**Solution To Case Study**

Based on the data provided by ARISOFT Corporation which contained 8 tables: customers, employees, offices, orders, payments, product lines, and products. As a database administrator following is the solution for all the scenarios given:

**Customer Credit Limit Update:**

ARISOFT Corporation wants to implement a system where the credit limit of a customer is automatically updated based on the total amount of payments received for that customer. Design an SQL query or procedure to calculate the total payments received for each customer and update their credit limit accordingly.

Solution:

delimiter $

create procedure Customer\_Credit\_Limit\_Update()

begin

#declare variables

declare done boolean default false;

declare custid int;

declare totalpayment decimal(10,2);

#declare cursor

declare mycursor cursor for

select customernumber,sum(amount)

from payments

group by customernumber;

#declare error handler

declare continue handler for not found set done = true;

#open cursor

open mycursor;

#fetch the rows and update the customers table

while not done do

fetch mycursor into custid, totalpayment;

update customers

set creditlimit = creditlimit + totalpayment

where customernumber = custid;

end while;

#close cursor

close mycursor;

end $

call Customer\_Credit\_Limit\_Update();

Explanation: First, we create a cursor that extracts the total payment done by each customer from the payments table. Then we read the individual row using fetch commands of the cursor and update the creditlimit column of the customers table as is expected in this scenario.

**Employee Promotion Check:**

ARISOFT Corporation wants to identify employees who are eligible for promotion based on their job performance. Create an SQL query or procedure to check if an employee has achieved a specified sales target (consider the orders and orderdetails tables) and, if so, update their job title to indicate a promotion.

**Office Territory Update:**

The territories assigned to each office need to be updated based on the country of the office location. Design an SQL query or procedure to automatically assign the correct territory code to each office based on its country. Use the offices table for this operation.

**Product Stock Monitoring:**

ARISOFT Corporation wants to monitor product stock levels and generate alerts when the quantity in stock falls below a specified threshold. Create an SQL query or trigger that automatically notifies the relevant stakeholders when a product's stock level becomes critical.

**Product Line Description Analysis:**

The marketing team at ARISOFT Corporation is interested in analyzing the effectiveness of product line descriptions. Develop an SQL query to retrieve the average length of product line descriptions (textDescription column in the productlines table) and identify which product lines have descriptions above or below the average length.

**Customer Order History:**

ARISOFT Corporation wants to provide customers with a summary of their order history, including the total amount spent and the number of orders placed. Design an SQL query or procedure to retrieve this information for a specific customer.

**Identify Late Shipments:**

The logistics department needs a way to identify orders with late shipments. Develop an SQL query or procedure to list orders that have a status of 'Shipped' but where the shipped date is later than the required date.

**Product Vendor Analysis:**

The procurement team wants to analyze the distribution of products among different vendors. Create an SQL query to retrieve the count of products supplied by each vendor (productVendor column in the products table) and order the results by the count in descending order.

**Employee Reporting Hierarchy:**

ARISOFT Corporation wants to ensure that the reporting hierarchy of employees is correctly represented. Design an SQL query or procedure to validate that the reportsTo field in the employees table accurately reflects the organizational reporting structure.

**Customer Payment History:**

The finance department needs a comprehensive report on customer payment history. Create an SQL query to retrieve the payment details for a specific customer, including check numbers, payment dates, and amounts.

**Note: Provide SQL code snippets or procedures to implement each solution, and explain any assumptions made during the design process.**