1，ArrayHandler

```
@Test
    public void test1()throws SQLException {
        //ArrayHandler适合取一条记录，把该记录的每一列封装到一个object类
型的数组中
        QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
        Object object[] = qRunner.query("select *from linm", new
ArrayHandler());
        for(Object object2:object)
        {
            System.out.println(object2);
        }
    }
```

2，ArrayListHandler

```
    @Test
    public void test2()throws SQLException{
        //ArrayListHandler适合取多条记录，把每一条记录的每列值封装到一个
数组中object[]，把数组封装到一个list中
        QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
        List<Object[]>obj = qRunner.query("select *from linm", new
ArrayListHandler());
        for(Object[] object2:obj)
        {
            for(Object os :object2){
                System.out.print(os+"   ");
            }
            System.out.println();
        }
    }
```

3，   ColumnListHandler

```
    @Test
    public void test3()throws SQLException{
        //ColumnListHandler:取出某一列的数据，封装到list中
        QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
```

```java
            List<Object> list = qRunner.query("select *from linm", new
ColumnListHandler(2));
            for(Object o:list)
            {
                    System.out.println(o);
            }
    }
```

4，KeyedHandler

```java
    @Test
    public void test4()throws SQLException{
            //map中的key是不可以重复的
            //KeyedHandler:取出多条记录，每一条记录封装在一个map集合中，在把
这个map封装在另外一个map中，key为指定的字段值
            QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
            //大map的key是是表中的某列数据，小map中的key是列名。所以大map是
object类型，小map是string
            Map<Object,Map<String,Object >> list = qRunner.query("select
*from linm", new KeyedHandler(1));
            for(Map.Entry<Object, Map<String, Object>> m:list.entrySet())
            {
                    System.out.println(m.getKey());
                    for(Map.Entry<String, Object>mm:m.getValue().entrySet())
                    {
                            System.out.println(mm.getKey()+"
"+mm.getValue());
                    }
                    System.out.println("--------------------");
            }
    }
```

5，MapHandler

```java
    @Test
    public void test5()throws SQLException{
            //map中的key是不可以重复的
            //MapHandler:适合取一条记录，把当前记录的列名和列值放到一个map集
合中
```

```java
		QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
		Map<String, Object> list = qRunner.query("select *from linm",
new MapHandler());
	for(Map.Entry<String, Object>li:list.entrySet())
			{
		System.out.println(li.getKey()+"  "+li.getValue());
			}
	}
```

6，MapListHandler

```java
	@Test
	public void test6()throws SQLException{
		//map中的key是不可以重复的
		//MapListHandler:适合取多条记录，把每一条记录封装到一个map中，在
把map封装在一个list集合中
		QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
		List<Map<String, Object> >list = qRunner.query("select *from
linm", new MapListHandler());
	for(Map<String, Object> ob:list)
			{
		for(Map.Entry<String, Object>li:ob.entrySet())
			{
	System.out.println(li.getKey()+"  "+li.getValue());
			}
			}
	}
```

7，ScalarHandler

```java
	@Test
	public void test7()throws SQLException{
		//map中的key是不可以重复的
		//ScalarHandler:适合取单行单列的数据
		QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
		Object object = qRunner.query("select count(*)from linm", new
ScalarHandler(1));
	System.out.println(object);
	System.out.println(object.getClass().getName());//得到object的数据类型
```

}

8，BeanHandler

```java
@Test
public void test8()throws SQLException{
        //map中的key是不可以重复的
        //BeanHandler:适合取单行的数据,没有就返回null值
        QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
        user user = qRunner.query("select *from linm", new
BeanHandler<user>(user.class));
        System.out.println(user.getId()+"   "+user.getNamess());

}
```

9，BeanListHandler

```java
@Test
public void test9()throws SQLException{
        //map中的key是不可以重复的
        //BeanListHandler:适合取多行数据,封装在list集合中，没有值的话
list数组的长度为
        QueryRunner qRunner = new QueryRunner(C3P0Util.getDataSource());
        List<user>user = qRunner.query("select *from linm", new
BeanListHandler<user>(user.class));
        for(user user2:user){
                System.out.println(user2.getId()+"   "+user2.getNamess());
        }

}
```