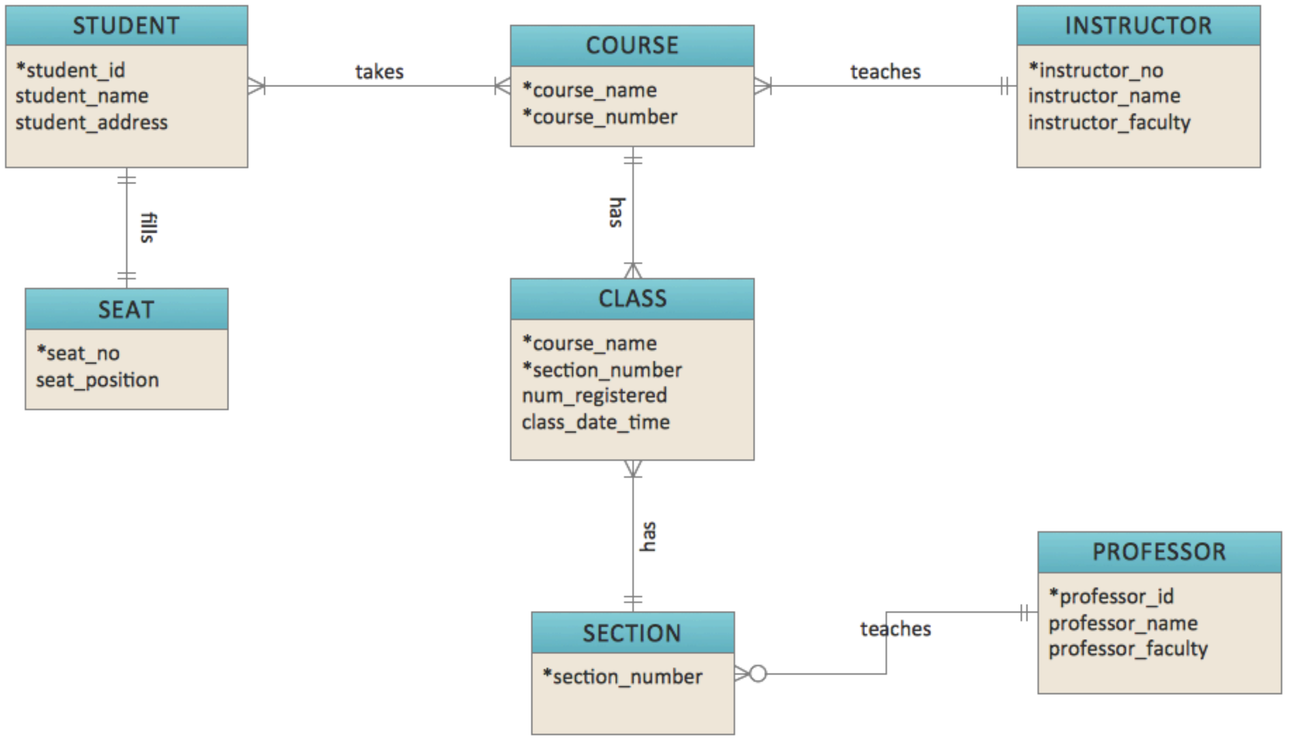
FIRST_NAME	LAST_NAME	ADDRESS	CITY	STATE	ZIP_CODE
Michael	Weston	123 Brickel	Miami	FL	33135
Fiona	Glenanne	123 Brickel	Miami	FL	33135
Sam	Axe	222 Taimiami	Miami	FL	33109
Madeline	Weston	945 Sunset Lane	Miami	FL	33114
Jesse	Porter	9973 A1A	Miami	FL	33132
Barry	Burkowski	3838 Orange Grove St	Miami	FL	33314

Database Table Relations

- Defined through Foreign Key Constraints in conjunction with Primary Keys.
- Types of Relations:
  - One to One - Record in Table A matches exactly one record in Table B
  - One to Many - Record in Table A matches many in Table B, but Table B matches only one record in Table A. (Think - An Order with multiple items)
  - Many to Many - Record in Table A matches many in Table B, and Table B matches many records in Table A.

Entity Relationship Diagram

- one to one:
- one to many:
- many to many:



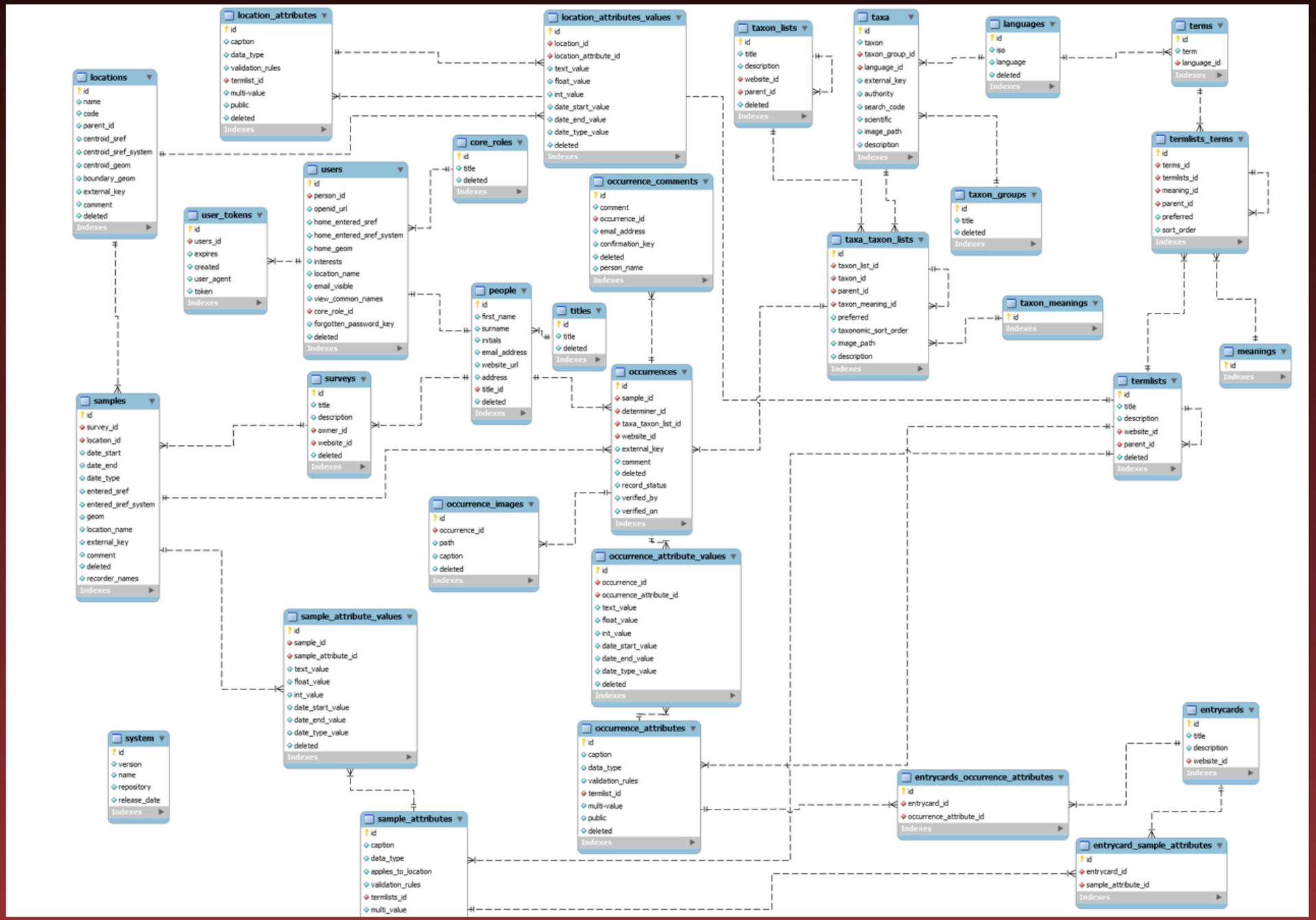
Example One to Many Relationship

ID	FIRST_NAME	LAST_NAME	ADDRESS	CITY	STATE	ZIP_CODE
1234	Michael	Weston	123 Brickel	Miami	FL	33135
1235	Fiona	Glenanne	123 Brickel	Miami	FL	33135
1236	Sam	Axe	222 Taimiami	Miami	FL	33109
1237	Madeline	Weston	945 Sunset Lane	Miami	FL	33114
1238	Jesse	Porter	9973 A1A	Miami	FL	33132
1239	Barry	Burkowski	3838 Orange Grove St	Miami	FL	33314

Table: CUSTOMER

ID	CUSTOMER_ID	DRINK_DESCRIPTION
122249	1234	Scotch
122250	1235	Pina Colada
122251	1236	Budwiser
122252	1237	White Wine
122253	1238	Stone IPA
122254	1239	Tequila Sunrise
122255	1234	Scotch
122256	1235	Pina Colada
122257	1236	Budwiser
122258	1237	Old Fashioned
122259	1238	Corona
122260	1239	Pina Colada

Table: DRINK_ORDER


location_attributes
id
caption
data_type
validation_rules
termlist_id
multi-value
public
deleted
Indexes
location_attributes_values
id
location_id
location_attribute_id
text_value
float_value
int_value
date_start_value
date_end_value
date_type_value
deleted
Indexes
taxon_lists
id
title
description
website_id
parent_id
deleted
Indexes
taxa
id
taxon
taxon_group_id
language_id
external_key
authority
search_code
scientific
image_path
description
Indexes
languages
id
iso
language
deleted
Indexes
terms
id
term
language_id
Indexes
locations
id
name
code
parent_id
centroid_sref
centroid_sref_system
centroid_geom
boundary_geom
external_key
comment
deleted
Indexes
core_roles
id
title
deleted
Indexes
termlists_terms
id
terms_id
termlists_id
meaning_id
parent_id
preferred
sort_order
Indexes
users
id
person_id
openid_url
home_entered_sref
home_entered_sref_system
home_geom
interests
location_name
email_visible
view_common_names
core_role_id
forgotten_password_key
deleted
Indexes
occurrence_comments
id
comment
occurrence_id
email_address
confirmation_key
deleted
person_name
Indexes
taxon_groups
id
title
deleted
Indexes
user_tokens
id
users_id
expires
created
user_agent
token
Indexes
taxa_taxon_lists
id
taxon_list_id
taxon_id
parent_id
taxon_meaning_id
preferred
taxonomic_sort_order
image_path
description
Indexes
taxon_meanings
id
Indexes
people
id
first_name
surname
initials
email_address
website_url
address
title_id
deleted
Indexes
titles
id
title
deleted
Indexes
meanings
id
Indexes
surveys
id
title
description
owner_id
website_id
deleted
Indexes
occurrences
id
sample_id
determiner_id
taxa_taxon_list_id
website_id
external_key
comment
deleted
record_status
verified_by
verified_on
Indexes
termlists
id
title
description
website_id
parent_id
deleted
Indexes
samples
id
survey_id
location_id
date_start
date_end
date_type
entered_sref
entered_sref_system
geom
location_name
external_key
comment
deleted
recorder_names
Indexes
occurrence_images
id
occurrence_id
path
caption
deleted
Indexes
occurrence_attribute_values
id
occurrence_id
occurrence_attribute_id
text_value
float_value
int_value
date_start_value
date_end_value
date_type_value
deleted
Indexes
sample_attribute_values
id
sample_id
sample_attribute_id
text_value
float_value
int_value
date_start_value
date_end_value
date_type_value
deleted
Indexes
occurrence_attributes
id
caption
data_type
validation_rules
termlist_id
multi-value
public
deleted
Indexes
entrycards
id
title
description
website_id
Indexes
system
id
version
name
repository
release_date
Indexes
entrycards_occurrence_attributes
id
entrycard_id
occurrence_attribute_id
Indexes
sample_attributes
id
caption
data_type
applies_to_location
validation_rules
termlists_id
multi_value
entrycard_sample_attributes
id
entrycard_id
sample_attribute_id
Indexes

Database Constraints

- A database constraint is a rule applied to the data stored in the database
- Not Null - Column is not allowed to have null values
- Unique - Column value must be unique
- Primary Key - Identifier of row, combines Not Null and Unique
- Foreign Key - Value must exist in referenced (foreign) table aka, referential integrity
- Check Constraint - Sets condition on data - like min max, or list of values (ENUM)

Database Transactions

- A database transaction is a series of one or more DML statements
- A transaction has a begin and an end
- Committing will make the data changes permanent
- A rollback will revert the changed data to the original state

ACID

- What is ACID?
  - Atomicity - all or nothing
  - Consistency - transactions are valid to rules of the DB
  - Isolation - Results of transactions are as if they are done end to end
  - Durability - Once a transaction is committed, it remains so

Transactions & ACID - What does it mean?

- Atomicity - All statements must be able to complete
- Consistency - Changes do not violate constraints
- Isolation - Reads inside the transaction 'see' changed data. Reads outside transaction 'see' original data (until commit)
- Durability - Once committed, changes are permanent
- Easy with one user, becomes very complex with many transacting users!

Lost Updates

- ACID can lead to lost updates
- Michael reads balance is 10 and decides to add 5. Thus: 10 + 5 = 15
- Before Michael commits, Sam reads balance is 10 and decides to add 10. Thus 10 + 10 = 20
- Actual balances should be 10 + 5 + 10 = 25
- But balance is updated to 20, because Michael's update is 'lost' since Sam's transaction never 'see's' the updated value

Preventing Lost Updates

- Locking is one technique which can be used to prevent lost updates
- Pessimistic Locking uses a database lock to prevent inflight transactions and will allow transactions to complete sequentially. ie - Select for update, will wait for an exclusive lock
- Optimistic Locking - Uses a version property which is checked in the update
- Examples of both will be explored later in the course!

SPRING FRAMEWORK

GURU