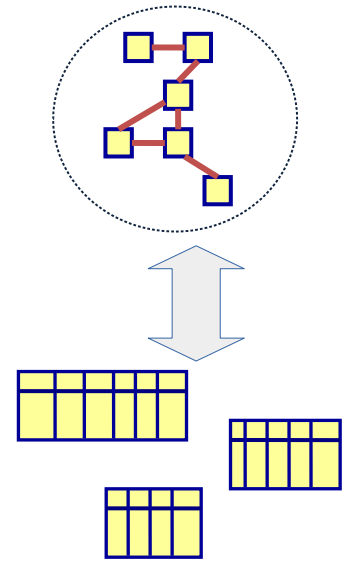CS544 EA
Hibernate

Entity Mapping

Entity

- An entity is a Domain Class
  - The most basic part of the domain
- **Classes map to Tables** in the DB
  - There is almost no mis-match here
  - They just need to **specify an ID** field

Entity Class

- JPA Requires that entity classes have:
  - A field to use as **ID**
  - A default **constructor**
  - Getters and setters for all properties

Java Bean

```java
package cs544.hibernate01.basic;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;

@Entity
public class Customer {
    @Id
    @GeneratedValue
    private Long id;
    private String firstName;
    private String lastName;

    public Customer() { }

    public Customer(String firstName, String lastName) {
        this.firstName = firstName;
        this.lastName = lastName;
    }

    public int getId() { return id; }
    public void setId(int id) { this.id = id; }
    public String getFirstName() { return firstName; }
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }
    public String getLastName() { return lastName; }
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
}
```

4

@Entity

- **@Entity** specifies that a class is an entity.
- By default the class name = entity name
  - This entity name is used in Queries   *"from Customer"*

- @Entity(name="OtherName")
  - Gives the entity a different name
  - Also causes the table name to change
    - When generating tables from annotations

Optional @Table

- **@Table**(name=”OtherName”)
  - Changes table name that entity is mapped to
  - Without changing the name of the Entity

- @Table also has options for:
  - Mapping to a different schema (db)
  - Specifying unique constraints (if generating schema)
  - Specifying indexes (if generating schema)