A1-20171-SQL-DBS301

DUE DATE: Nothing will be accepted after this date. It will get a zero

**Due BEFORE MIDNIGHT on the FRIDAY of WEEK 6**

**PLAGIARISM**   
When you submit an assignment, you are saying that the submission is your own work and as such you wish to be given credit for the work.

Occasionally, especially when working on a design problem or writing programs (but never on exams or tests!), it may be necessary to ask someone for a small amount of help. You are permitted to do so, provided you meet the following two conditions.

1 You acknowledge the help on the work you hand in, including explaining the work or portion of the work done or assisted by another person.

2 You understand the work that you hand in, so that you could explain the reasoning behind the parts of the work done for you or assisted by another.

Any other assistance by another person constitutes a violation called plagiarism and will be treated as such.

When 2 or more of you work together as a group, only one assignment should be submitted with all the members’ names on it at the time of submission. If two submissions exist, that are essentially the same, then both submissions cannot possibly make the claim that the work is "solely their own". This is a serious matter and is considered plagiarism. It is recorded in accordance with College policy and may result in a penalty such as expulsion from the College. Please do not put yourself in this position. Plagiarism is very easy to detect.

**Anyone not in this list cannot be added to the group later**

Name(s) Student ID(s) Section A, B or C

\_Shan Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_113004154\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_A\_\_\_\_\_\_\_\_

\_Yuyang Gou\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_053187134\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_A\_\_\_\_\_\_\_\_

\_Han Yu\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_052904133\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_A\_\_\_\_\_\_\_\_

\_Qi Bao Zheng\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_047971130\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_A\_\_\_\_\_\_\_\_\_\_\_\_

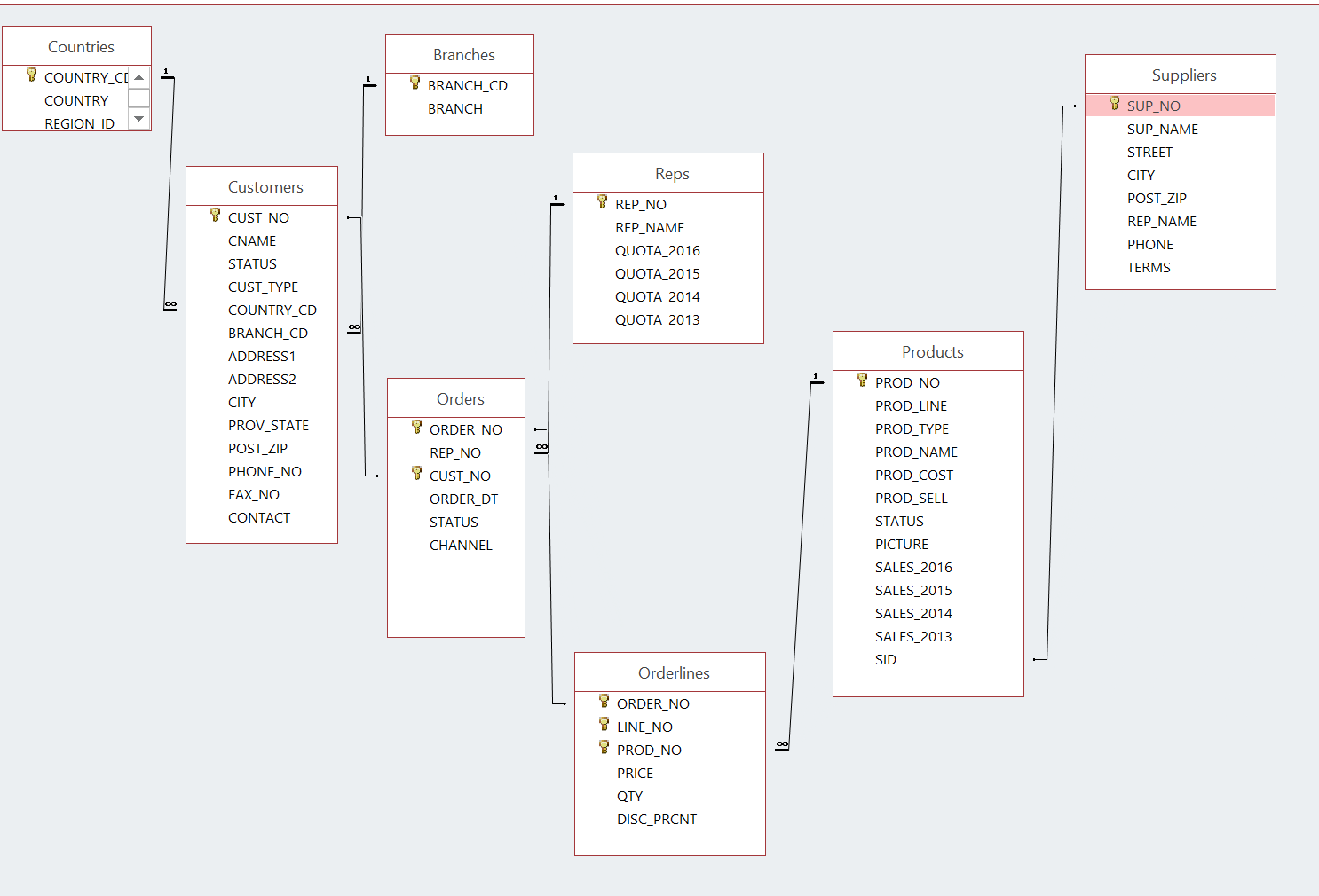
The above acknowledges you understand.

INSTRUCTION:

1 Rename this WORD file A1-xxxxx where xxxxx is replaced with your email account name, not the whole email. If you are in a group pick one of them.

2 When finished email this back to me and CC everyone in your group as proof it was sent.

3 The following tables are used for this assignment. You will be given a script to load.



NOTE:

If any output goes on for more than 100 lines, only cut and paste the first 100 or so. I don’t want to get this word document too big.

1. Display the customer number, customer name and country code for all the customers that are in SPAIN. The country code for Spain is SPA. Please note that you are given SPA, or spa or SpA to use and not Spain.

**SQL:**

select

cust\_no,

cname,

country\_cd

from customers

where upper(country\_cd) = 'SPA';

**OUTPUT:**

CUST\_NO CNAME COUNTRY\_CD

---------------------- ------------------------------ ----------

1095 Supremax Montagna 1 SPA

1096 Supremax Montagna 2 SPA

1097 Supremax Montagna 3 SPA

1098 GO Outlet Madrid SPA

1019 Supremax Montagna 5 SPA

1035 Ultra Sports 1 SPA

6 rows selected

2. How many orders have the product number 40302?

**SQL**

select count(\*) as "Total Orders"

from orders o

join orderlines ol

on o.order\_no = ol.order\_no

and ol.prod\_no= 40302;

**OUTPUT:**

Total Orders

----------------------

22

3. List the customer number, customer name and order number for customers that ordered product 40302. Put result in customer number order.

**SQL:**

select

c.cust\_no,

c.cname,

o.order\_no

from customers c

join orders o

on c.cust\_no = o.cust\_no

join orderlines ol

on o.order\_no = ol.order\_no

and ol.prod\_no = 40302

order by c.cust\_no;

**OUTPUT:**

CUST\_NO CNAME ORDER\_NO

---------------------- ------------------------------ ----------------------

1001 GO Outlet Montreal 67

1005 GO Outlet Boston 153

1019 Supremax Montagna 5 78

1036 Ultra Sports 2 11

1039 Vacation Central 1 65

1041 Vacation Central 3 33

1045 Mountain Madness 3 16

1051 Sportwaren G.m.b.H. 1 139

1062 123 Fitness PTE Ltd 38

1066 Wilderness Wonderment Ltd 164

1069 Andes Camping Supplies 3 157

1071 Lookout Below Ltd 37

1078 Act'N'Up Fitness 4 44

1083 Over the Top Cycles 1 77

1095 Supremax Montagna 1 205

1102 Pro Form Supplies 4 28

1121 GO Outlet Manchester 15

1127 Fresh Air Co 1 135

1129 Fresh Air Co 3 101

1130 Fresh Air Lte 4 120

1139 Fredies Sport Whse 1 194

1148 Juan's Sports 2 2

22 rows selected

4 Display the customer number for Ultra Sports 5.

**SQL:**

select cust\_no

from customers

where cname = 'Ultra Sports 5';

**OUTPUT:**

CUST\_NO

----------------------

1002

5 Display the customer number, customer name, order number, product name, the total dollars for that line. Give that last column the name of TOTAL.

Put the output into customer number order from highest to lowest and display only order numbers less than 75

**SQL:**

select

c.cust\_no,

c.cname,

o.order\_no,

p.prod\_name,

sum(ol.price \* ol.qty) as "TOTAL"

from customers c

join orders o

on c.cust\_no = o.cust\_no

and o.order\_no < 75

join orderlines ol

on o.order\_no = ol.order\_no

join products p

on ol.prod\_no = p.prod\_no

group by c.cust\_no, c.cname, o.order\_no, p.prod\_name

order by c.cust\_no desc;

**OUTPUT:**

CUST\_NO CNAME ORDER\_NO PROD\_NAME TOTAL

---------------------- ------------------------------ ---------------------- ------------------------------ ----------------------

1148 Juan's Sports 2 2 GO Ski Gear Bag 1024

1148 Juan's Sports 2 2 GO Small Waist Pack 18

1148 Juan's Sports 2 2 Microwave Detective 1488

1148 Juan's Sports 2 2 Star Gazer-2 1036

1148 Juan's Sports 2 2 Star Lite 5610

1148 Juan's Sports 2 2 Sun Shelter-8 1890

1145 Florida Sun Sports 3 71 GO Large Waist Pack 96

1145 Florida Sun Sports 3 71 GO Ski Gear Bag 1536

1145 Florida Sun Sports 3 71 Pocket U.V. Alerter 324

1145 Florida Sun Sports 3 71 Pro-Lite Water Filter 165

1145 Florida Sun Sports 3 71 Star Gazer-2 2072

1145 Florida Sun Sports 3 71 Star Lite 165

1136 Hill Street Sports 2 69 GO Headband 640

1136 Hill Street Sports 2 69 Pack n' Hike 262

1136 Hill Street Sports 2 69 Pocket Radon Alerter 2886

1136 Hill Street Sports 2 69 RiverKind Soap 4114

1136 Hill Street Sports 2 69 StarDome 615

1136 Hill Street Sports 2 70 Enviro-T 3330

1136 Hill Street Sports 2 70 GO Large Waist Pack 168

1136 Hill Street Sports 2 70 GO Water Bottle 256

1136 Hill Street Sports 2 70 Star Lite 2475

1136 Hill Street Sports 2 70 StarDome 615

1136 Hill Street Sports 2 70 Sun Shelter-15 5679

1135 Hill Street Sports 1 55 GO Water Bottle 512

1135 Hill Street Sports 1 55 MoonBeam 1920

1135 Hill Street Sports 1 55 Pro-Lite Water Filter 3960

1135 Hill Street Sports 1 55 Star Gazer-2 1036

1135 Hill Street Sports 1 55 Star Gazer-3 555

1135 Hill Street Sports 1 55 Sun Shelter-15 5679

1130 Fresh Air Lte 4 20 Dover-2 333

1130 Fresh Air Lte 4 20 Enviro-Kit 588

1130 Fresh Air Lte 4 20 GO Camp Kettle 777

1130 Fresh Air Lte 4 20 GO Ski Gear Bag 1536

1130 Fresh Air Lte 4 20 MoonLite 84

1130 Fresh Air Lte 4 20 Sun Shelter-8 2652

1130 Fresh Air Lte 4 22 GO Camp Kettle 1092

1130 Fresh Air Lte 4 22 GO Ski Gear Bag 2048

1130 Fresh Air Lte 4 22 MoonBeam 840

1130 Fresh Air Lte 4 22 Pocket Radon Alerter 1911

1130 Fresh Air Lte 4 22 Pro-Lite Water Filter 1815

1130 Fresh Air Lte 4 22 Star Gazer-3 4440

1130 Fresh Air Lte 4 43 Dover-2 1443

1130 Fresh Air Lte 4 43 GO Wristband 128

1130 Fresh Air Lte 4 43 MoonLite 588

1130 Fresh Air Lte 4 43 Pocket Radon Alerter 936

1130 Fresh Air Lte 4 43 RiverKind Soap 4697

1130 Fresh Air Lte 4 43 Star Lite 2475

1130 Fresh Air Lte 4 46 Enviro-Kit 432

1130 Fresh Air Lte 4 46 GO Duffel Bag 896

1130 Fresh Air Lte 4 46 MoonBeam 1560

1130 Fresh Air Lte 4 46 Pack n' Hike 524

1130 Fresh Air Lte 4 46 Pocket Radon Alerter 2418

1130 Fresh Air Lte 4 46 Star Gazer-2 3626

1130 Fresh Air Lte 4 62 GO Ski Gear Bag 1024

1130 Fresh Air Lte 4 62 MoonBeam 960

1130 Fresh Air Lte 4 62 MoonGlow 2451

1130 Fresh Air Lte 4 62 RiverKind Soap 4114

1130 Fresh Air Lte 4 62 StarDome 6150

1130 Fresh Air Lte 4 62 Sun Shelter-8 3408

1128 Fresh Air Co 2 8 GO Cookset 2268

1128 Fresh Air Co 2 8 GO Headband 800

1128 Fresh Air Co 2 8 Pocket Water Filter 3780

1128 Fresh Air Co 2 8 RiverKind Detergent 3024

1128 Fresh Air Co 2 8 Star Gazer-3 4440

1127 Fresh Air Co 1 9 Dover-2 333

1127 Fresh Air Co 1 9 GO Cookset 1728

1127 Fresh Air Co 1 9 GO Water Bottle 512

1127 Fresh Air Co 1 9 MoonBeam 1920

1127 Fresh Air Co 1 9 Pro-Lite Water Filter 3960

1127 Fresh Air Co 1 9 RiverKind Soap 4697

1121 GO Outlet Manchester 15 EnviroSak 216

1121 GO Outlet Manchester 15 GO Large Waist Pack 768

1121 GO Outlet Manchester 15 GO Small Waist Pack 126

1121 GO Outlet Manchester 15 GO Water Bottle 512

1121 GO Outlet Manchester 15 Star Gazer-2 5180

1121 GO Outlet Manchester 15 Star Gazer-3 11100

1121 GO Outlet Manchester 15 Star Lite 1650

1121 GO Outlet Manchester 15 StarDome 6150

1120 GO Outlet London 29 GO Cookset 1512

1120 GO Outlet London 29 GO Wristband 640

1120 GO Outlet London 29 MoonGlow 2064

1120 GO Outlet London 29 Pocket U.V. Alerter 558

1120 GO Outlet London 29 RiverKind Detergent 1806

1120 GO Outlet London 49 Day Tripper 14

1120 GO Outlet London 49 GO Cookset 1512

1120 GO Outlet London 49 GO Headband 320

1120 GO Outlet London 49 Pack n' Hike 262

1120 GO Outlet London 49 Pocket Water Filter 3780

1120 GO Outlet London 49 RiverKind Shampoo 558

1120 GO Outlet London 61 GO Camp Kettle 1533

1120 GO Outlet London 61 GO Wristband 384

1120 GO Outlet London 61 MoonGlow 1677

1120 GO Outlet London 61 Pocket Water Filter 8370

1120 GO Outlet London 61 Pro-Lite Water Filter 165

1120 GO Outlet London 61 Star Lite 5610

1119 GO Outlet Frankfurt 6 GO Camp Kettle 777

1119 GO Outlet Frankfurt 6 GO Ski Gear Bag 1536

1119 GO Outlet Frankfurt 6 MoonBeam 840

1119 GO Outlet Frankfurt 6 Pocket Radon Alerter 1404

1119 GO Outlet Frankfurt 6 Star Gazer-2 1036

1119 GO Outlet Frankfurt 6 Sun Shelter-30 1692

6 Display a count of how many different country codes there are

**SQL:**

select count(distinct(country\_cd)) as "Total Countries"

from customers;

**OUTPUT:**

Total Countries

----------------------

14

7 Find the total dollar value for all orders from London. Each row will show customer name, order number and total dollars for the order. Sort by order number

**SQL:**

select

c.cname,

o.order\_no,

sum(ol.price \* ol.qty) as "Total Dollar Value"

from customers c

join orders o

on c.cust\_no = o.cust\_no

and upper(c.branch\_cd) = 'LON'

join orderlines ol

on o.order\_no = ol.order\_no

group by c.cname, o.order\_no

order by o.order\_no;

**OUTPUT:**

CNAME ORDER\_NO Total Dollar Value

------------------------------ ---------------------- ----------------------

Fresh Air Co 2 8 14312

Rock Steady 3 12 10645

GO Outlet London 29 6580

GO Outlet London 35 10213

GO Outlet London 49 6446

GO Outlet London 61 17739

Fresh Air Co 3 101 3673

GO Outlet London 107 28466

Trees to Seas Ltd 122 5056

Fresh Air Co 3 125 2025

Fresh Air Co 2 147 14667

Trees to Seas Ltd 170 5728

Fresh Air Co 3 174 8495

Fresh Air Co 2 189 9867

Fresh Air Co 2 202 5477

15 rows selected

**Going back to the same tables you have used for labs that came from demobld10g**

8 Display the (a) employee number, (b) full employee name, (c) job and (d) hire date.

- Limit the display to all employees hired in May or November of any year.

- The most recently hired employees are displayed first.

- Exclude people hired in 1994 and 1995.

- Full name should be in the form 🡪 *Lastname, Firstname --* with an alias called *Full Name.*

- Hire date should point to the last day in May, June, July, August or December of that year (NOT to the exact hire date)

- The format is in the form of *May 31st of 1996* –better if there is no big gap between month and 31st

- The hire date column should be called *Start Date*.

**NOTE: Do NOT use a LIKE operator.**

You should display ONE row per output line by limiting the width of the *Full Name* to 25 characters.

**SQL:**

select

employee\_id,

last\_name || ', ' || first\_name as "Full Name",

job\_id,

to\_char(last\_day(hire\_date), 'fmMonth fmDD"th of" YYYY') as "Start Date"

from employees

where

to\_char(hire\_date, 'fmMonth') in ('May', 'June', 'July', 'August', 'December')

and to\_number(to\_char(hire\_date, 'YYYY')) not between 1992 and 1996

order by hire\_date desc;

**OUTPUT:**

EMPLOYEE\_ID Full Name JOB\_ID Start Date

---------------------- ----------------------------------------------- ---------- ----------------------

178 Grant, Kimberely SA\_REP May 31th of 1999

144 Vargas, Peter ST\_CLERK July 31th of 1998

202 Fay, Pat MK\_REP August 31th of 1997

104 Ernst, Bruce IT\_PROG May 31th of 1991

100 King, Steven AD\_PRES June 30th of 1987

9 List the employee number, full name, job and the modified salary for all employees

- whose monthly earning (without the increase) is outside the range $6,000 – $11,000

- and who are employed as a Vice Presidents or Managers (President is not counted here).

- You should use **Wild Card** characters for this.

- the modified salary for a VP will be 30% higher

- and managers a 20% salary increase.

- Sort the output by the top salaries (before this increase).

Heading will be: 🡪 *Employees with Increased Pay*

**The output lines should look like this sample line:**

Employee 101 named Neena Kochhar with Job ID of AD\_VP will have a new salary of $22100

**SQL:**

select

'Employee ' || employee\_id || ' named ' || first\_name || ' ' || last\_name ||

' with Job ID of ' || job\_id || ' will have a new salary of $' ||

case

when upper(job\_id) like '%VP' then salary \* 1.3

else salary \* 1.2

end as "Employees with Increased Pay"

from employees

where

salary not between 6000 and 11000

and upper(job\_id) like '%VP'

or upper(job\_id) like '%MGR'

order by salary desc;

**OUTPUT:**

Employees with Increased Pay

-----------------------------------------------------------------------------------------

Employee 101 named Neena Kochhar with Job ID of AD\_VP will have a new salary of $22100

Employee 102 named Lex De Haan with Job ID of AD\_VP will have a new salary of $22100

Employee 205 named Shelley Higgins with Job ID of AC\_MGR will have a new salary of $14400

10 Display last\_name, job id and salary for all employees who earn more than all lowest paid employees per department that are in locations outside the US.

Exclude President and Vice Presidents from this query.

Sort the output by job id ascending.

If a JOIN is needed you must use a “newer” method (USING/JOIN)

**SQL:**

select

last\_name,

job\_id,

salary

from employees

where

salary > all(

select min(e.salary)

from employees e join departments d on e.department\_id = d.department\_id

join locations l on d.location\_id = l.location\_id

and upper(l.country\_id) <> 'US'

group by e.department\_id

)

and job\_id not like '%VP' and job\_id not like '%PRES'

order by job\_id;

**OUTPUT:**

LAST\_NAME JOB\_ID SALARY

------------------------- ---------- ----------------------

Higgins AC\_MGR 12000

Hunold IT\_PROG 9000

Hartstein MK\_MAN 13000

Zlotkey SA\_MAN 10500

Abel SA\_REP 11000

11 Who are the employees (show last\_name, salary and job) who work either in IT , ACCOUNTING or MARKETING department and earn more than the worst paid person in the SHIPPING department.

Sort the output by the last name alphabetically.

**You need to use ONLY the Subquery method (NO joins allowed).**

**SQL:**

select

last\_name,

salary,

job\_id

from employees

where department\_id in (

select department\_id

from departments

where upper(department\_name) in ('IT', 'ACCOUNTING', 'MARKETING')

)

and salary > (

select min(salary)

from employees

where department\_id = (

select department\_id

from departments

where upper(department\_name) = 'SHIPPING'

)

)

order by last\_name;

**OUTPUT:**

LAST\_NAME SALARY JOB\_ID

------------------------- ---------------------- ----------

Ernst 6000 IT\_PROG

Fay 6000 MK\_REP

Gietz 8300 AC\_ACCOUNT

Hartstein 13000 MK\_MAN

Higgins 12000 AC\_MGR

Hunold 9000 IT\_PROG

Lorentz 4200 IT\_PROG

7 rows selected

12 Display Department\_id, Job\_id and the Lowest salary for this combination but only if that Lowest Pay falls in the range $6000 - $18000.

Exclude people who

(a) work as some kind of *Representative* job from this query and

(b) departments IT and SALES

Sort the output according to the Department\_id and then by Job\_id.

You MUST NOT use the Subquery method.

SQL:

select

e.department\_id,

e.job\_id,

min(e.salary)

from employees e

join departments d

on e.department\_id = d.department\_id

and upper(d.department\_name) not in ('IT', 'SALES')

and upper(e.job\_id) not like upper('%rep')

group by e.department\_id, e.job\_id

having

min(e.salary) >= 6000 and min(e.salary) <= 18000

order by e.department\_id, e.job\_id;

OUTPUT:

DEPARTMENT\_ID JOB\_ID MIN(E.SALARY)

---------------------- ---------- ----------------------

20 MK\_MAN 13000

90 AD\_VP 17000

110 AC\_ACCOUNT 8300

110 AC\_MGR 12000