**SQL Training**

### Attention

1. The <XXXX> used in this article represents the employee ID. Please replace it with your own employee ID.

### Create data object

#### Create tables and table indexes

Table ：CUX\_OM\_CUSTOMERS\_XXXX

For storing customer information。

| Field | Type | Nullable | Default Value | Description |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| CUSTOMER\_ID | BIGINT | N |  |  |
| CUSTOMER\_NUMBER | VARCHAR(30) | N |  |  |
| CUSTOMER\_NAME | VARCHAR(100) | N |  |  |
| TELEPHONE | VARCAHR(50) | Y |  |  |
| START\_DATE\_ACTIVE | DATE | Y |  |  |
| END\_DATE\_ACTIVE | DATE | Y |  |  |
| CREATED\_BY | BIGINT | N |  |  |
| CREATION\_DATE | DATE | N |  |  |
| LAST\_UPDATED\_BY | BIGINT | N |  |  |
| LAST\_UPDATE\_DATE | DATE | N |  |  |
| LAST\_UPDATE\_LOGIN | BIGINT | Y |  |  |

Create indexes：

| Index | Unique | Field | Description |
| --- | --- | --- | --- |
|  |  |  |  |
| CUX\_OM\_CUSTOMERS\_XXXX\_U1 | Y | CUSTOMER\_ID |  |
| CUX\_OM\_CUSTOMERS\_XXXX\_U2 | Y | CUSTOMER\_NUMBER |  |
| CUX\_OM\_CUSTOMERS\_XXXX\_U3 | Y | CUSTOMER\_NAME |  |

Table：CUX\_OM\_ITEMS\_XXXX

For storing product information。

| Field | Type | Nullable | Default Value | Description |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| ITEM\_ID | BIGINT | N |  |  |
| ITEM\_NUMBER | VARCHAR(30) | N |  |  |
| ITEM\_NAME | VARCHAR(100) | N |  |  |
| UOM\_CODE | VARCHAR(25) | Y |  |  |
| START\_DATE\_ACTIVE | DATE | Y |  |  |
| END\_DATE\_ACTIVE | DATE | Y |  |  |
| CREATED\_BY | BIGINT | N |  |  |
| CREATION\_DATE | DATE | N |  |  |
| LAST\_UPDATED\_BY | BIGINT | N |  |  |
| LAST\_UPDATE\_DATE | DATE | N |  |  |
| LAST\_UPDATE\_LOGIN | BIGINT | Y |  |  |

Create indexes：

| Index | Unique | Field | Description |
| --- | --- | --- | --- |
|  |  |  |  |
| CUX\_OM\_ITEMS\_XXXX\_U1 | Y | ITEM\_ID |  |
| CUX\_OM\_ITEMS\_XXXX\_U2 | Y | ITEM\_NUMBER |  |

table：CUX\_OM\_HEADERS\_XXXX

For storing sales order header information。

| Field | Type | Nullable | DefaultValue | Description |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| HEADER\_ID | BIGINT | N |  |  |
| ORG\_ID | BIGINT | Y | 101 |  |
| ORDER\_NUMBER | VARCHAR(30) | N |  |  |
| CUSTOMER\_ID | BIGINT | Y |  |  |
| ORDER\_DATE | DATE | N |  |  |
| ORDER\_STATUS | VARCHAR(30) | Y |  | status list:  UNAPPROVED  APPROVED  CANCELLED  CLOSED |
| DESCRIPTION | VARCHAR(240) | Y |  |  |
| CREATED\_BY | BIGINT | N |  |  |
| CREATION\_DATE | DATE | N |  |  |
| LAST\_UPDATED\_BY | BIGINT | N |  |  |
| LAST\_UPDATE\_DATE | DATE | N |  |  |
| LAST\_UPDATE\_LOGIN | BIGINT | Y |  |  |
| ATTRIBUTE\_CATEGORY | VARCHAR(30) | Y |  |  |
| ATTRIBUTE1 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE2 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE3 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE4 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE5 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE6 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE7 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE8 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE9 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE10 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE11 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE12 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE13 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE14 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE15 | VARCHAR(150) | Y |  |  |

Create Indexes：

| Index | Unique | Field | Description |
| --- | --- | --- | --- |
|  |  |  |  |
| CUX\_OM\_HEADERS\_XXXX \_U1 | Y | HEADER\_ID |  |
| CUX\_OM\_HEADERS\_XXXX \_U2 | Y | ORDER\_NUMBER |  |
| CUX\_OM\_HEADERS\_XXXX \_N1 |  | CUSTOMER\_ID |  |

Table：CUX\_OM\_LINES\_XXXX

For storing sales order line information。

| Field | Type | Nullable | Default Value | Description |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| LINE\_ID | BIGINT | N |  |  |
| HEADER\_ID | BIGINT | N |  |  |
| LINE\_NUMBER | VARCHAR(30) | N |  |  |
| ITEM\_ID | BIGINT | N |  |  |
| UNIT\_PRICE | DECIMAL(10,2) | Y |  |  |
| QUANTITY | DECIMAL(10,2) | Y |  |  |
| DESCRIPTION | VARCHAR(240) | Y |  |  |
| ADDITION1 | VARCHAR(150) | Y |  |  |
| ADDITION2 | VARCHAR(150) | Y |  |  |
| ADDITION3 | VARCHAR(150) | Y |  |  |
| ADDITION4 | VARCHAR(150) | Y |  |  |
| ADDITION5 | VARCHAR(150) | Y |  |  |
| CREATED\_BY | BIGINT | N |  |  |
| CREATION\_DATE | DATE | N |  |  |
| LAST\_UPDATED\_BY | BIGINT | N |  |  |
| LAST\_UPDATE\_DATE | DATE | N |  |  |
| LAST\_UPDATE\_LOGIN | BIGINT | Y |  |  |
| ATTRIBUTE\_CATEGORY | VARCHAR(30) | Y |  |  |
| ATTRIBUTE1 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE2 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE3 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE4 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE5 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE6 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE7 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE8 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE9 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE10 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE11 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE12 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE13 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE14 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE15 | VARCHAR(150) | Y |  |  |

Create Indexes：

| Index | Unique | Field | Description |
| --- | --- | --- | --- |
|  |  |  |  |
| CUX\_OM\_LINES\_XXXX \_U1 | Y | LINE\_ID |  |
| CUX\_OM\_LINES\_XXXX \_N1 |  | HEADER\_ID |  |
| CUX\_OM\_LINES\_XXXX \_N2 |  | ITEM\_ID |  |

#### Import data

excel：《1 – Table Data.xls》

1．CUX\_OM\_CUSTOMERS\_XXXX

2．CUX\_OM\_ITEMS\_XXXX

4．CUX\_OM\_HEADERS\_XXXX

5．CUX\_OM\_LINES\_XXXX

### Questions

#### NO.01

Please create the above objects in the database: table, table index。

#### NO.02

Please import the corresponding table data in the file: "1 – Table Data.xlsx" into the table。

#### NO.03

Create a view：CUX\_OM\_HEADERS\_V

CREATE VIEW CUX\_OM\_HEADERS\_V AS

SELECT

h.HEADER\_ID, h.ORDER\_NUMBER, h.CUSTOMER\_ID, h.ORDER\_DATE, h.ORDER\_STATUS, h.DESCRIPTION,

h.CREATED\_BY, h.CREATION\_DATE, h.LAST\_UPDATED\_BY, h.LAST\_UPDATE\_DATE, h.LAST\_UPDATE\_LOGIN,

h.ATTRIBUTE\_CATEGORY, h.ATTRIBUTE1, h.ATTRIBUTE2, h.ATTRIBUTE3, h.ATTRIBUTE4, h.ATTRIBUTE5,

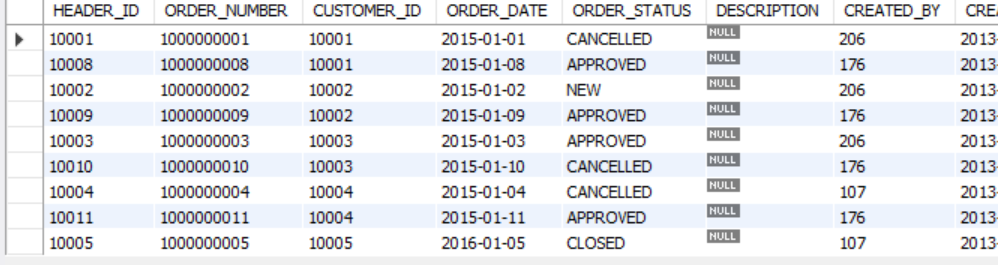
h.ATTRIBUTE6, h.ATTRIBUTE7, h.ATTRIBUTE8, h.ATTRIBUTE9, h.ATTRIBUTE10, h.ATTRIBUTE11,

h.ATTRIBUTE12, h.ATTRIBUTE13, h.ATTRIBUTE14, h.ATTRIBUTE15, c.CUSTOMER\_NUMBER,

c.CUSTOMER\_NAME, c.TELEPHONE

FROM cux\_om\_headers\_46319 h

INNER JOIN cux\_om\_customers\_46319 c ON h.CUSTOMER\_ID = c.CUSTOMER\_ID;



* view：CUX\_OM\_HEADERS\_V

| Field | Type | Nullable | DefaultValue | Description |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| HEADER\_ID | BIGINT | N |  |  |
| ORDER\_NUMBER | VARCHAR(30) | N |  |  |
| CUSTOMER\_ID | BIGINT | Y |  |  |
| ORDER\_DATE | DATE | N |  |  |
| ORDER\_STATUS | VARCHAR(30) | Y |  | status list:  UNAPPROVED  APPROVED  CANCELLED  CLOSED |
| DESCRIPTION | VARCHAR(240) | Y |  |  |
| CREATED\_BY | BIGINT | N |  |  |
| CREATION\_DATE | DATE | N |  |  |
| LAST\_UPDATED\_BY | BIGINT | N |  |  |
| LAST\_UPDATE\_DATE | DATE | N |  |  |
| LAST\_UPDATE\_LOGIN | BIGINT | Y |  |  |
| ATTRIBUTE\_CATEGORY | VARCHAR(30) | Y |  |  |
| ATTRIBUTE1 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE2 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE3 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE4 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE5 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE6 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE7 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE8 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE9 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE10 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE11 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE12 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE13 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE14 | VARCHAR(150) | Y |  |  |
| ATTRIBUTE15 | VARCHAR(150) | Y |  |  |
| CUSTOMER\_NUMBER | VARCHAR(30) |  |  |  |
| CUSTOMER\_NAME | VARCHAR(100) |  |  |  |
| TELEPHONE | VARCHAR(50) | Y |  |  |

#### NO.04

Write SQL to display the following information and sort it by customer\_number and order\_number in ascending order。

SELECT

h.ORDER\_NUMBER, c.CUSTOMER\_NUMBER, c.CUSTOMER\_NAME, h.ORDER\_DATE, h.ORDER\_STATUS, h.DESCRIPTION

FROM cux\_om\_headers\_46319 h

INNER JOIN cux\_om\_customers\_46319 c ON h.CUSTOMER\_ID = c.CUSTOMER\_ID

ORDER BY c.CUSTOMER\_NUMBER ASC, h.ORDER\_NUMBER ASC;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| order\_number | customer\_number | customer\_name | order\_date | order\_status | description |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

#### NO.05

Write SQL to display order and order line information, sorted by order number and line number in ascending order (including orders with all statuses)

SELECT

h.ORDER\_NUMBER, c.CUSTOMER\_NAME, h.ORDER\_DATE, h.ORDER\_STATUS, l.LINE\_NUMBER, i.ITEM\_NUMBER, i.ITEM\_NAME

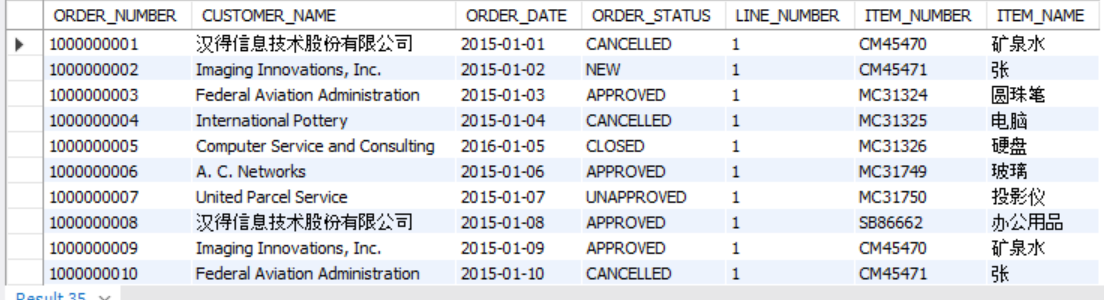
FROM cux\_om\_headers\_46319 h

INNER JOIN cux\_om\_customers\_46319 c ON h.CUSTOMER\_ID = c.CUSTOMER\_ID

INNER JOIN cux\_om\_lines\_46319 l ON h.HEADER\_ID = l.HEADER\_ID

INNER JOIN cux\_om\_items\_46319 i ON l.ITEM\_ID = i.ITEM\_ID

ORDER BY h.ORDER\_NUMBER ASC, l.LINE\_NUMBER ASC;



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| order\_number | customer\_name | order\_date | order\_status | line\_number | item\_number | item\_name |
|  |  |  |  |  |  |  |

#### NO.06

Write SQL to display products purchased by customers

SELECT

c.CUSTOMER\_NUMBER, c.CUSTOMER\_NAME, i.ITEM\_NUMBER, i.ITEM\_NAME, i.UOM\_CODE

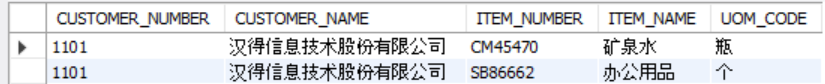
FROM cux\_om\_customers\_46319 c

INNER JOIN cux\_om\_headers\_46319 h ON c.CUSTOMER\_ID = h.CUSTOMER\_ID

INNER JOIN cux\_om\_lines\_46319 l ON h.HEADER\_ID = l.HEADER\_ID

INNER JOIN cux\_om\_items\_46319 i ON l.ITEM\_ID = i.ITEM\_ID

WHERE c.CUSTOMER\_NUMBER LIKE '1101';



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| customer\_number | customer\_number | item\_number | item\_name | uom\_code |
| 1101 | 汉得信息技术股份有限公司 | SB86662 | 办公用品 | 个 |

#### NO.07

Write SQL to count all orders (excluding canceled orders) and calculate the average sales unit price and the highest and lowest unit price by product.

SELECT

i.ITEM\_NUMBER, i.ITEM\_NAME, AVG(l.UNIT\_PRICE) AS AVERAGE\_UNIT\_PRICE,

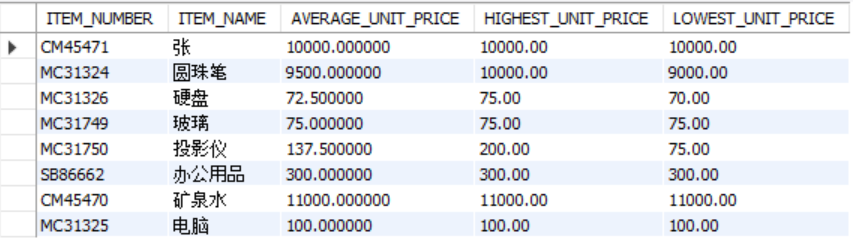
MAX(l.UNIT\_PRICE) AS HIGHEST\_UNIT\_PRICE, MIN(l.UNIT\_PRICE) AS LOWEST\_UNIT\_PRICE

FROM cux\_om\_lines\_46319 l

INNER JOIN cux\_om\_headers\_46319 h ON l.HEADER\_ID = h.HEADER\_ID

INNER JOIN cux\_om\_items\_46319 i ON l.ITEM\_ID = i.ITEM\_ID

WHERE h.ORDER\_STATUS NOT LIKE 'CANCELLED'

GROUP BY i.ITEM\_NUMBER, i.ITEM\_NAME;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| item\_number | item\_name | average\_unit\_price | highest\_unit\_price | lowest\_unit\_price |
|  |  |  |  |  |

#### NO.08

Write SQL to query customer information about canceled orders.

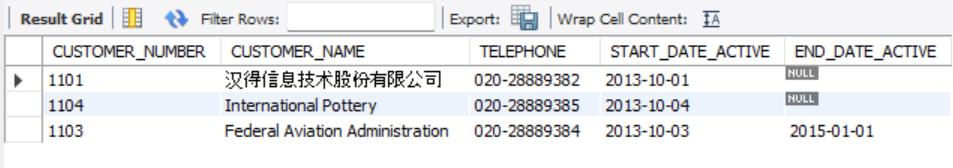
SELECT

c.CUSTOMER\_NUMBER, c.CUSTOMER\_NAME, c.TELEPHONE, c.START\_DATE\_ACTIVE, c.END\_DATE\_ACTIVE

FROM cux\_om\_headers\_46319 h

INNER JOIN cux\_om\_customers\_46319 c ON h.CUSTOMER\_ID = c.CUSTOMER\_ID

WHERE h.ORDER\_STATUS LIKE 'CANCELLED';



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| customer\_number | customer\_name | telephone | start\_date\_active | end\_date\_active |
|  |  |  |  |  |

#### NO.09

Write SQL to count the company's total sales in 2015. (All orders with status CANCELLED are excluded from statistics)

SELECT

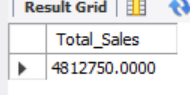
SUM(l.UNIT\_PRICE \* l.QUANTITY) AS Total\_Sales

FROM cux\_om\_headers\_46319 h

INNER JOIN cux\_om\_lines\_46319 l ON h.HEADER\_ID = l.HEADER\_ID

WHERE YEAR(h.ORDER\_DATE) LIKE '2015'

AND h.ORDER\_STATUS NOT LIKE 'CANCELLED';



|  |
| --- |
| Total Sales |
|  |

#### NO.10

Write a script to add an index to the table: CUX\_OM\_ITEMS\_XXXX: a normal index, and the applicable field: ITEM\_NAME.

CREATE INDEX cux\_om\_items\_46319\_N1 ON cux\_om\_items\_46319 (ITEM\_NAME);

