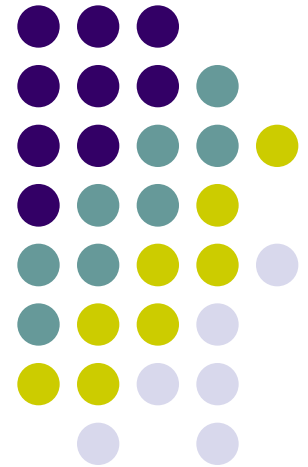

Consultas con Hibernate





HQL - Introducción

- Lenguaje de consulta OO
- Independiente de la base de datos
- Soporte de consultas polimórficas

```
String hql = "from Persona";  
Query query = session.createQuery(hql);  
List resultado = query.list();
```



HQL - Parameter

```
String hql = "from Persona as p where p.estado =  
:estado";
```

```
Query query = session.createQuery(hql);  
query.setParameter("estado", "nuevo");
```

```
List resultado = query.list();
```



HQL - UniqueResult

```
long id = 123456;  
String hql = "from Persona as p where p.id = :id";  
Query query = session.createQuery(hql);  
query.setParameter("id", id);  
Persona persona = (Persona) query.uniqueResult();
```



HQL - Paging

```
String hql = "from Persona";  
Query query = session.createQuery(hql);  
query.setFirstResult(1);  
query.setMaxResults(25);  
List resultado = query.list();
```



HQL - SQL

```
Query query = session.createQuery("select *  
from Persona");  
query.addEntity(Persona.class);  
List result = query.list();
```

HQL - Ejemplos



```
select p.direccion from Personas as p where p.id = :id
```

```
from Persona as p where p.direccion.id = :id
```

```
from Persona as p where p.direccion.calle = :direccion
```



HQL - Ejemplos

```
select p.telefonos from Persona as p where p.id = :id
```

```
from Persona as p where p.direccion.nombre like '%s%'
```

```
from Persona as p where p.nombre is not null
```




HQL - Ejemplos

```
from Persona as p where p.estado in ('Normal',  
'Borrado', 'Pendiente')
```

```
from Persona as p where p.direccion.nombre like '%s%'  
order by p.nombre
```

```
select a.patente FROM Persona as p JOIN p.autos a  
WHERE p.id = :id
```

HQL - Funciones de agregación



select **sum(e.salario)** from Empleado as e where.....

select **avg(e.salario)** from Empleado as e where.....

select **count(*)** from Empleado as e where.....

select **min(e.salario)** from Empleado as e where.....

select **max(e.salario)** from Empleado as e where.....



Hibernate Criteria

Es una API que nos facilita las consultas a la base de datos. La ventaja fundamental de esta API es que nos permite tratar la composición de la consulta de una forma totalmente orientada a objetos.



Hibernate Criteria

Es una API que nos facilita las consultas a la base de datos. La ventaja fundamental de esta API es que nos permite tratar la composición de la consulta de una forma totalmente orientada a objetos.

```
Criteria crit = sess.createCriteria(Empleado.class);  
List personas = crit.list();
```

Criteria - Ejemplos



```
List personas= session.createCriteria(Persona.class)
    .add(Restrictions.like("nombre", "perez%"))
    .add(Restrictions.gt( "peso", new Float(50)))
    .addOrder(Order.asc("edad"))
    .list();
```