Employee Payroll System implementation with Oracle PL/SQL

Business Scenario and Functional Scope for Payroll Automation

The business scenario involves managing employee payments in a smooth and efficient way using an automated system. The Employee Payroll System is designed to handle tasks like storing employee details, calculating salaries, processing payments, managing tax deductions, and creating payroll reports. It helps reduce manual work, ensures correct salary calculations, and follows tax rules. The system supports different pay grades and includes overtime, bonuses, allowances and other deductions. It can process payroll for individual employees or for many at once, depending on what the business needs.

This system saves time and lowers the chance of mistakes by automating the entire payroll process. It creates clear reports for both employees and managers, showing payment history, earnings, and tax details. No matter what the size of the company, the system is flexible and useful, helping businesses run their payroll tasks faster and more accurately.

The database consists of three main tables,

1. **Payroll -** Records monthly payroll transactions including basic pay, overtime, bonuses, deductions, taxes, and net salary.
2. **Employees** - Stores employee information including personal details, department, position, salary grade, and base salary.
3. **Salary_Grades** - Defines different salary levels with minimum and maximum salary ranges and associated tax rates.

	PAYROLL_ID	EMP_ID	PAY_PERIOD	BASIC_PAY	OVERTIME_HOURS	OVERTIME_PAY	BONUS	ALLOWANCES	DEDUCTIONS	TAX_AMOUNT	NET_SALARY	PAYMENT_DA
1	10001	1001	01-2023	5000	5	625	0	200	100	860.75	4864.25	01-FEB-23
2	10002	1002	01-2023	6250	0	0	500	300	150	1380	5520	01-FEB-23
3	10003	1003	01-2023	4583.33	2	229.17	0	150	120	727.88	4114.62	01-FEB-23
4	10004	1004	01-2023	4833.33	0	0	200	180	100	768.5	4344.83	01-FEB-23
5	10005	1005	01-2023	7666.67	0	0	1000	500	200	2291.67	6675	01-FEB-23

	EMP_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE	HIRE_DATE	DEPARTMENT	POSITION	SALARY_GRADE	BASE_SALARY
1	1001	Kamal	Perera	kamal.perera@gmail.com	077-1234567	15-JAN-20	IT	Developer	2	60000
2	1002	Nadeesha	Fernando	nadeesha.fernando@yahoo.com	071-9876543	05-MAR-21	HR	HR Manager	3	75000
3	1003	Tharindu	Jayasinghe	tharindu.j@outlook.com	075-4567890	12-JUN-19	Finance	Accountant	2	55000
4	1004	Ruwani	Gunasekara	ruwani.gunasekara@gmail.com	076-1122334	08-AUG-18	Marketing	Marketing Specialist	2	58000
5	1005	Sachintha	Dissanayake	sachintha.d@gmail.com	070-3344556	20-NOV-17	IT	Senior Developer	4	92000

	GRADE_ID	GRADE_NAME	MIN_SALARY	MAX_SALARY	TAX_PERCENTAGE
1	1	Entry Level	30000	45000	10
2	2	Junior	45000.01	65000	15
3	3	Mid-Level	65000.01	85000	20
4	4	Senior	85000.01	110000	25
5	5	Executive	110000.01	200000	30

Fig. 1.PL/SQL table showing sample data entries

Implemented PL/SQL Components

Procedures

1) proc_emp_monthly_payroll

Purpose - Processes monthly payroll for a specific employee.

This procedure calculates and records monthly payroll for an individual employee. It takes the emp ID, pay period, overtime hours, bonus, allowances, and deductions as input parameters. The procedure performs several validation checks, calculates the basic pay, overtime pay, tax amount and net salary and then creates a payroll record.

- Input validation for employee existence, pay period format and overtime hours
- Calculation of monthly basic pay from annual salary
- Overtime pay calculation based on hourly rate
- Tax calculation based on salary grade
- Net salary calculation considering all components
- Comprehensive error handling with custom exceptions

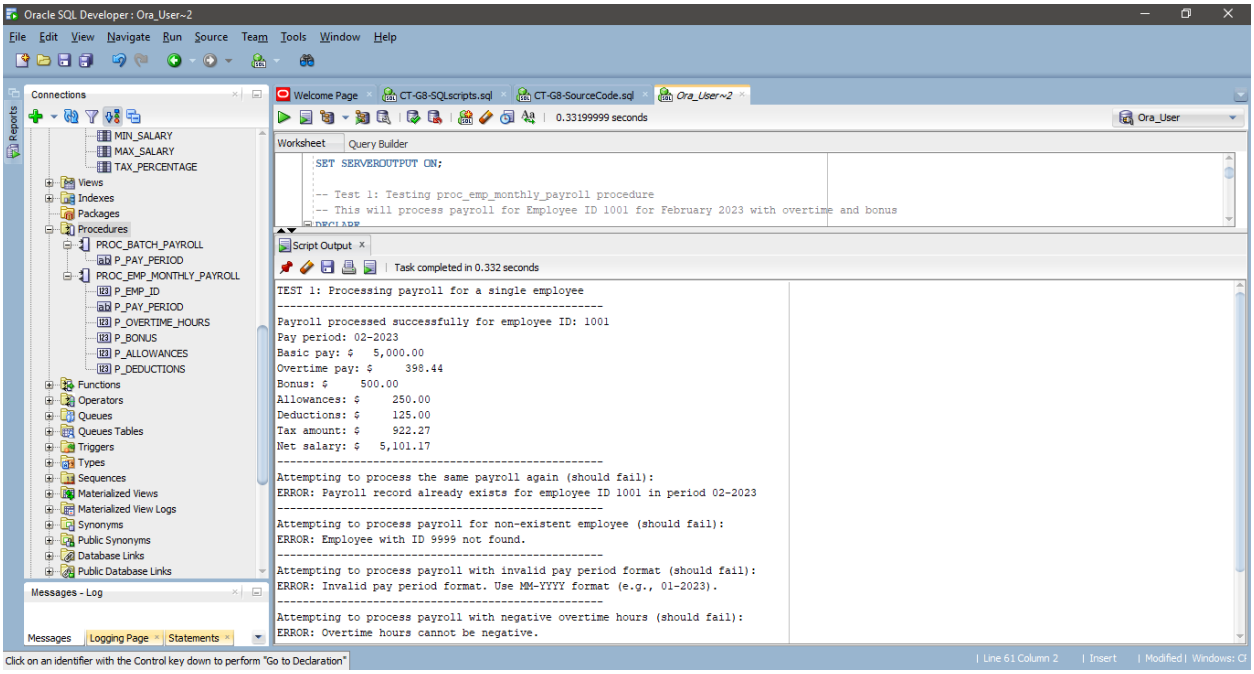

Fig. 2.Employee monthly payroll processing.

2) proc_batch_payroll

Purpose - Processes payroll for all employees for a specific pay period.

This procedure performs batch processing of payroll for all active employees in the database for a given pay period. It utilizes a cursor to iterate through all employees and processes each one individually, handling any errors that might occur during processing without stopping the entire batch.

- Cursor based batch processing of all employees
- Record types for structured data handling
- Validation of pay period format
- Exception handling for each employee individually
- Skip logic for employees with existing payroll records
- Detailed reporting of processed employees, skipped employees and any errors

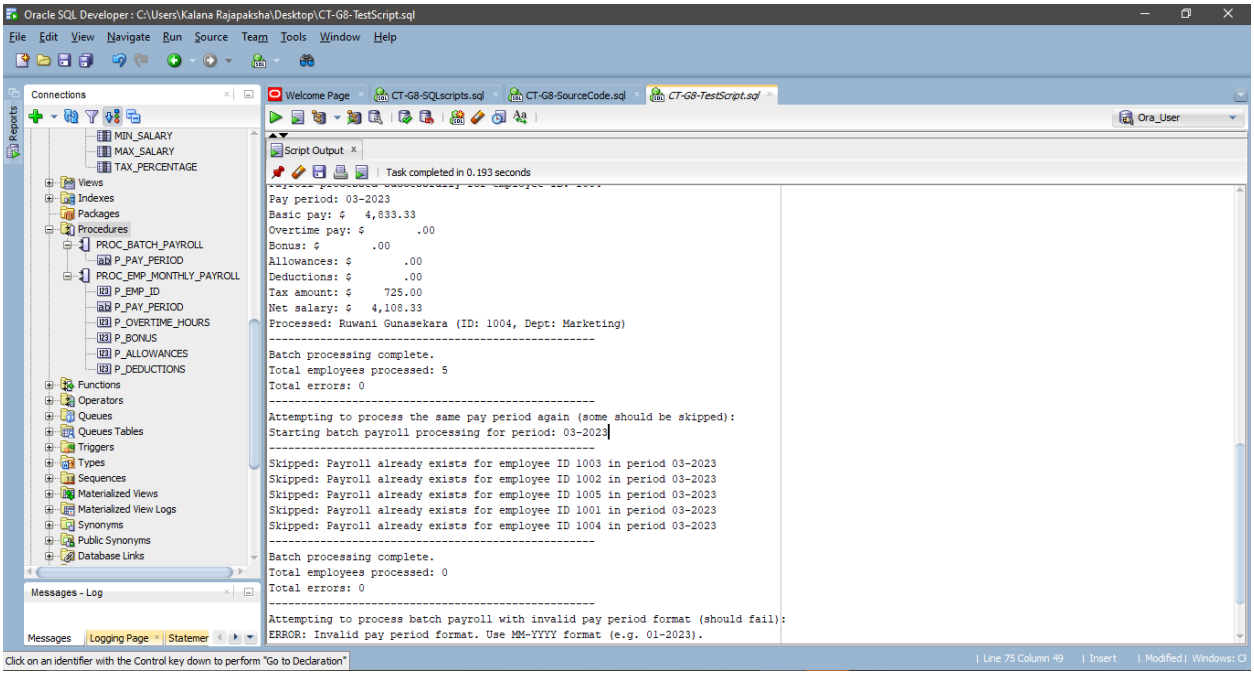

Fig. 3.Batch payroll processing procedure.

Functions

1) func_calculate_annual_tax

Purpose - Calculates the total tax paid by an employee in a specific year.

This function retrieves all payroll records for a specified employee and year, sums up the tax amounts, and returns the total annual tax. It uses a cursor to iterate through relevant payroll records and handles various error conditions.

- Use Cursor computation of annual tax
- Validation of employee existence and year format
- Special handling for cases with no payroll records
- Conditional Processing Based on Payroll Status
- Detailed error messages for different error conditions

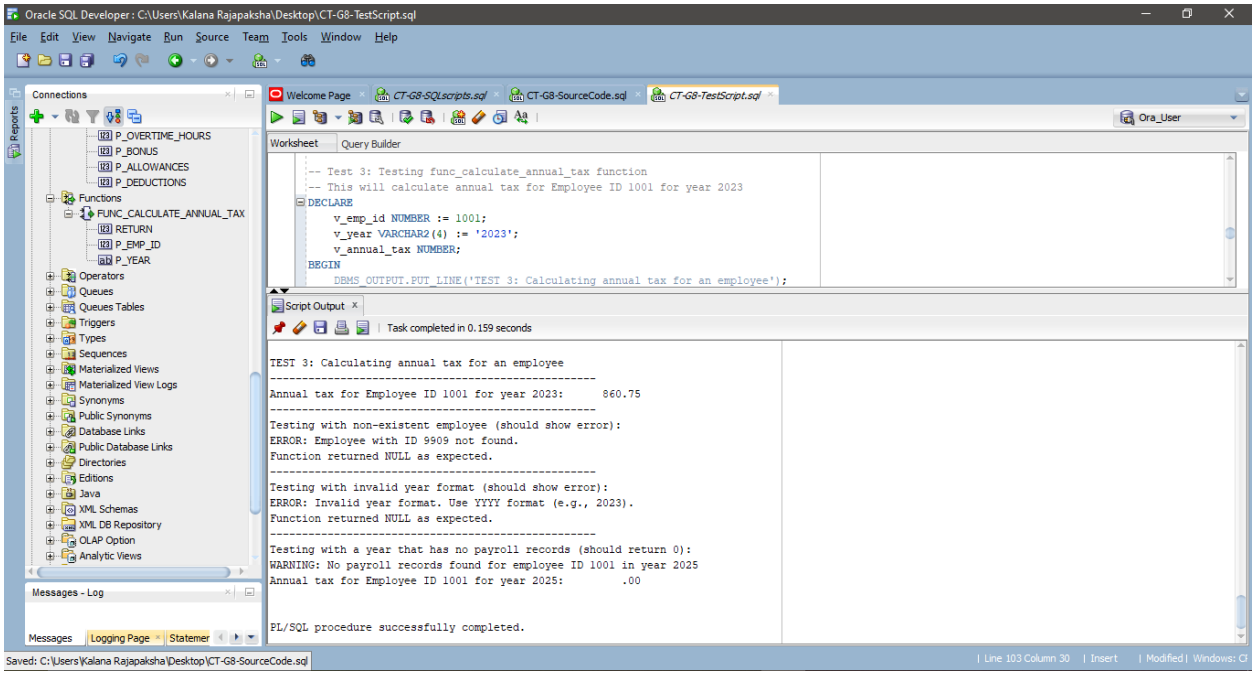

Fig. 4.Function to calculate an employee's total annual tax.

2) func_get_total_earnings

Purpose - Calculates the total earnings for a specific payroll record.

This function takes a payroll ID and finds the related payroll information. It adds up the employee's basic pay, overtime, bonus and any allowances to get the total earnings. If some values are missing, it treats them as zero to avoid calculation errors. If no payroll record is found or if an unexpected problem occurs, the function shows a message and returns zero.

- Adds up all income parts such as basic pay, bonus, overtime and allowances
- Handles missing values by treating them as zero
- Checks if the payroll record exists
- Shows a clear message if no record is found
- Handles unexpected issues safely and returns a default value
- Provides helpful messages for easier troubleshooting

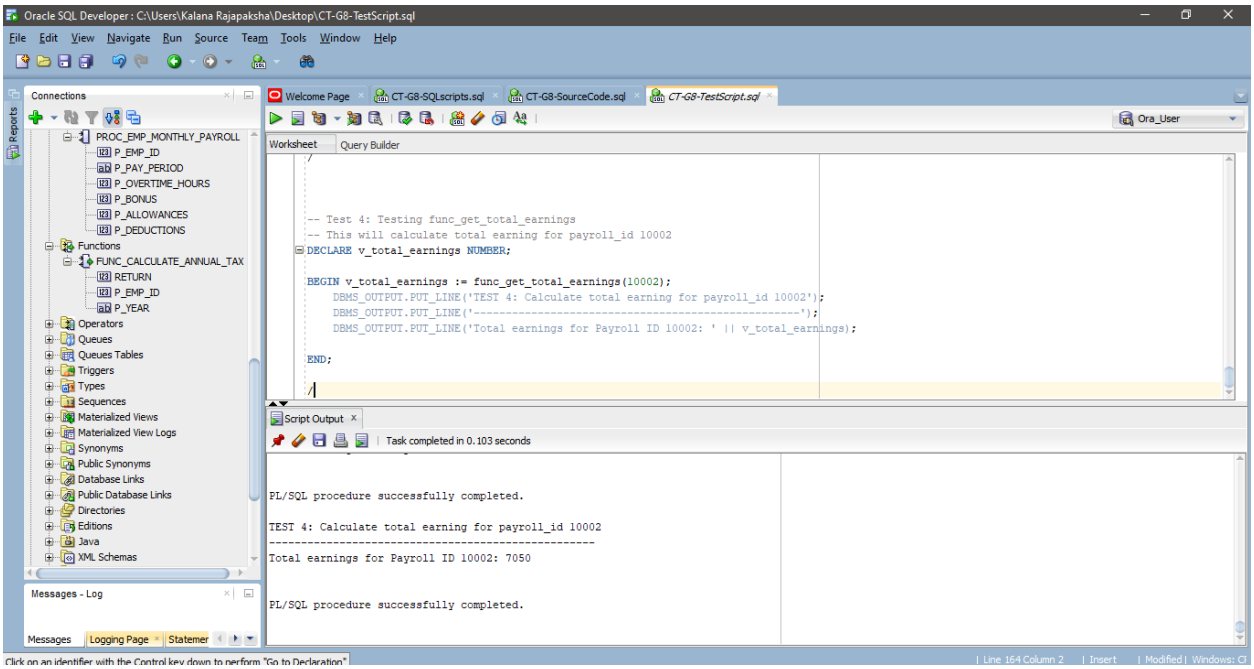

Fig. 5. Function to compute total earnings for a payroll record.

Sample Output

1) Processing Individual Payroll - proc_emp_monthly_payroll

```
TEST 1: Processing payroll for a single employee
--------------------------------------------------
ERROR: Payroll record already exists for employee ID 1001 in period 02-2023
--------------------------------------------------
Attempting to process the same payroll again (should fail):
ERROR: Payroll record already exists for employee ID 1001 in period 02-2023
--------------------------------------------------
Attempting to process payroll for non-existent employee (should fail):
ERROR: Employee with ID 9999 not found.
--------------------------------------------------
Attempting to process payroll with invalid pay period format (should fail):
ERROR: Invalid pay period format. Use MM-YYYY format (e.g., 01-2023).
--------------------------------------------------
Attempting to process payroll with negative overtime hours (should fail):
ERROR: Payroll record already exists for employee ID 1001 in period 03-2023
```

2) Processing Batch Payroll – proc_ batch_payroll

```
TEST 2: Processing batch payroll for all employees
--------------------------------------------------
Starting batch payroll processing for period: 03-2023
--------------------------------------------------
Payroll processed successfully for employee ID: 1003

Pay period: 03-2023
Basic pay: $   4,583.33
Overtime pay: $       .00
Bonus: $       .00
Allowances: $       .00
Deductions: $       .00
Tax amount: $     687.50
Net salary: $   3,895.83
Processed: Tharindu Jayasinghe (ID: 1003, Dept: Finance)
Payroll processed successfully for employee ID: 1002
Pay period: 03-2023
Basic pay: $   6,250.00
Overtime pay: $       .00
Bonus: $       .00
Allowances: $       .00
Deductions: $       .00
Tax amount: $   1,250.00
Net salary: $   5,000.00
Processed: Nadeesha Fernando (ID: 1002, Dept: HR)
```

```
--------------------------------------------------
Batch processing complete.
Total employees processed: 5
Total errors: 0
--------------------------------------------------
Attempting to process the same pay period again (some should be skipped):
Starting batch payroll processing for period: 03-2023
--------------------------------------------------
Skipped: Payroll already exists for employee ID 1003 in period 03-2023
Skipped: Payroll already exists for employee ID 1002 in period 03-2023
Skipped: Payroll already exists for employee ID 1005 in period 03-2023
Skipped: Payroll already exists for employee ID 1001 in period 03-2023
Skipped: Payroll already exists for employee ID 1004 in period 03-2023
--------------------------------------------------
Batch processing complete.
Total employees processed: 0
Total errors: 0
--------------------------------------------------
Attempting to process batch payroll with invalid pay period format (should fail):
ERROR: Invalid pay period format. Use MM-YYYY format (e.g. 01-2023).
```

3) Calculating Annual Tax - func_calculate_annual_tax

```
TEST 3: Calculating annual tax for an employee
--------------------------------------------------
Annual tax for Employee ID 1001 for year 2023:      860.75
--------------------------------------------------
Testing with non-existent employee (should show error):
ERROR: Employee with ID 9909 not found.
Function returned NULL as expected.
--------------------------------------------------
Testing with invalid year format (should show error):
ERROR: Invalid year format. Use YYYY format (e.g., 2023).
Function returned NULL as expected.
--------------------------------------------------
Testing with a year that has no payroll records (should return 0):
WARNING: No payroll records found for employee ID 1001 in year 2025
Annual tax for Employee ID 1001 for year 2022:         .00
```

4) Calculating Total Earning – func_get_total_earnings

```
TEST 4: Calculate total earning for payroll_id 10002
--------------------------------------------------
Total earnings for Payroll ID 10002: 7050
```

PL/SQL Features Used

Stored Procedures and Functions	Creation of standalone procedures and functions with parameter passing.
Cursors	Used in proc_ batch_payroll to iterate through employees
Composite Data Types	Record types used to structure data in proc_ batch_payroll
Exception Handling	Custom exceptions defined and handled for various error conditions.
Control Statements	If-then-else, loop, exit when for flow control
Built-in Functions	Use of nvl, substr, to_char, regexp_like, etc

Conclusion

The Employee Payroll System successfully implements a comprehensive solution for managing employee payroll processing. It demonstrates the use of advanced PL/SQL features including procedures, functions, cursors, and exception handling to create a robust and efficient database application. The system provides flexible options for both individual and batch payroll processing and includes detailed error handling to ensure data integrity.

----------END----------