Analytical Functions are Cool!

Presented by: Michael Davis
CEO OmegaSoft, LLC
Agenda

• What are Analytic Functions
• Analytic Function Syntax
• Analytic Function Examples
  • Simple Example
  • Top N Ranking Functions
  • Lag/Lead Functions
  • Pivot queries
• Question & Answer
What are Analytic Functions

- Extension of ANSI SQL
- Major categories
  - Grouping Sets
  - With Clause
  - Top n Ranking
  - Aggregate Window
What are Analytic Functions

• *Extension of ANSI SQL (Cont)*
• **Major catagories**
  – First and Last
  – Reporting Functions
  – Lag and Lead
  – Case and Width_Bucket Functions
Analytic Function syntax

• **Function Clause**
  – The function clause begins with a SQL function. A function statement can include any one of the 33 Analytical Functions, such as SUM, COUNT, ETC.
Analytic Function syntax

• Over Statement
  – Select MAX() OVER ()
  – The OVER() statement signals a start of an Analytic function. That is what differentiates an Analytical Function from a regular Oracle SQL function.
Analytic Function syntax

• *Partitioning clause*
  
  – Select MAX() OVER(partition by field1).

The portioning clause is used to setup the group of data that the Analytic function would be applied to. Though, it’s akin to a Group by statement in a SQL query, it is applied to the result set of a query, and not a group.
Analytic Function syntax

• **Order by Clause**
  
  – Select MAX() OVER(Partition by field order by)

Order by specify the order of the window in the group by statement. The Order by clause is a keyword in the Oracle Analytic syntax that is requirement for using some Analytic functions.
Analytic Function syntax

• **Window Clause**
  – species the relative rows to which the Analytic function needs to be applied.
Analytic Function Examples

• **Simple Example**

```sql
SQL> Select * 
1      from ( Select cust_name, sum(clm_amt)clm_amt 
2            from  customer 
3            group by cust_name 
4            order by clm_amt desc desc ) v 
5         ) 
6       and rownum < 11 
7 /
```
## Analytic Function Examples

### Simple Example result

<table>
<thead>
<tr>
<th>CUST_NAME</th>
<th>CLM_AMT</th>
</tr>
</thead>
<tbody>
<tr>
<td>XYZ</td>
<td>100,000,000</td>
</tr>
<tr>
<td>Lexus Corp</td>
<td>80,000,000</td>
</tr>
<tr>
<td>First America</td>
<td>60,000,000</td>
</tr>
<tr>
<td>Yelp</td>
<td>78,000,000</td>
</tr>
<tr>
<td>ABC</td>
<td>75,000,000</td>
</tr>
<tr>
<td>Omega Int.</td>
<td>74,000,000</td>
</tr>
<tr>
<td>S Corp</td>
<td>70,000,000</td>
</tr>
<tr>
<td>Acme</td>
<td>25,000,000</td>
</tr>
<tr>
<td>Sun Enterprise</td>
<td>23,000,000</td>
</tr>
<tr>
<td>Film studio</td>
<td>17,000,000</td>
</tr>
</tbody>
</table>
Analytic Function Examples

• **Analytic Version**

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
       2  SUM(clm_amt) OVER (partition by cust_name) clm_amt
       3 /

  

  SQL> select cust_name,
### Analytic Function Examples

**Analytic Example result**

<table>
<thead>
<tr>
<th>CUST_NAME</th>
<th>CLM_AMT</th>
</tr>
</thead>
<tbody>
<tr>
<td>XYZ</td>
<td>100,000,000</td>
</tr>
<tr>
<td>Lexus Corp</td>
<td>80,000,000</td>
</tr>
<tr>
<td>First America</td>
<td>60,000,000</td>
</tr>
<tr>
<td>Yelp</td>
<td>78,000,000</td>
</tr>
<tr>
<td>ABC</td>
<td>75,000,000</td>
</tr>
<tr>
<td>Omega Int.</td>
<td>74,000,000</td>
</tr>
<tr>
<td>S Corp</td>
<td>70,000,000</td>
</tr>
<tr>
<td>Acme</td>
<td>25,000,000</td>
</tr>
<tr>
<td>Sun Enterprise</td>
<td>23,000,000</td>
</tr>
<tr>
<td>Film studio</td>
<td>17,000,000</td>
</tr>
</tbody>
</table>
Analytic Function Examples

• **LEAD/LAG Functions**

```sql
SQL> select cust_name,
       2   LAG(clm_amt,1,0) OVER (partition by cust_name order by clm_amt)
       3     CLM_AMT_PREF

       3   LEAD(clm_amt,1,0) OVER (partition by cust_name order by clm_amt)
       4     CLM_AMT_NEXT

       4   from member
       5 /
```
## Analytic Function Examples

**LEAD/LAG Function result**

<table>
<thead>
<tr>
<th>CUST_NAME</th>
<th>CLM_AMT</th>
<th>CLM_AMT_PREV</th>
<th>CLM_AMT_NEXT</th>
</tr>
</thead>
<tbody>
<tr>
<td>XYZ</td>
<td>100,000,000</td>
<td>0</td>
<td>80,000,000</td>
</tr>
<tr>
<td>Lexus Corp</td>
<td>80,000,000</td>
<td>100,000,000</td>
<td>60,000,000</td>
</tr>
<tr>
<td>First America</td>
<td>60,000,000</td>
<td>80,000,000</td>
<td>78,000,000</td>
</tr>
<tr>
<td>Yelp</td>
<td>78,000,000</td>
<td>60,000,000</td>
<td>75,000,000</td>
</tr>
<tr>
<td>ABC</td>
<td>75,000,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Omega Int.</td>
<td>74,000,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S Corp</td>
<td>70,000,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acme</td>
<td>25,000,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sun Enterprise</td>
<td>23,000,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Film studio</td>
<td>17,000,000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Analytic Function Examples

• **Top N ranking Function**

```sql
SQL> select cust_name, clm_amt,
  2  dense_rank() OVER (partition by clm_amt)
  3  dr
  4  from ( select cust_name,
  5  SUM(clm_amt) OVER (partition by cust_name) clm_amt
  6  )
  7 /
```
Analytic Function Examples

- **Top N ranking result**

<table>
<thead>
<tr>
<th>CUST_NAME</th>
<th>CLM_AMT</th>
<th>DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>XYZ</td>
<td>100,000,000</td>
<td>1</td>
</tr>
<tr>
<td>Lexus Corp</td>
<td>80,000,000</td>
<td>2</td>
</tr>
<tr>
<td>First America</td>
<td>60,000,000</td>
<td>3</td>
</tr>
<tr>
<td>Yelp</td>
<td>78,000,000</td>
<td>4</td>
</tr>
<tr>
<td>ABC</td>
<td>75,000,000</td>
<td>5</td>
</tr>
<tr>
<td>Omega Int.</td>
<td>74,000,000</td>
<td>6</td>
</tr>
<tr>
<td>S Corp</td>
<td>70,000,000</td>
<td>7</td>
</tr>
<tr>
<td>Acme</td>
<td>25,000,000</td>
<td>8</td>
</tr>
<tr>
<td>Sun Enterprise</td>
<td>23,000,000</td>
<td>9</td>
</tr>
<tr>
<td>Film studio</td>
<td>17,000,000</td>
<td>10</td>
</tr>
</tbody>
</table>
The end...

• **Email Address:**
  – mdavis@omegasoftgroup.com
Questions and Answers

• Got Questions?
References


• Tom Kyte (2001). Expert One-on-One Oracle: Wrox Press Ltd.