

Week	Session	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
WEEK 1		Theme I: Spring – The Nature of life is to grow					
	a.m.	Overview	DI	AOP	Hibernate	Persistence API	Exam
	p.m.	Spring Context	Spring Startup	Spring Apps	Entities	Review	Homework
	Eve	Homework	Homework	Homework	Homework		
WEEK 2		Theme II: Hibernate – Rest and Activity are the Steps of Progress					
	a.m.	Associations	Complex	Queries	Web Apps	Transactions	Exam
	p.m.	Collections	Queries	Optimization	Concurrency	Review	Homework
	Eve	Homework	Homework	Homework	Homework		
WEEK 3		Theme III: Applications – Life is Found in Layers					
	a.m.	Spring MVC	Security	Spring Boot	Messaging	Project Start	Exam
	p.m.	Spring Data	Validation	REST	Integration	Review	Integration Project
	Eve	Homework	Homework	Homework	Homework		
WEEK 4		Theme IV: Integration Project – The Whole is Greater than the Sum of the Parts					
	a.m.	Integration Project	Integration Project	Integration Project	Presentations		
	p.m.						
	Eve						



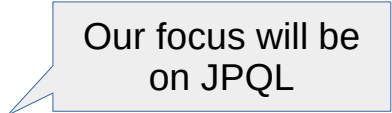
CS544 EA

Hibernate

Queries

Queries

- So far we've used `.find()` or `.getReference()`
 - And then follow references to related objects
 - But what if you don't know an entity's ID?
- JPA offers **several ways** to query the DB
 - JPQL: Java Persistence Query Language (SQL like)
 - Criteria API: Create queries with Java objects
 - Stored Procedure Queries: executed stored procedures
 - Native Queries: Execute SQL and get objects



Our focus will be
on JPQL

JPQL

- The Java Persistence Query Language (JPQL) is a standardization of the Hibernate Query language (HQL).
 - JPQL is a subset of HQL (HQL has a few extensions)
- JPQL syntax is **similar to SQL**, but OO:
 - Understands objects and attributes
 - Understands associations between objects
 - Understands inheritance and polymorphism

```
List<Account> accounts = em.createQuery("from Account a "  
    + "where a.class <> CheckingAccount "  
    + "and a.owner.firstName = 'Frank'", Account.class)  
    .getResultList();
```