**实验一：分析SQL执行计划，执行SQL语句的优化指导**

**实验内容：**

* 对Oracle12c中的HR人力资源管理系统中的表进行查询与分析。
* 首先运行和分析教材中的样例：本训练任务目的是查询两个部门('IT'和'Sales')的部门总人数和平均工资，以下两个查询的结果是一样的。但效率不相同。
* 设计自己的查询语句，并作相应的分析，查询语句不能太简单。

**实验过程**

**1.运行并分析教材样例**

* **1.1查询一**

SELECT d.department\_name，count(e.job\_id)as "部门总人数",

avg(e.salary)as "平均工资"

from hr.departments d,hr.employees e

where d.department\_id = e.department\_id

and d.department\_name in ('IT','Sales')

GROUP BY department\_name;

* **1.2查询二**

SELECT d.department\_name，count(e.job\_id)as "部门总人数",

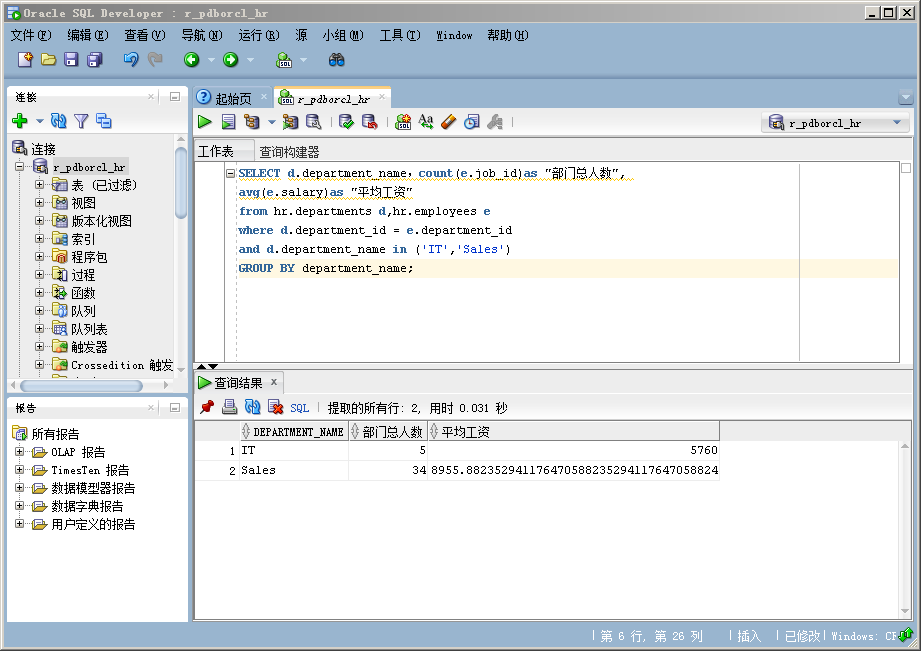
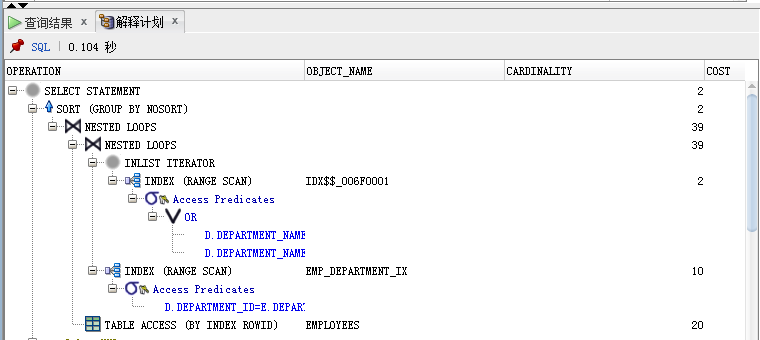
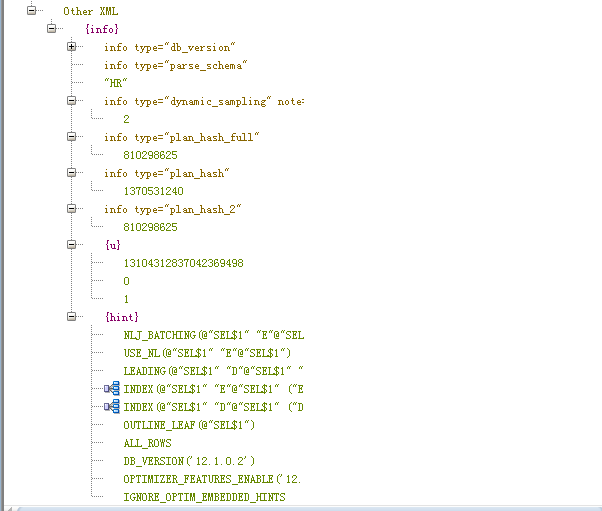
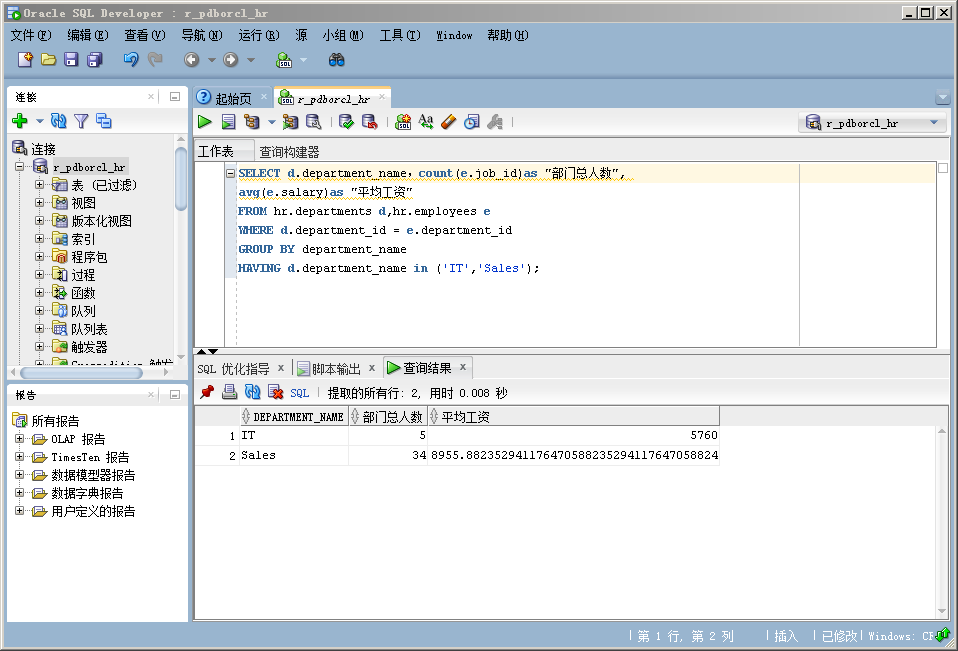
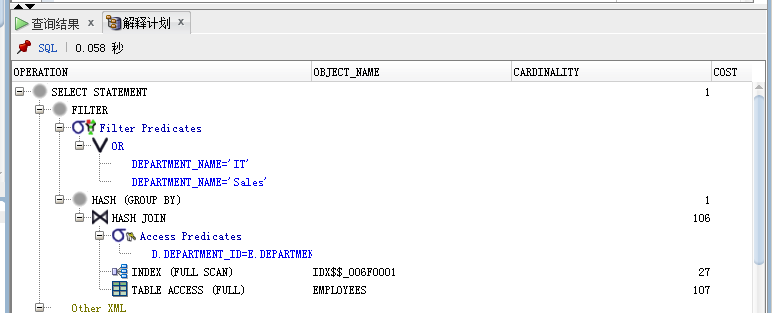
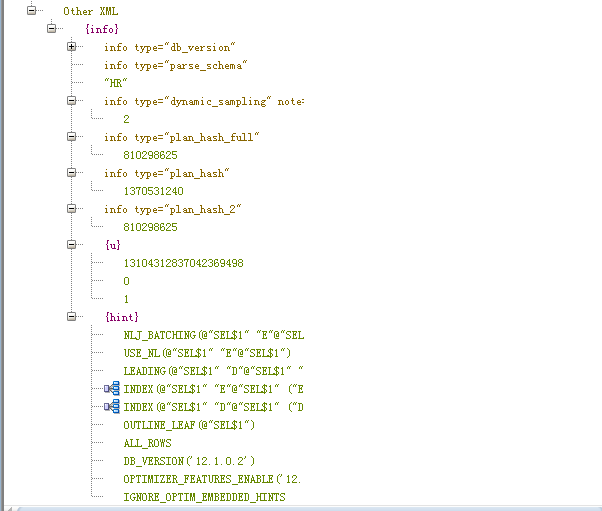
avg(e.salary)as "平均工资"

FROM hr.departments d,hr.employees e

WHERE d.department\_id = e.department\_id

GROUP BY department\_name

HAVING d.department\_name in ('IT','Sales');

* **1.3查询截图：**
* 1.3.1查询一截图 [](https://raw.githubusercontent.com/tsxbox/Oracle/master/a.png)[](https://raw.githubusercontent.com/tsxbox/Oracle/master/b.png) [](https://raw.githubusercontent.com/tsxbox/Oracle/master/c.png)
* 1.3.2查询二截图 [](https://raw.githubusercontent.com/tsxbox/Oracle/master/g.png)[](https://raw.githubusercontent.com/tsxbox/Oracle/master/e.png) [](https://raw.githubusercontent.com/tsxbox/Oracle/master/f.png)
* **1.4 查询分析：**  
  通过截图可以发现，查询语句一用时0.031s,查询语句2用时0.008s。查询语句二用时更少，更加优化。  
  第一个查询语句group by分组 计算每一个部门，把所有的组的数据都查询过，只是没有显示。where条件限定了，多计算了除开IT，Sales的分组，所以用时更多。  
  第二个查询只计算IT和Sales组的数据，所以时间更少，更优化。having和where类似，唯一的差别是where过滤元组，having过滤分组。
* **1.5sqldevolper优化指导**  
  sqldevloper对两个查询均未给出优化指导

**2.自我设计**

* **2.1自己的查询语句**

SELECT D.DEPARTMENT\_NAME,L.CITY,COUNT(\*)

FROM EMPLOYEES E,DEPARTMENTS D,LOCATIONS L

WHERE E.DEPARTMENT\_ID = D.DEPARTMENT\_ID AND D.LOCATION\_ID = L.LOCATION\_ID AND E.SALARY > 1000

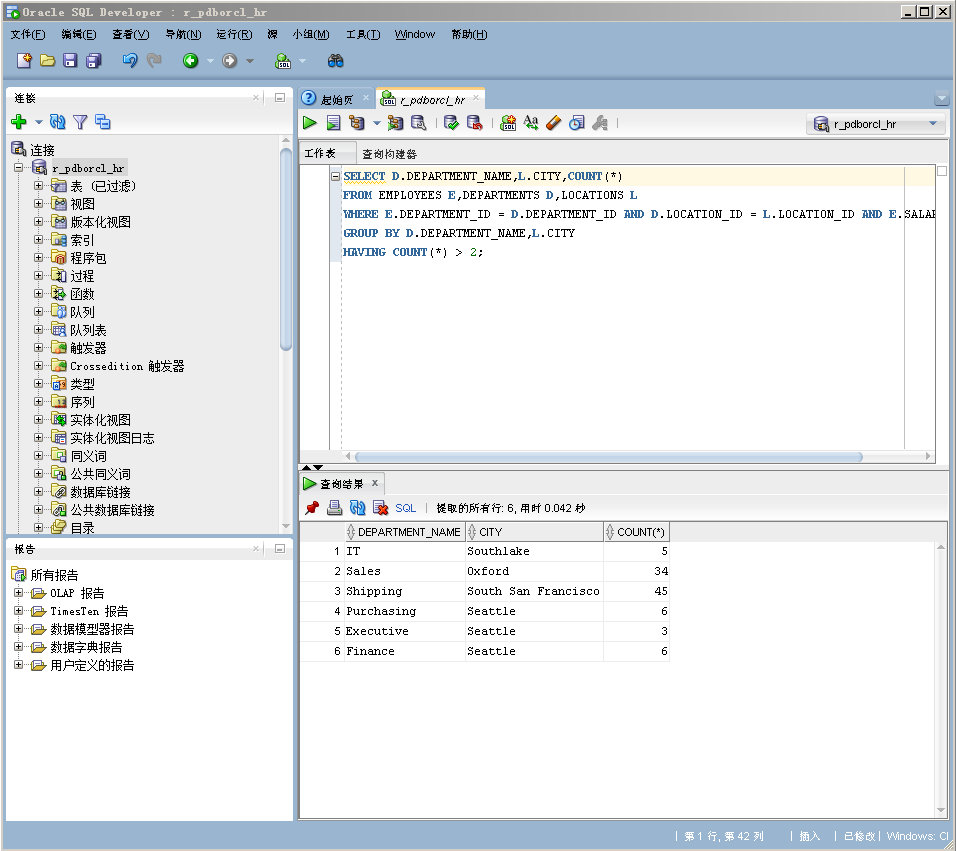
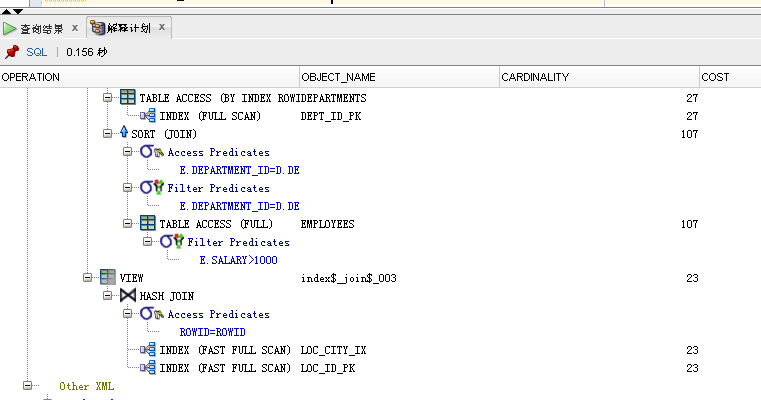
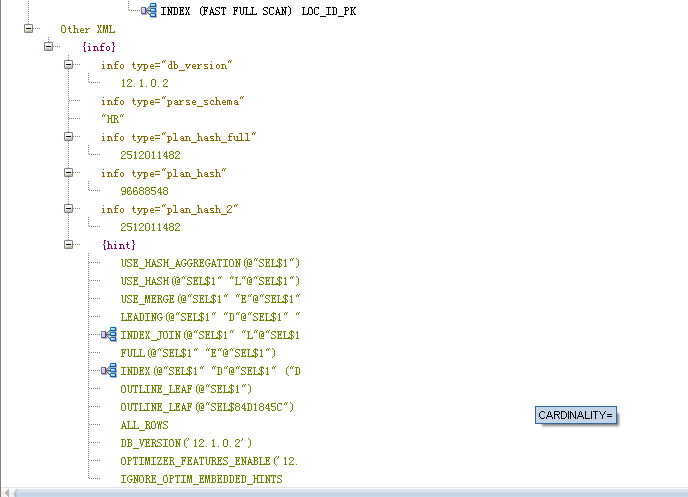
GROUP BY D.DEPARTMENT\_NAME,L.CITY

HAVING COUNT(\*) > 2;

* **2.2自己的查询语句分析**

查询需求：列出每个部门中工资高于一千的员工数量超过二人的部门，显示部门名字，地区名称，以及总人数。  
详细分析：该查询由基本的SELECT、FROM和WHERE三个子句构成基本查询结构，再使用GROUP BY对查询结果按列进行分组，使相等的分为一组。HAVING子句对分组结果进行选择，仅输出满足条件的组，并和GROUP BY结合使用。该查询运用了等值连接E.DEPARTMENT\_ID = D.DEPARTMENT\_ID AND D.LOCATION\_ID = L.LOCATION\_ID AND E.SALARY > 1000 实现了多个关系的查询，并表示出了工资大于一千，以及表之间对应应该满足的条件，最后由HAVING子句实现人数超过2人的条件完成查询。

* **2.3sqldevoper执行结果**

[](https://raw.githubusercontent.com/tsxbox/Oracle/master/x.png)[](https://raw.githubusercontent.com/tsxbox/Oracle/master/y.png) [](https://raw.githubusercontent.com/tsxbox/Oracle/master/z.png)