

# Exemplos de Query

```
List cats = session.createQuery(
    "from Cat as cat where cat.birthdate < ?")
    .setDate(0, date)
    .list();
```

```
List mothers = session.createQuery(
    "select mother from Cat as cat join cat.mother as mother where cat.name = ?")
    .setString(0, name)
    .list();
```

```
List kittens = session.createQuery(
    "from Cat as cat where cat.mother = ?")
    .setEntity(0, pk)
    .list();
```

```
Cat mother = (Cat) session.createQuery(
    "select cat.mother from Cat as cat where cat = ?")
    .setEntity(0, izi)
    .uniqueResult();]]
```

```
Query mothersWithKittens = (Cat) session.createQuery(
    "select mother from Cat as mother left join fetch mother.kittens");
Set uniqueMothers = new HashSet(mothersWithKittens.list());
```

# Iteração de resultados

- `list()`
  - Aloca todos os resultados da query em memória
- `iterate()`
  - Acessa a base a cada chamada de `next()`
  - Útil quando espera-se resultados existentes em cache

```
// fetch ids
Iterator iter = sess.createQuery("from eg.Qux q order by q.likeliness").iterate();
while ( iter.hasNext() ) {
    Qux qux = (Qux) iter.next(); // fetch the object
    // something we couldnt express in the query
    if ( qux.calculateComplicatedAlgorithm() ) {
        // delete the current instance
        iter.remove();
        // dont need to process the rest
        break;
    }
}
```

# Queries que retornam tuplas

- Cada tupla é interpretada como um array
  - Não precisa de Cast?!?!?

```
Iterator kittensAndMothers = sess.createQuery(
    "select kitten, mother from Cat kitten join kitten.mother mother")
    .list()
    .iterator();

while ( kittensAndMothers.hasNext() ) {
    Object[] tuple = (Object[]) kittensAndMothers.next();
    Cat kitten  = tuple[0];
    Cat mother  = tuple[1];
    ....
}
```

# Queries com parâmetros

- “Binding Values”
  - ?
  - :param
- :param
  - Não são sensíveis a ordem de ocorrência
  - Podem ocorrer diversas vezes em uma única query
  - São auto-documentáveis

```
//named parameter (preferred)
Query q = sess.createQuery("from DomesticCat cat where cat.name = :name");
q.setString("name", "Fritz");
Iterator cats = q.iterate();
```

```
//positional parameter
Query q = sess.createQuery("from DomesticCat cat where cat.name = ?");
q.setString(0, "Izi");
Iterator cats = q.iterate();
```

- Especificar o número máximo de ocorrências
- Primeira linha a ser retornada
  - ???

```
Query q = sess.createQuery("from DomesticCat cat");  
q.setFirstResult(20);  
q.setMaxResults(10);  
List cats = q.list();
```