**HW3 due by Wednesday Oct. 12 by 11:59 PM**

The create\_expense.sql script (you can find it on Blackboard in the Homework 3 folder) creates a database which contains the 7 tables described below. The data contains a tracking system for expense reports filed by employees at a manufacturing company. **Please watch the Panopto video for week 5 before doing this Homework 3.**

|  |  |
| --- | --- |
| **employees** | **Field Description** |
| Ssn (pk) | Unique SSN ID# for employee |
| First\_name | Employee first name |
| Last\_name | Employee last\_name |
| Dept (fk) | Dept ID# |
| Start\_year | Year of employment |

|  |  |  |
| --- | --- | --- |
| **trips** | | **Field Description** |
| Employee (pk, fk) | | SSN of employee travelling |
| Trip\_ID (pk) | | Unique Trip ID# |
| Start\_date | | Start date of trip |
| End\_date | | End date of trip |
| Reason\_code (fk) | | Code for reason for trip |
|  | |  |
| **expenses** | **Field Description** | | |
| Employee (pk, fk)  Trip\_id (pk, fk) | SSN of employee travelling  Unique Trip ID# | | |
| Expense\_seq (pk) | Sequence# for expense report line item | | |
| Account\_no (fk) | Account number for line item | | |
| Gross\_amount | Gross dollar amount of line item | | |
| tax | Sales tax (if applicable) of line item | | |
|  |  | | |

|  |  |
| --- | --- |
| **dept\_codes** | **Field Description** |
| Dept\_ID (pk) | Dept ID# |
| Dept\_name | Name of department |

|  |  |  |  |
| --- | --- | --- | --- |
| **Reason\_codes** | | **Field Description** | |
| Reason\_code (pk) | | Reason ID# | |
| Reason\_description | | Description of reason for trip | |
|  | |  | |
| **account\_codes** | **Field Description** | |
| Account\_no (pk) | Account ID# | |
| Account\_description | Description of account | |
| Account\_type | Category of account | |
| **reimbursements** | **Field Description** | |
| Employee (pk, fk) | SSN of employee travelling | |
| Trip\_id (pk, fk) | Unique Trip ID# | |
| Auditor | Auditor last name | |
| Reimbursement\_amount | Amount of reimbursement | |
| Reimbursement\_date | Date of reimbursement | |

Diagram

Description automatically generated

Please put all of your work into **this** **single Word doc**.

1. (60 points) Design and create a data warehouse for the Expense database. The decisions about which fields to include and how to aggregate the data are left to you. You do not need to include every single data point from the 7 tables given. Use your judgement as to what will be interesting/useful for the organization. But please make sure that you pull (combine) data from **at least four tables** and compute relevant aggregate statistics. Please see many examples from class lectures and you may adapt those codes for your purpose (for this dataset).

**Submit a screenshot of the first 25 rows of your data warehouse (paste into this Word document) and the SQL code that you used to create it. Please copy and paste your SQL code into this Word document.**

**use expense;**

**SET sql\_mode = (SELECT REPLACE(@@SQL\_MODE, "ONLY\_FULL\_GROUP\_BY", ""));**

**CREATE OR REPLACE VIEW FourTableComb AS**

**SELECT em.ssn,first\_name, last\_name, total\_expense,**

**tr.number\_of\_trips, ROUND(sum(DISTINCT re.reimbursement\_amount),2) AS total\_reimbursement, COUNT(DISTINCT re.reimbursement\_amount) AS reimbursement\_num,**

**COALESCE(max(re.reimbursement\_date), "Haven't reimbursed yet") AS last\_reimbursement\_date, ROUND(total\_expense-sum(DISTINCT re.reimbursement\_amount),2) AS Amount\_not\_reimbursed**

**FROM employees em**

**JOIN (SELECT t.employee, t.trip\_id, str\_to\_date(t.start\_date,"%m/%d/%Y") as start\_date, str\_to\_date(t.end\_date,"%m/%d/%Y") as end\_date, COUNT(\*) AS number\_of\_trips**

**FROM trips t**

**GROUP BY t.employee) as tr**

**ON em.ssn = tr.employee**

**JOIN (SELECT e.employee, COUNT(DISTINCT e.trip\_id) as num\_of\_trips, e.expense\_seq, e.account\_no,**

**ROUND(SUM(gross\_amount),2) as total\_expense**

**FROM expenses e**

**GROUP BY e.employee) as ex**

**ON tr.employee = ex.employee**

**JOIN (SELECT r.employee, r.trip\_id, auditor, ROUND(reimbursement\_amount,2) as reimbursement\_amount, str\_to\_date(reimbursement\_date,"%m/%d/%Y") as reimbursement\_date**

**FROM reimbursements r**

**) as re**

**ON re.employee = tr.employee**

**AND re.employee = ex.employee**

**GROUP BY em.ssn, first\_name, last\_name**

**ORDER BY Amount\_not\_reimbursed DESC, total\_expense DESC;**

**图形用户界面, 表格

描述已自动生成**

2. (40 points) Create **four** SQL queries on your data warehouse that answer interesting questions. At least two queries should be more than simple queries. For example, more complex queries could include Joins, a Group By element or a subquery or use some aggregate functions and summary calculations (see examples in the class lectures’ slides).

**Submit a copy of each query SQL code (paste into this Word document), and the screenshot of each query results (or the first 25 rows if it is longer) and a one or two sentence description of the question your SQL code was addressing and what you found in the results.**

1. Since there are no built-in function to describe the distribution of numbers and the difficulty of comparing the median and mean is high, this code is supposed to calculate the difference of the number of one's total expense which is lower than the average expense to that which is larger than the average expense, aiming at studying the potential skewness of the distribution. And by doing this query, I find that the number of the lower part is obviously larger than that of the larger part, which means that the distribution of expense is likely to be right skewed.

SELECT(SELECT COUNT(\*)

FROM FourTableComb

WHERE total\_expense/number\_of\_trips <

(SELECT AVG(total\_expense/number\_of\_trips) FROM FourTableComb)) -

(SELECT COUNT(\*)

FROM FourTableComb

WHERE total\_expense/number\_of\_trips >

(SELECT AVG(total\_expense/number\_of\_trips) FROM FourTableComb)) AS Diff\_Lower\_to\_Higher;

表格

描述已自动生成

1. To find out the success ratio of reimbursement, and how many percentage of money could be imbursed all the time and what the average percentage of money could be reimbursed each time, we use the following code, and we find that the success rate of reimbursement is 68.7%, the total money reimbursed is 55.4% and we can derive that the average money reimbursed from the cost per time is 80.7%.

SELECT AVG(reimbursement\_num/number\_of\_trips) AS reimburse\_rate, AVG(Amount\_not\_reimbursed/total\_expense) AS total\_reimburse\_money\_rate,AVG(Amount\_not\_reimbursed/total\_expense)/ AVG(reimbursement\_num/number\_of\_trips) AS reimburse\_money\_rate\_per\_time

FROM FourTableComb;

表格

低可信度描述已自动生成

1. To find out which auditor has a loosest and tightest review process, we use the data of the employee which has a top 10 reimbursement amount to check which one reimburse the most and less money for employee. And we find that Kanel has the highest total reimbursement amount, which is about 40791.8 and Luterek has the lowest total reimbursement amount, which is about 21117.6.

SELECT re.auditor, sum(DISTINCT re.reimbursement\_amount) as total\_reimbursement\_amount

FROM

(SELECT ssn, total\_reimbursement/total\_expense as reimbursement\_ratio

FROM FourTableComb

WHERE total\_expense > 10000

ORDER BY reimbursement\_ratio DESC LIMIT 10

) as f

JOIN reimbursements re ON re.employee = f.ssn

WHERE re.reimbursement\_amount > 0

GROUP BY re.auditor

ORDER BY total\_reimbursement\_amount DESC;

表格

描述已自动生成

1. In addition, supposing today is 2017-12-31, to find out how long hasn't the employee reimbursed and which employees hasn't reimbursed for at least 6 months. We find out that there are 22 people who hasn't reimbursed for at least 6 months and the longest one is Teodor Lynch and the longest time is 312 days

SELECT ssn,first\_name, last\_name,DATEDIFF(STR\_TO\_DATE('2017-12-31',"%Y-%m-%d"), STR\_TO\_DATE(last\_reimbursement\_date,"%Y-%m-%d")) as Time\_since\_last\_time

FROM FourTableComb

WHERE DATEDIFF(STR\_TO\_DATE('2017-12-31',"%Y-%m-%d"), STR\_TO\_DATE(last\_reimbursement\_date,"%Y-%m-%d")) > 180

ORDER BY Time\_since\_last\_time DESC;

图形用户界面, 表格

中度可信度描述已自动生成