QBC查询就是query by criteria使用criteria对象进行查询。

（1）全表查询：

Criteria ce = session.createCriteria(person.class);其中person表示需要查询的实体类的类名

```
List<person> list = ce.list();
for(person person : list){
        System.out.println(person.getName());
}
```

（2）条件查询（完全面向对象的查询）

单条件查询：

```
Criteria ce = session.createCriteria(person.class);
                //添加条件:name="值"
                ce.add(Restrictions.eq("name", "21a12nqili"));
```

其中那name表示需要查询的字段名，21a12nqili表示查询name的字段值

```
List<person> list = ce.list();
for(person person : list){
        System.out.println(person.getName());
}
```

多条件查询：

```
//添加条件:id="值" and name="值"
                ce.add(Restrictions.and(Restrictions.eq("id", 6),Restrictions.eq("name", "555")));
```

其中and可以换成or或者not，其中not时只有一个条件，and和or可以有多个条件

```
List<person> list = ce.list();
for(person person : list){
        System.out.println(person.getName());
}
```

（3）分页查询

```
Criteria ce = session.createCriteria(person.class);
                //分页查询
                ce.setFirstResult(0);//起始行
                ce.setMaxResults(3);//查询的行数
                List<person> list = ce.list();
                for(person person : list){
```

```
                                System.out.println(person.getName());
                        }
```

（4）排序查询

```
Criteria ce = session.createCriteria(person.class);
                // 排序查询:order by id desc
                ce.addOrder(org.hibernate.criterion.Order.desc(“id”));
```

其中id表示需要排序的字段名，desc表示降序排列，asc表示是升序排序

```
                List<person> list = ce.list();
                for (person person : list) {
                        System.out.println(person.getName());
                }
```

（5）聚合查询

```
Criteria ce = session.createCriteria(person.class);
                // 聚合查询：查询总记录数
                CountProjection projection = Projections.count(“id”);
                ce.setProjection(projection);
                Object countLong = ce.uniqueResult();
                System.out.println(countLong);
                // 聚合查询：查询id的最大值
                ce.setProjection(Projections.max(“name”));
                String max = (String)ce.uniqueResult();
                System.out.println(max);
```

同理可以查询最大值，最小值，平均值，总和等。但是要注意数据的类型，将查询到的数据进行转换。

（6）投影查询

```
Criteria ce = session.createCriteria(person.class);
                // 投影查询
                ProjectionList pList = Projections.projectionList();
                pList.add(Property.forName(“id”));
                pList.add(Property.forName(“name”));
                ce.setProjection(pList);
                List<Object[]> list = ce.list();
                for (Object[] objects : list) {
                        for(Object object:objects){
                                System.out.print(object+“,”);
```

```
        }
        System.out.println();
    }
```