项目九　综合案例：电商销售大数据分析



《Hive数据仓库应用教程》

任务 9-1 电商销售大数据分析

9.1.1 任务描述

全世界有数十亿人在使用电商平台进行购物，这些平台在较短的时间内产生了大量的数据。本项目将学会如何使用Hive数据仓库来载入电商销售的历史数据，并使用HiveQL对各项指标进行统计分析，进一步熟悉Hive数据仓库的实际应用。







9.1.2 任务准备知识

1.电商销售大数据

（1）电子商务通常是指在全球各地广泛的商业贸易活动中，在因特网开放的网络环境下，基于客户端/服务端应用方式，买卖双方不谋面地进行各种商贸活动，实现消费者的网上购物、商户之间的网上交易和在线电子支付以及各种商务活动。

（2）电商数据包括所有注册用户的浏览信息、购买消费记录、用户对商品的评价、在其平台上卖家的买卖记录、产品交易量、库存量，以及商家的信用信息等等。

（3）电商经历了基于用户数量的时代，基于销量的时代，目前已处于基于大数据的时代，电子商务的竞争在很大程度上就是大数据的竞争。

2．数据概况

本案例数据来自奢侈品电商后台的部分历史销售数据，平台为了创造更多的销售利润，现需要对已有的销售大数据进用户销售数据的分析，通过对各项指标的统计分析，进行用户画像，提出合理的促销方案，从而进行精准营销。

用户的历史销售数据以txt的文本格式存储，每条数据是订单记录，销售订单的格式描述如下：

序号	字段	描述	序号	字段	描述
1	uid	用户标识	7	livetime	居住时间
2	pid	商品标识	8	marriage	婚姻状况
3	sex	性别	9	level1	商品一级分类
4	age	年龄	10	level2	商品二级分类
5	vocation	职业	11	level3	商品三级分类
6	city	居住城市	12	sale	销售额

9.1.2 任务准备知识

2．数据准备

我们将历史数据上传到node2虚拟机的data目录里，然后加载到Hive数据库中，操作的步骤如下：

步骤1：将本机的历史销售数据e-data.txt上传到node2的data目录下，如图9-1所示：

步骤2：创建Hive数据库eshop，如图9-2所示：

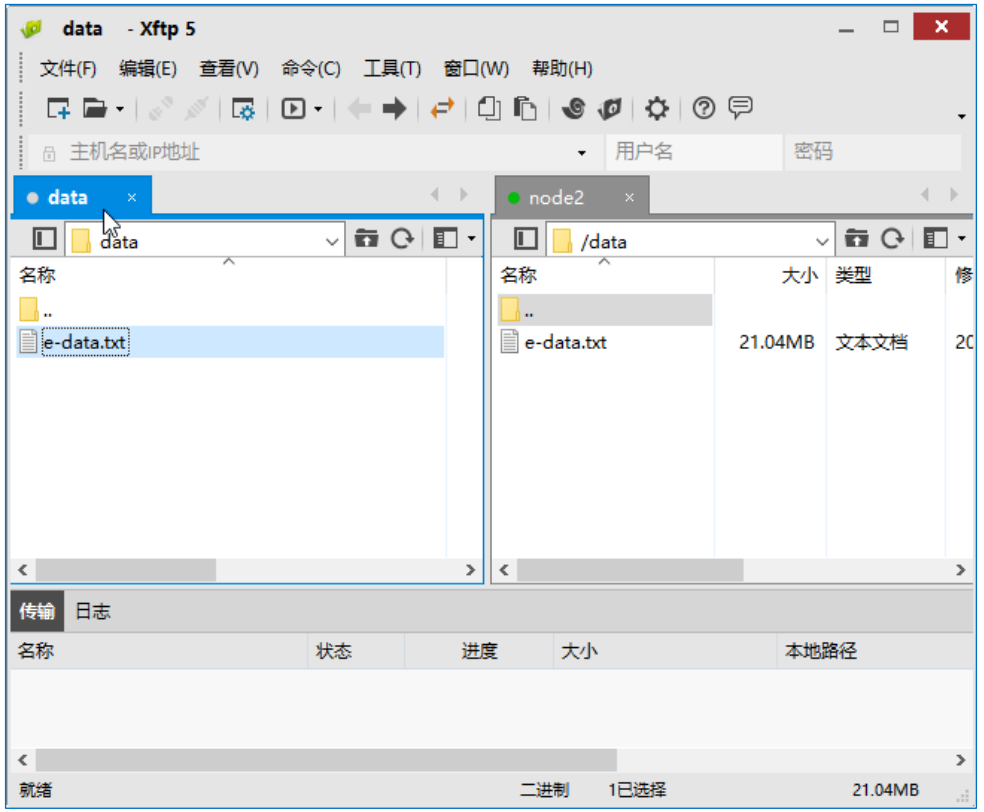


图9-1

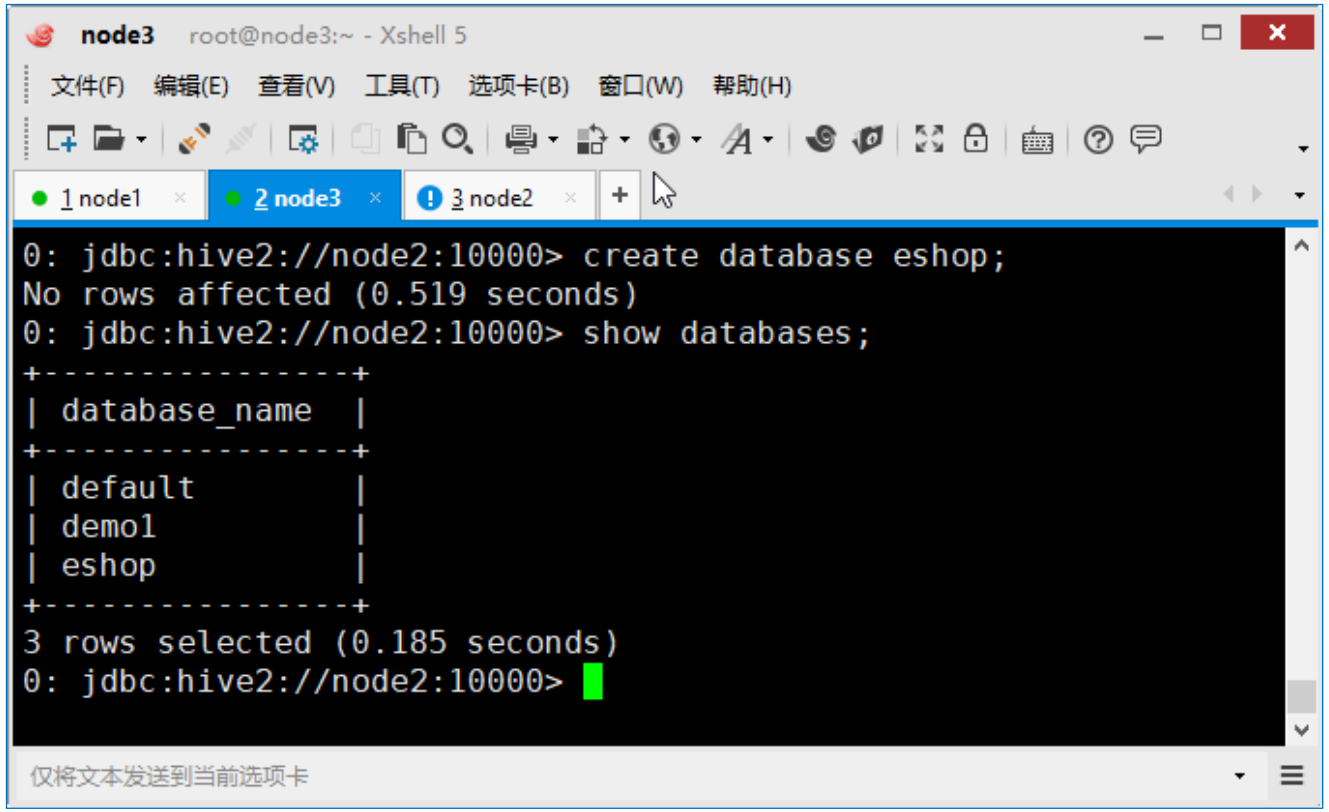


图9-2

9.1.2 任务准备知识

2．数据准备

步骤3：创建Hive销售订单表sales，如图9-3所示，建表成功如图9-4所示

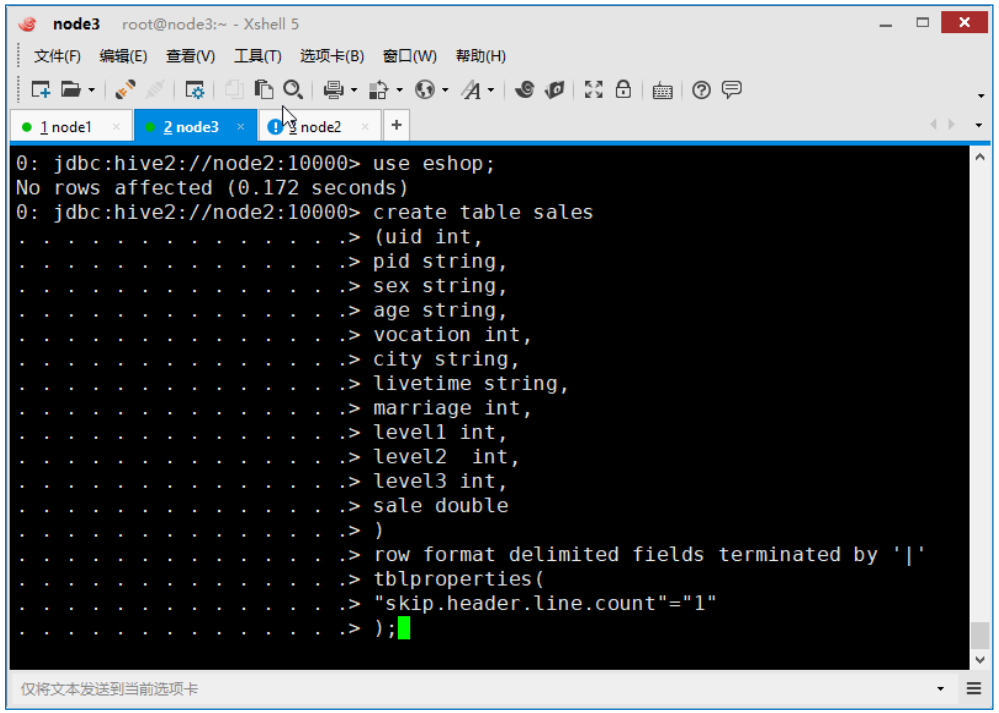


图9-3

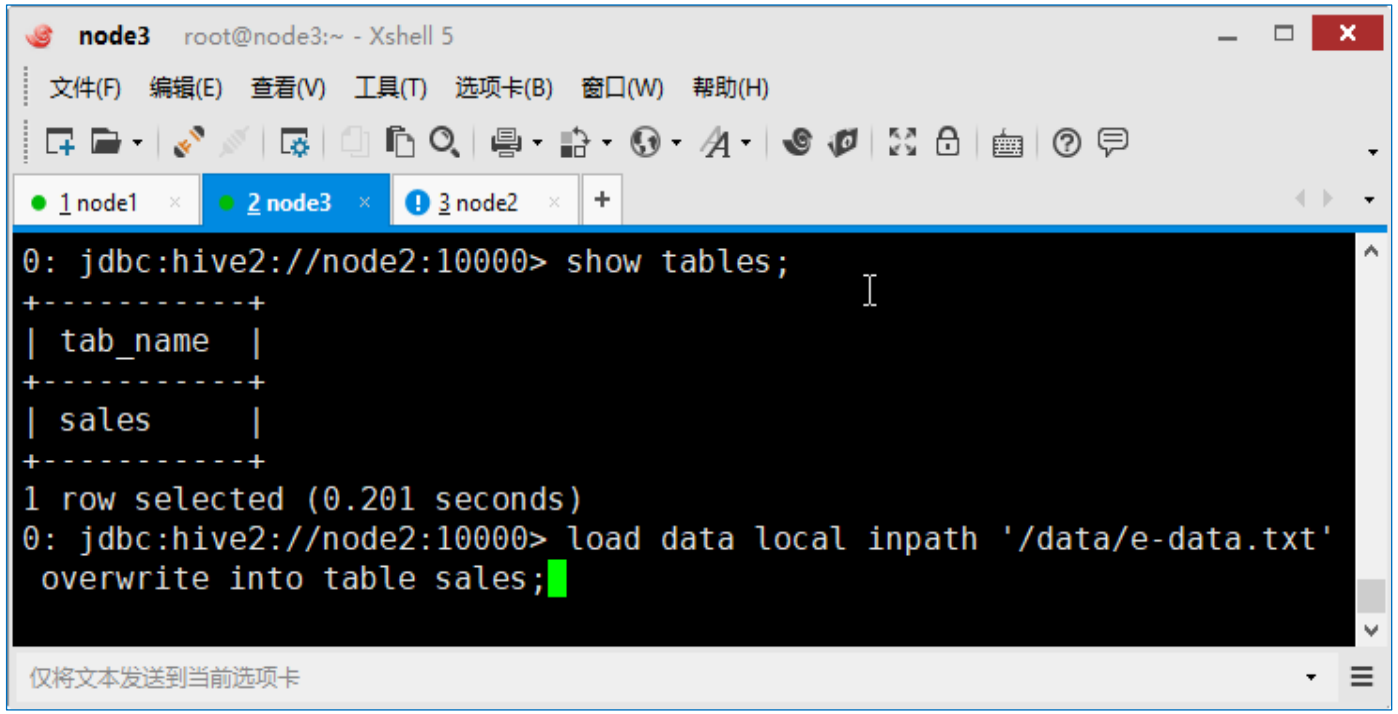


图9-4

9.1.2 任务准备知识

2．数据准备

步骤4：将历史数据装载到订单表sales里，HiveQL脚本设计如下：

```
load data local inpath '/data/e-data.txt' overwrite into table sales;
```

查询一下数据，如果看到如下图9-5所示表示历史数据加载成功。

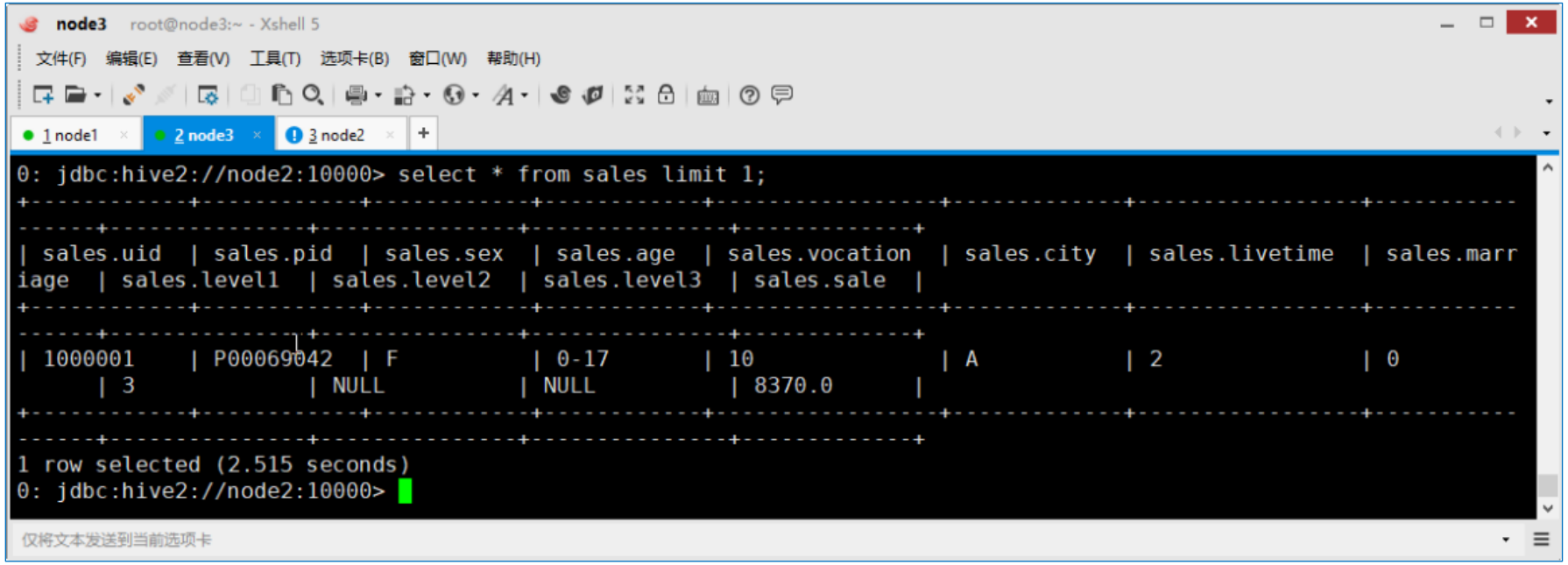


图9-5

9.1.3 任务实施

1．统计订单总数

统计订单历史数据中订单总数，我们使用count函数来完成这个统计需求，count(1)可以对所有数据行进行统计，也就是统计出了所有的订单数，HiveQL脚本设计如下：运行结果如图9-6所示。

```
select count(1) as items from sales;
```

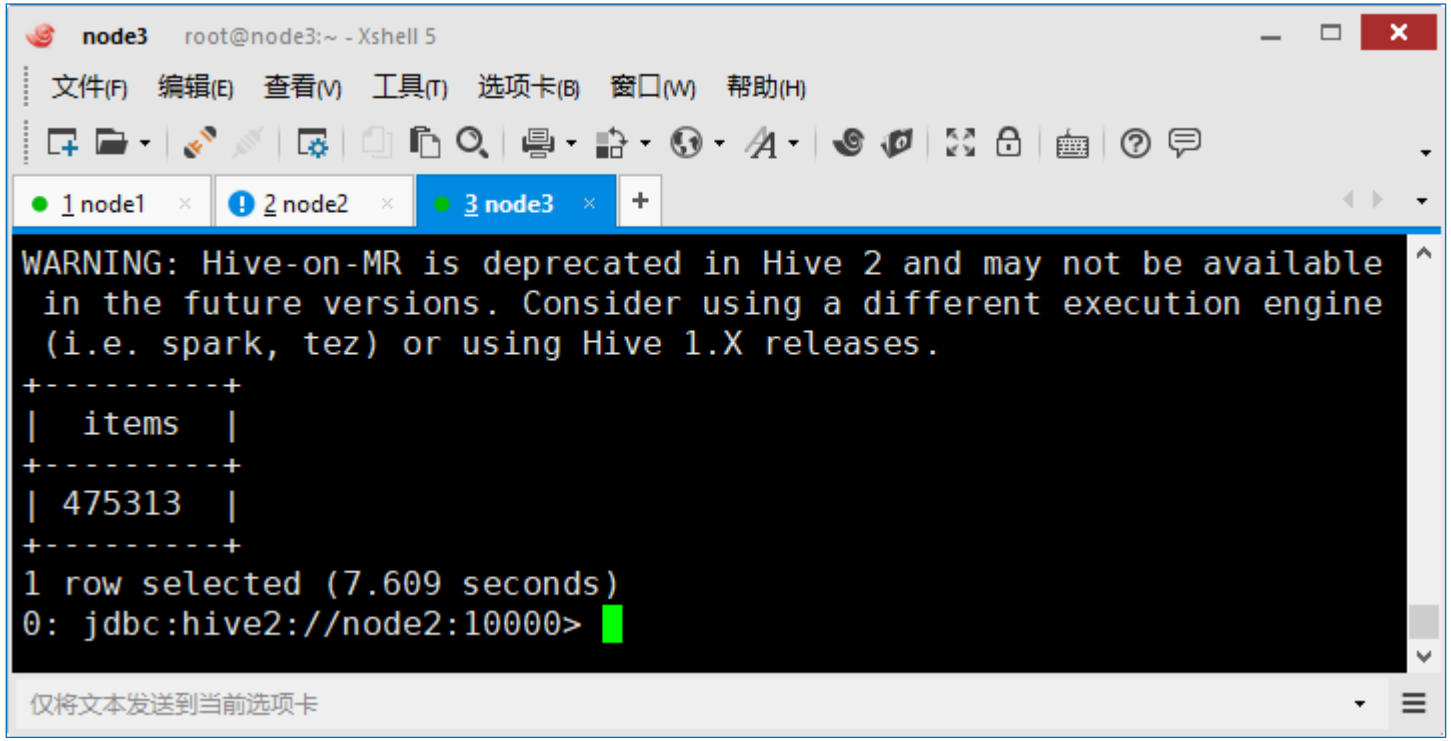


图9-6

2．统计总销售额

HiveQL脚本设计如下：

```
select
sum(sale)  sales,
sum(sale)/count(distinct uid) avg_pay,
sum(sale)/count(sale) avg_price
from sales;
```

运行结果如图9-7

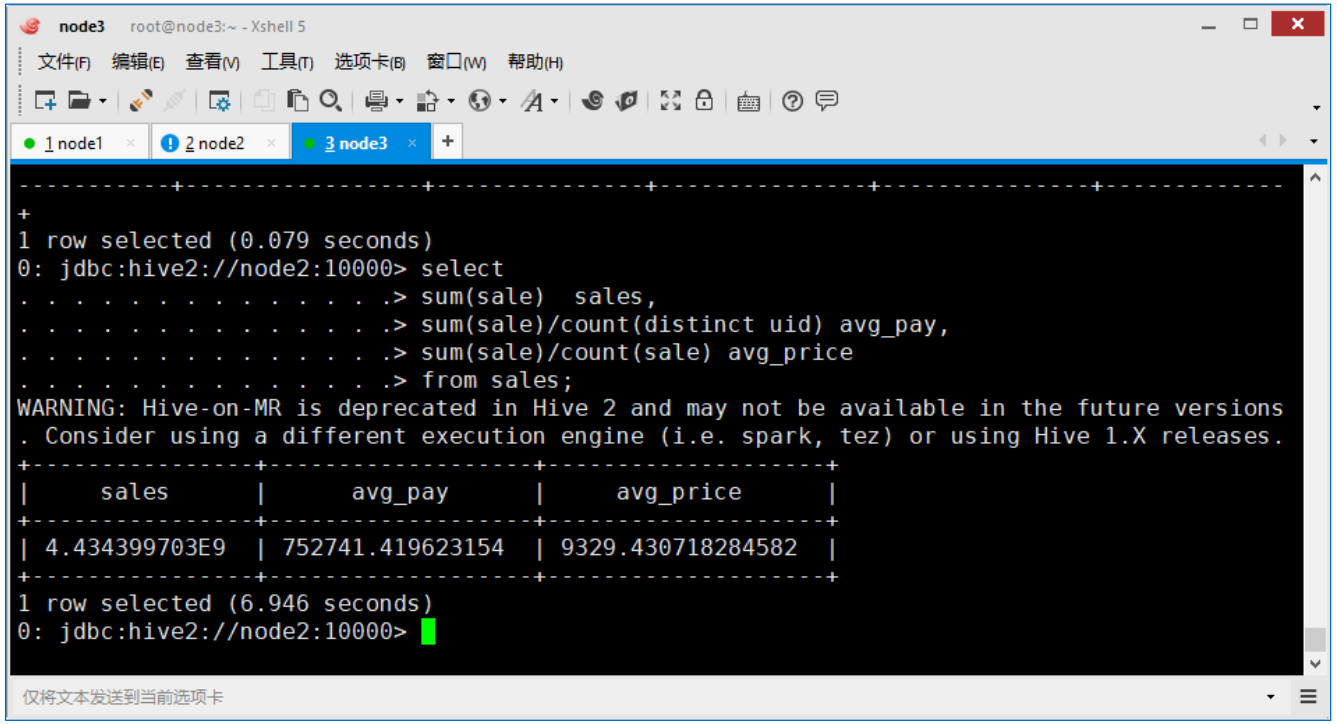


图9-7

3．统计各性别消费情况

HiveQL脚本设计如下：

```
select *,
cast(users/(sum(users) over () ) as decimal (8,2)) as
sex_percent,
cast( sum_pay/(sum(sum_pay) over ()) as
decimal(8,2) ) as pay_percent
from (select  sex ,
count(distinct uid) users ,
cast((sum(sale)/count(distinct uid))*100  as
decimal(8,2)) avg_user,
sum(sale) sum_pay from  sales group by sex
)  as s order by pay_percent desc;
```

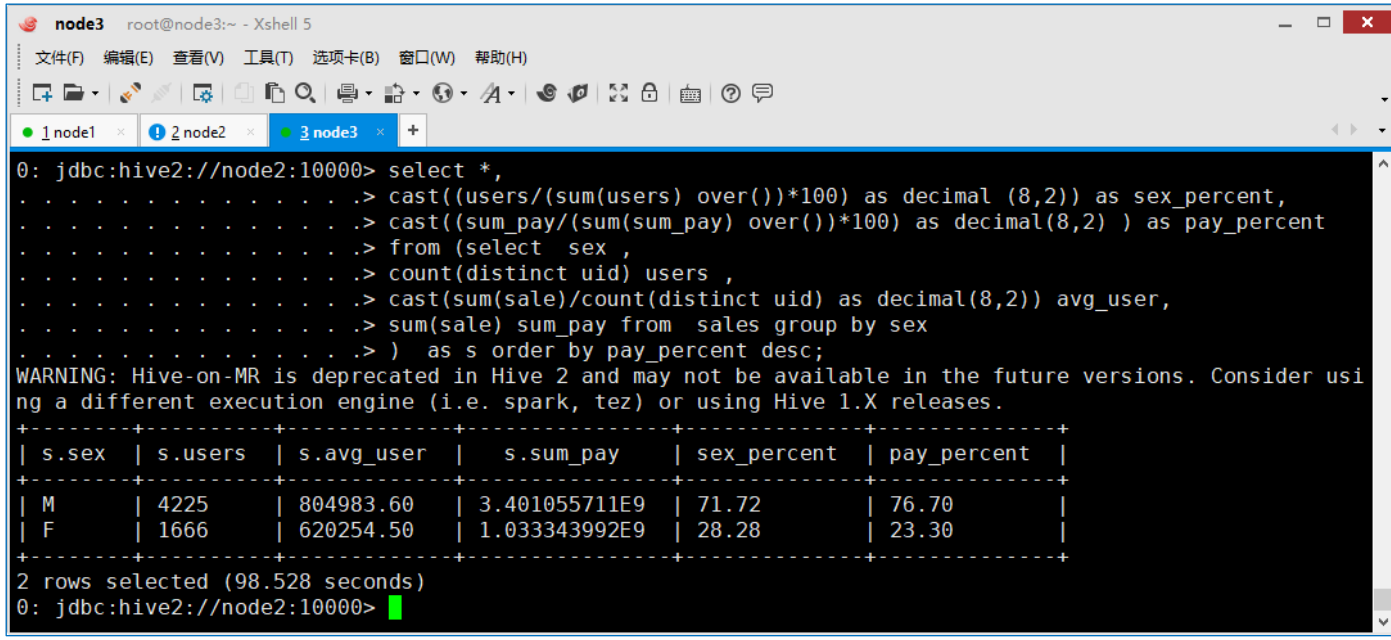


图9-8

运行结果如图9-8

4．统计各年龄段消费情况

HiveQL脚本设计如下：

```
with s as (select age ,
count(distinct uid) users,
sum(sale) sum_pay,
cast(sum(sale)/count(distinct uid)  as
decimal(10,2)) avg_pay
from sales group by age)
select s.*,
cast((users/sum(users) over())*100 as
decimal(10,2)) age_percent,
cast((sum_pay/sum(sum_pay) over())*100 as
decimal(10,2)) pay_percent
from s order by pay_percent desc;
```

运行结果如图9-9

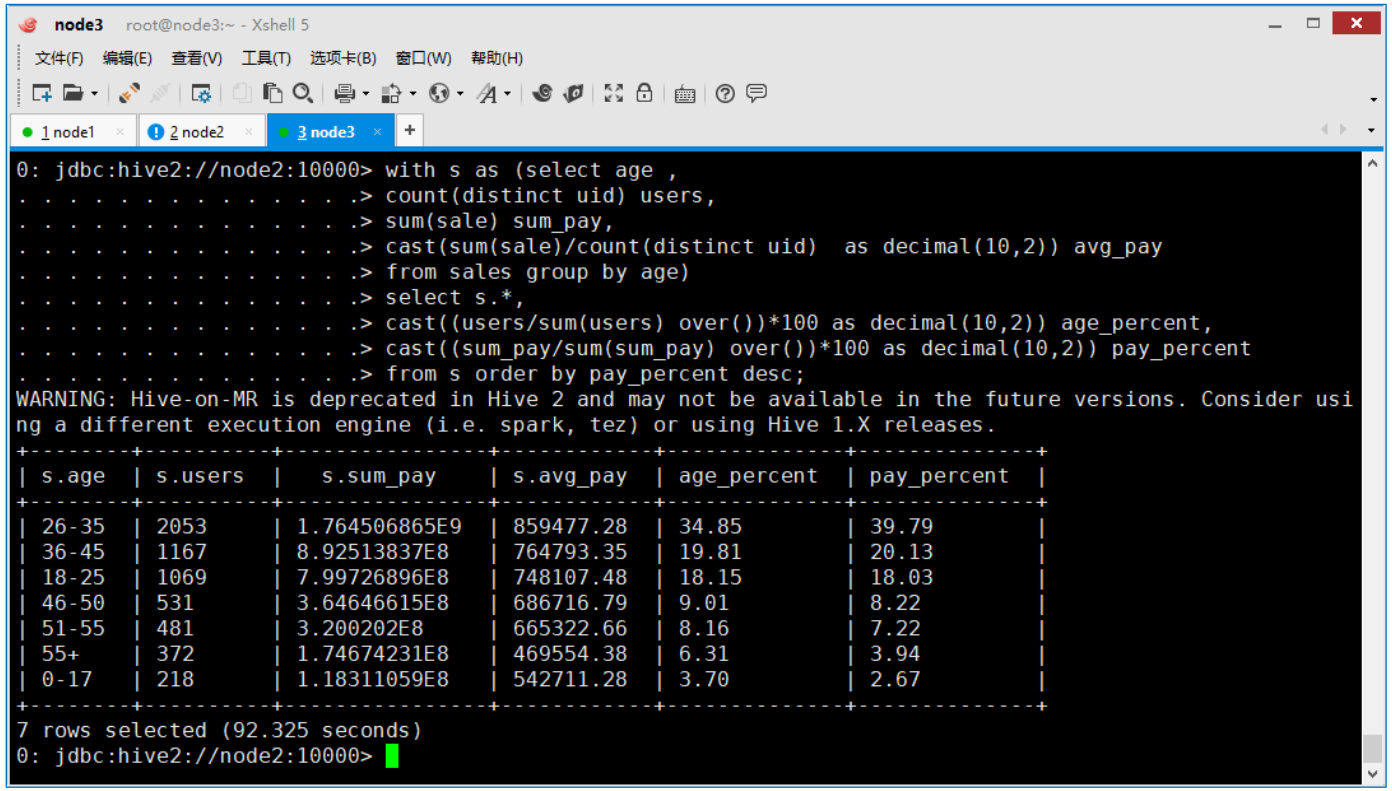


图9-9

5．统计各职业消费情况

HiveQL脚本设计如下：

```
with s as (select vocation job,
count(distinct uid) users,sum(sale) sum_pay,
cast(sum(sale)/count(distinct uid)  as
decimal(10,2)) avg_pay
from sales group by vocation)
select s.*,
cast((users/sum(users) over()  )*100 as
decimal(10,2)) vocation_percent,
cast((sum_pay/sum(sum_pay) over())*100   as
decimal(10,2)) pay_percent
from s order by pay_percent desc;
```

s.job	s.users	s.sum_pay	s.avg_pay	vocation_percent	pay_percent
4	740	5.79293967E8	782829.69	12.56	13.06
0	688	5.53212058E8	804087.29	11.68	12.48
7	669	4.83311039E8	722438.03	11.36	10.90
1	517	3.69899525E8	715472.97	8.78	8.34
17	491	3.42512805E8	697582.09	8.33	7.72
12	376	2.65026164E8	704856.82	6.38	5.98
20	273	2.59410461E8	950221.47	4.63	5.85
14	294	2.24804263E8	764640.35	4.99	5.07
2	256	2.07388223E8	810110.25	4.35	4.68
16	235	2.07143861E8	881463.24	3.99	4.67
6	228	1.64112627E8	719792.22	3.87	3.70
3	170	1.41903147E8	834724.39	2.89	3.20
15	140	1.03227055E8	737336.11	2.38	2.33
10	192	1.01925075E8	530859.77	3.26	2.30
5	111	9.7680872E7	880007.86	1.88	2.20
11	128	9.3352676E7	729317.78	2.17	2.11
19	71	6.4096871E7	902772.83	1.21	1.45
13	140	6.3205687E7	451469.19	2.38	1.43
18	67	5.2439692E7	782681.97	1.14	1.18
9	88	4.745103E7	539216.25	1.49	1.07
8	17	1.3002605E7	764859.12	0.29	0.29

运行结果如图9-10

图9-10

6．统计各婚姻状况消费情况

HiveQL脚本设计如下：

```
with s as(select marriage,
count(distinct uid) users,
sum( sale) sum_pay,
cast(sum( sale)/count(distinct uid) as
decimal(10,2)) avg_pay
from sales group by marriage)
select s.*,
cast((users/sum(users) over() )*100  as
decimal(10,2)) marriage_percent,
cast((sum_pay/sum(sum_pay) over() )*100  as
decimal(10,2)) pay_percent
from s order by pay_percent desc;
```

运行结果如图9-11

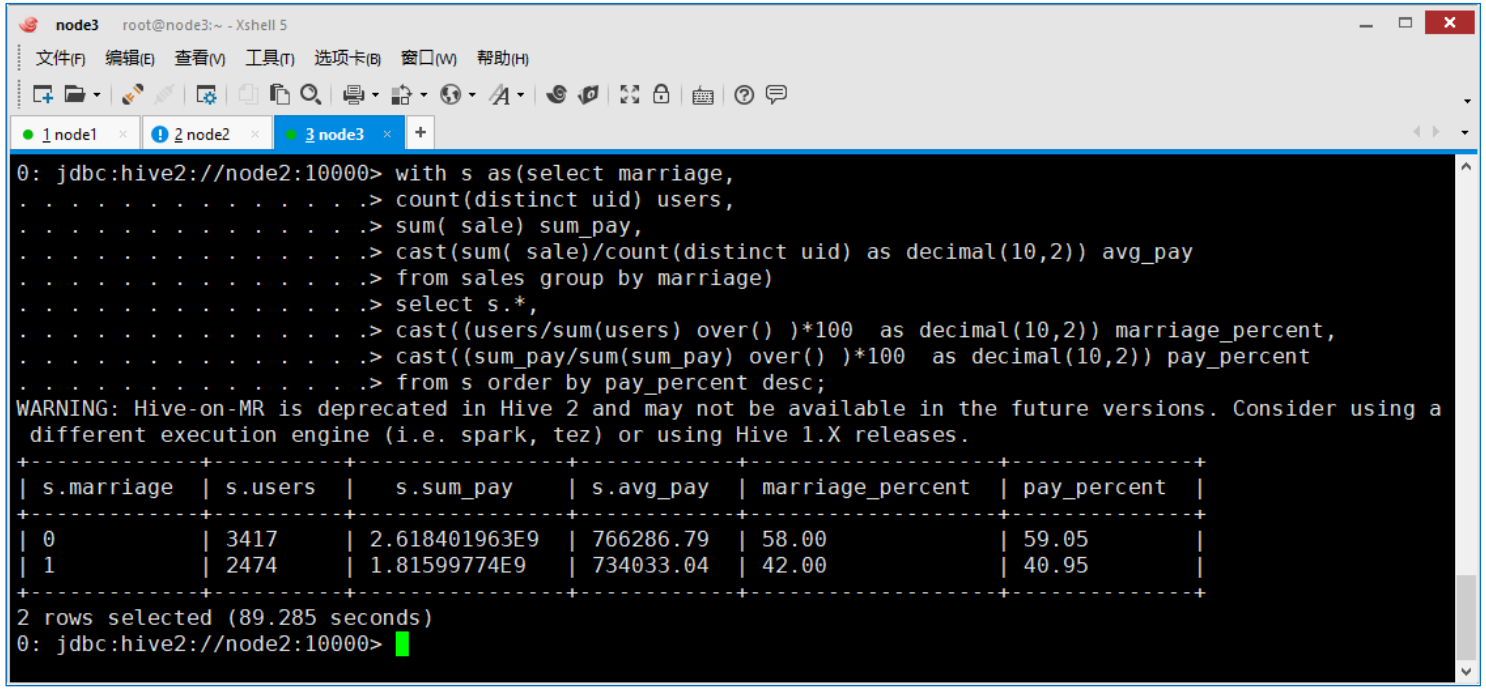


图9-11

7．统计订单量排名前十的产品

HiveQL脚本设计如下：

```
with s as (select pid,
count(pid)  sale_items
from sales group by pid
)select * from (select rank() over(order by
sale_items desc) sales_rank,
s.*,
cast((sale_items/sum(sale_items) over() )*100
as  decimal(10,2)) items_percent
from s) t where sales_rank<=10;
```

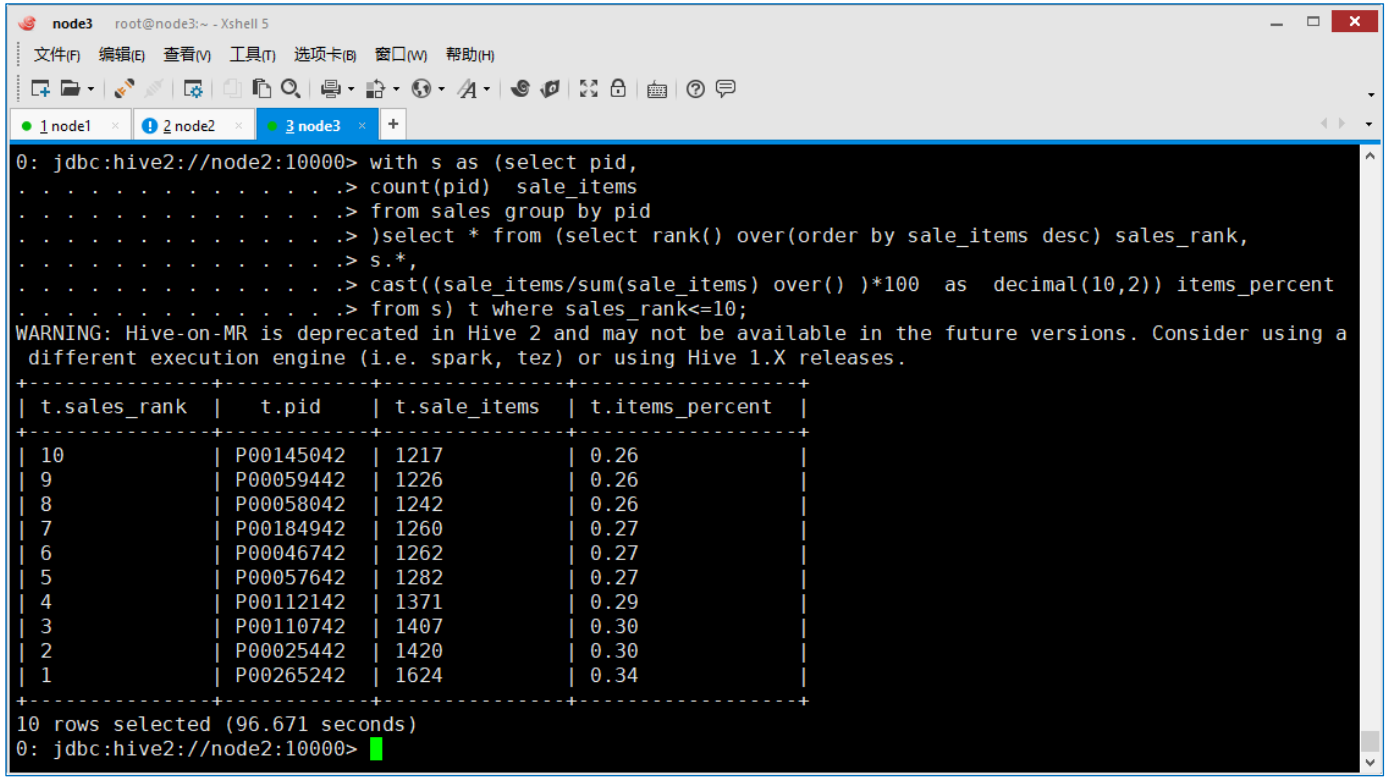


图9-12

运行结果如图9-12

8．统计销售额排名前十的产品

HiveQL脚本设计如下：

```
with s as (select pid,sum( sale)  sum_pay
from sales group by pid)select * from
(select rank() over(order by sum_pay desc)
pay_rank,
s.* ,
cast((sum_pay/sum(sum_pay) over() )*100 as
decimal(10,2) ) pay_percent
from s ) t where pay_rank <=10 order by
pay_rank;
```

运行结果如图9-13

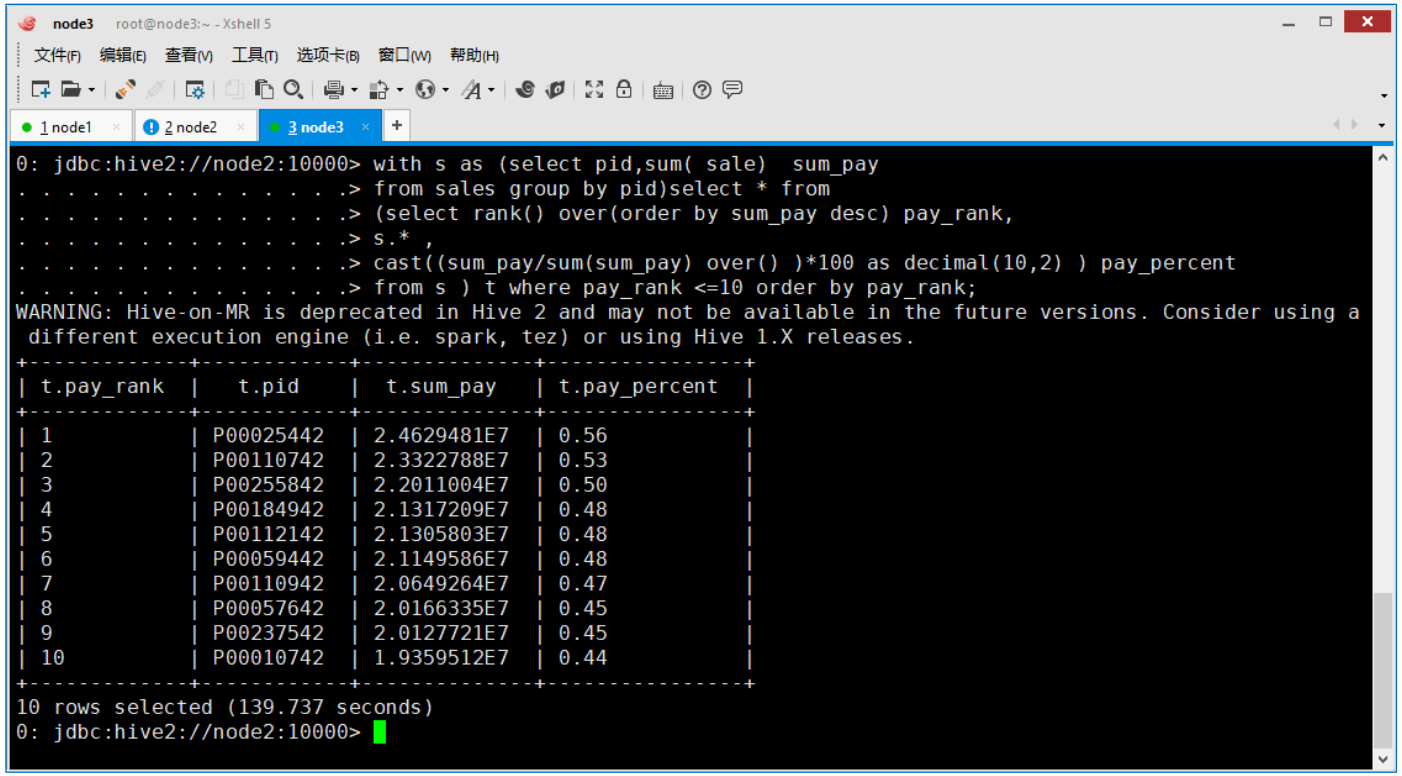


图9-13

9．统计一级产品类别

HiveQL脚本设计如下：

```
with s as (select  level1,
count(pid) sale_items,sum( sale) pay_sum
from  sales group by level1)
select s.*,
cast((sale_items/sum(sale_items) over() )*100
as decimal(10,2)) items_precent,
cast((pay_sum/sum(pay_sum) over() )*100 as
decimal(10,2)) pay_precent
from s order by pay_precent desc;
```

s.level1	s.sale_items	s.pay_sum	items_precent	pay_precent
1	122049	1.660427766E9	25.68	37.44
5	131483	8.20291707E8	27.66	18.50
8	99299	7.44435854E8	20.89	16.79
6	17770	2.81744214E8	3.74	6.35
2	20785	2.33926736E8	4.37	5.28
3	17582	1.77510358E8	3.70	4.00
16	8576	1.26768995E8	1.80	2.86
11	21126	9.9028993E7	4.44	2.23
10	4465	8.8075597E7	0.94	1.99
15	5465	8.069733E7	1.15	1.82
7	3246	5.3254974E7	0.68	1.20
4	10313	2.4051447E7	2.17	0.54
14	1322	1.736803E7	0.28	0.39
18	2696	8031791.0	0.57	0.18
9	357	5526390.0	0.08	0.12
17	505	5123975.0	0.11	0.12
12	3438	4640949.0	0.72	0.10
13	4836	3494597.0	1.02	0.08

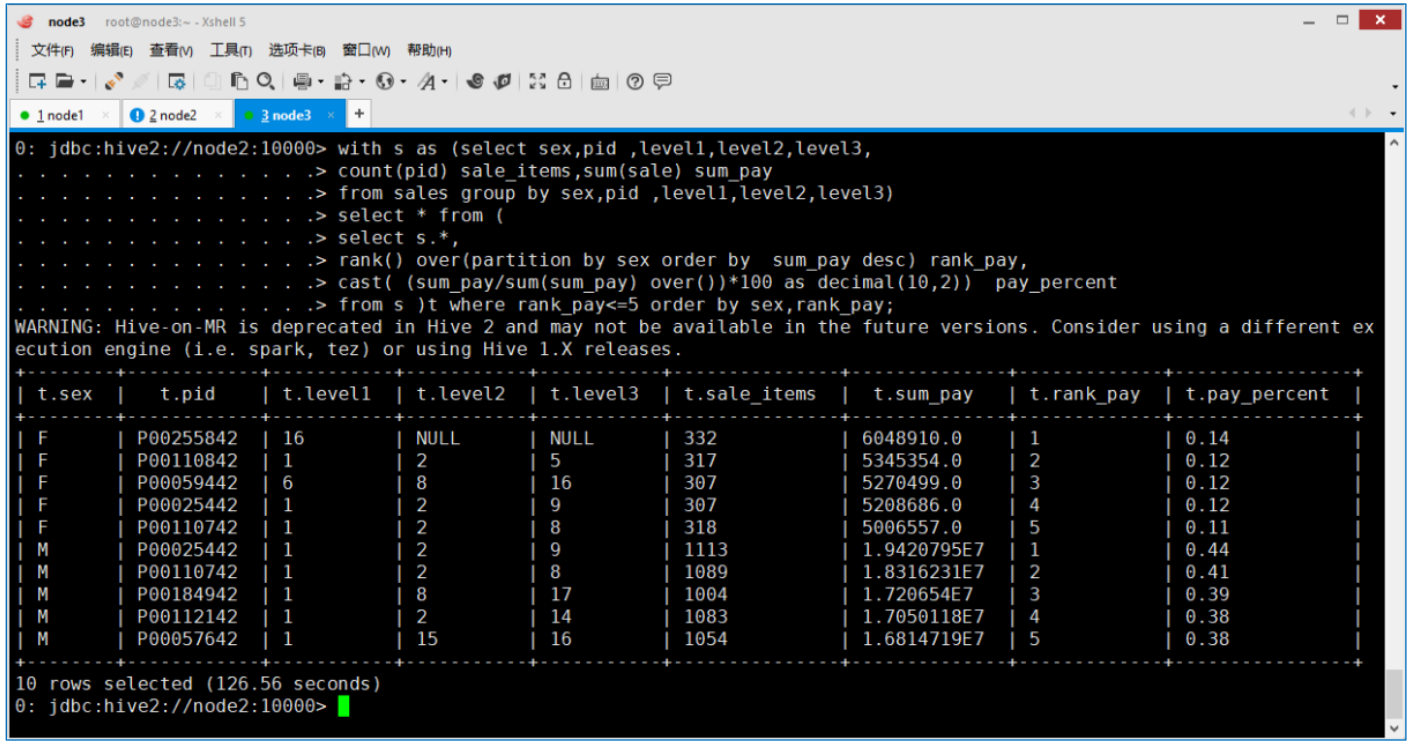
图9-14

运行结果如图9-14

10．统计各性别销售额前五的产品

HiveQL脚本设计如下：

```
with s as (select sex,pid ,level1,level2,level3,
count(pid) sale_items,sum(sale) sum_pay
from sales group by sex,pid ,level1,level2,level3)
select * from (
select s.*,
rank() over(partition by sex order by  sum_pay
desc) rank_pay,
cast( (sum_pay/sum(sum_pay) over())*100 as
decimal(10,2))  pay_percent
from s )t where rank_pay<=5 order by
sex,rank_pay;
```



运行结果如图9-15

图9-15

项目总结

们在项目中首先学习了项目案例的概述和需求分析，项目数据的数据结构，了解字段的含义，其次学习如何数据导入操作，将原始数据导入Hive数据库，最后学习了数据的分析操作方法，运用HiveQL语句实现各项统计分析功能。



感谢您的观看

