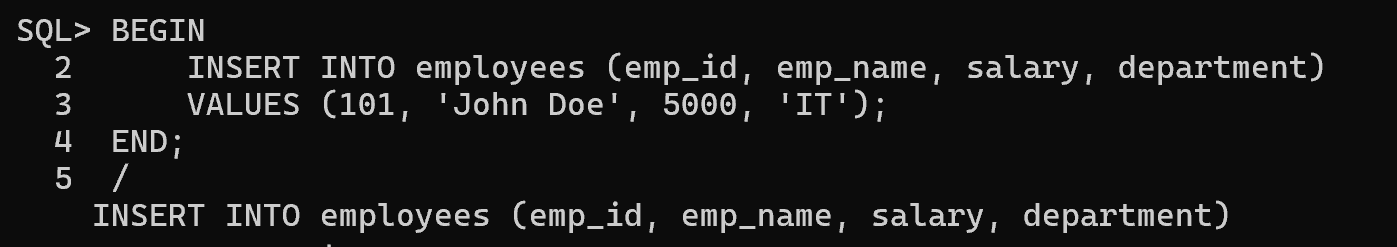
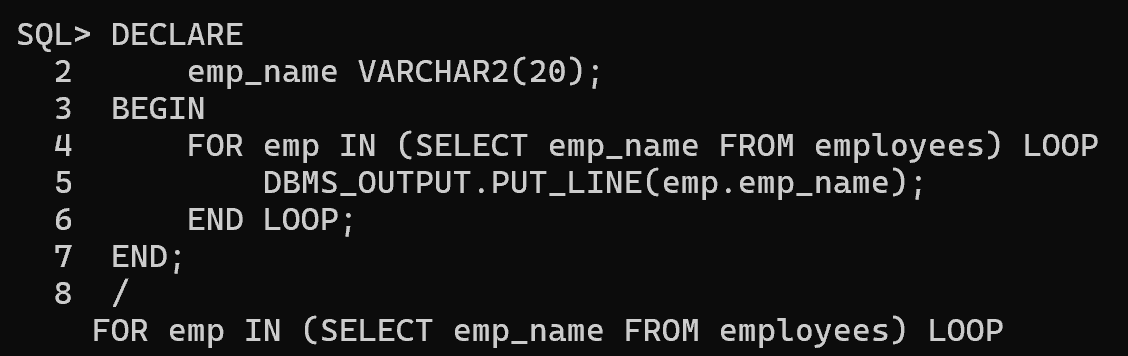
**Name:Sharvayu Zade**

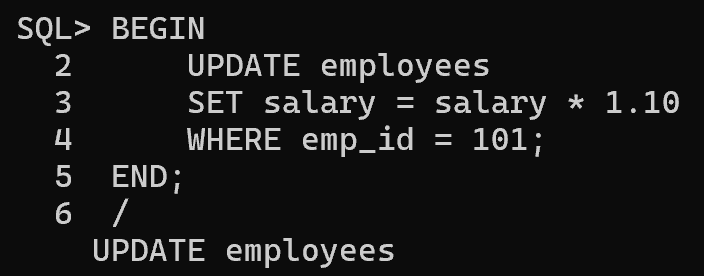
**PRN:23070521135**

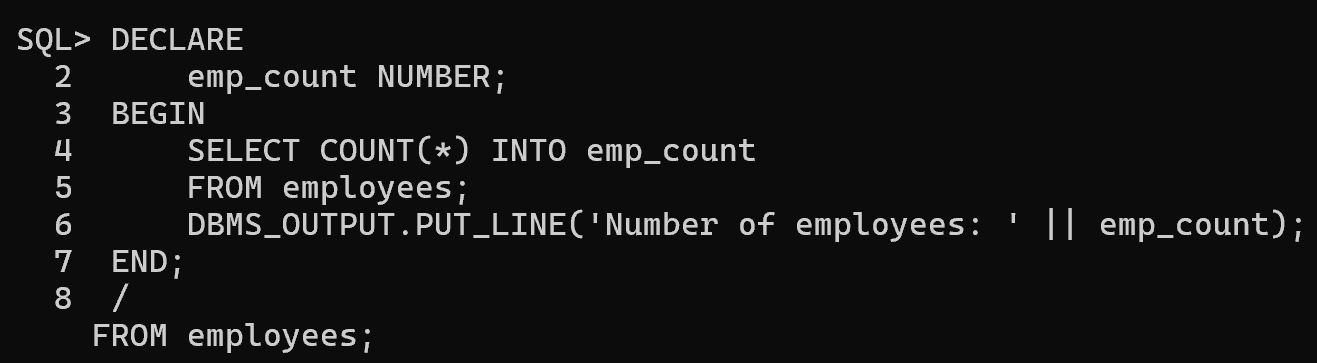
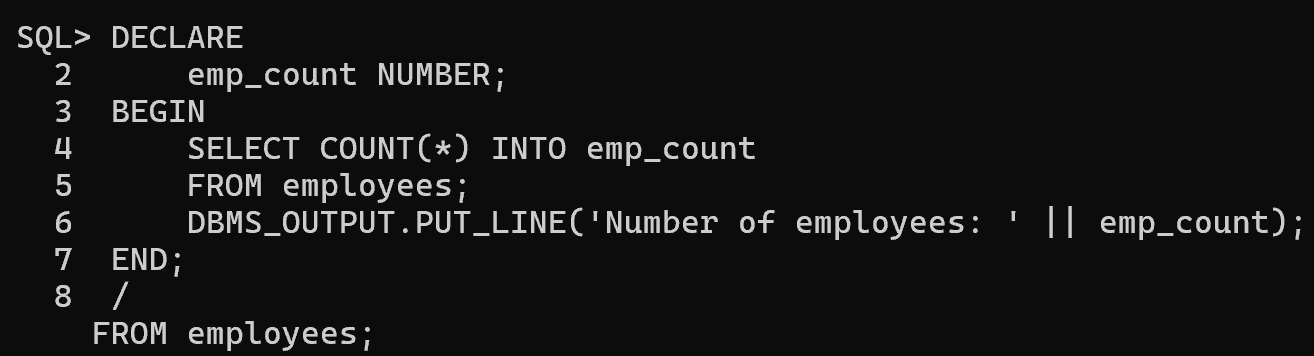
**SEC:B(B2)**

**Tasks on PL/SQL Basics with Database**

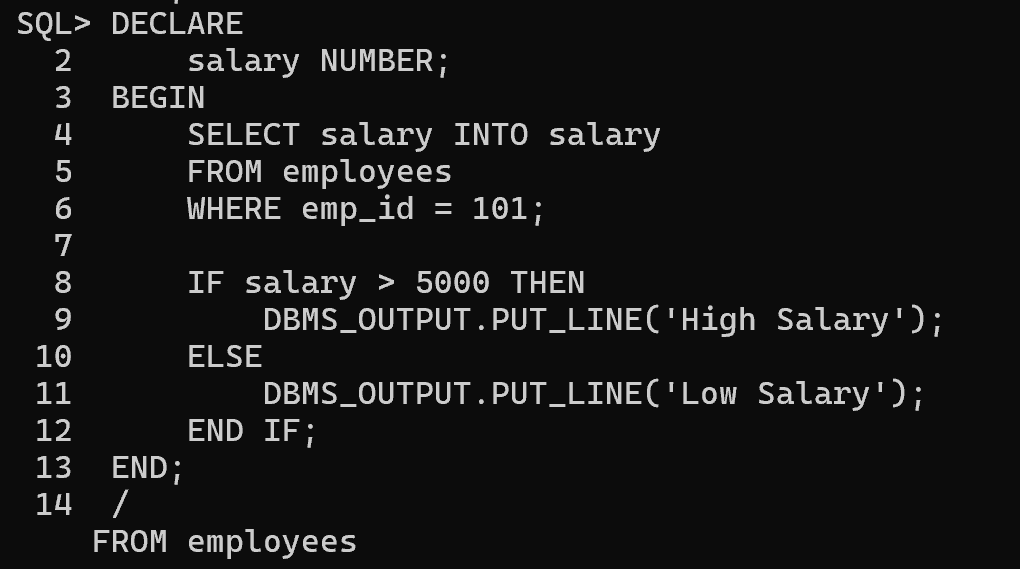
**Task 1:** Write a PL/SQL block to insert a new employee into the employees table. **Table:** employees(emp\_id, emp\_name, salary, department) Insert an employee with emp\_id = 101, emp\_name = 'John Doe', salary = 5000, department = 'IT'. 

**Task 2:** Create a PL/SQL block to retrieve and display all employee names from the employees table. 

**Task 3:** Write a PL/SQL block to update the salary of an employee whose emp\_id = 101 by increasing it by **10%**. 

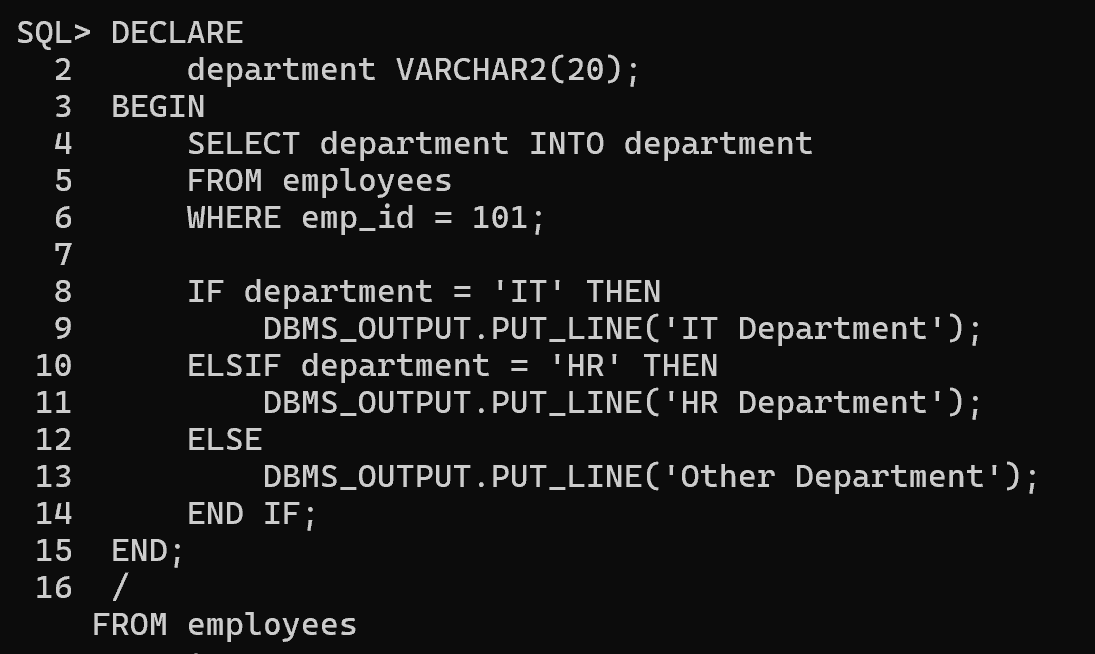
**Task 4:** Create a PL/SQL block to delete an employee whose emp\_id = 105. **Task 5:** Display the count of employees in the employees table. 

**Tasks on Conditional Statements with Database**

**Task 6:** Write a PL/SQL block that checks if an employee's salary is above **5000**. If yes, print "High Salary"; otherwise, print "Low Salary". 

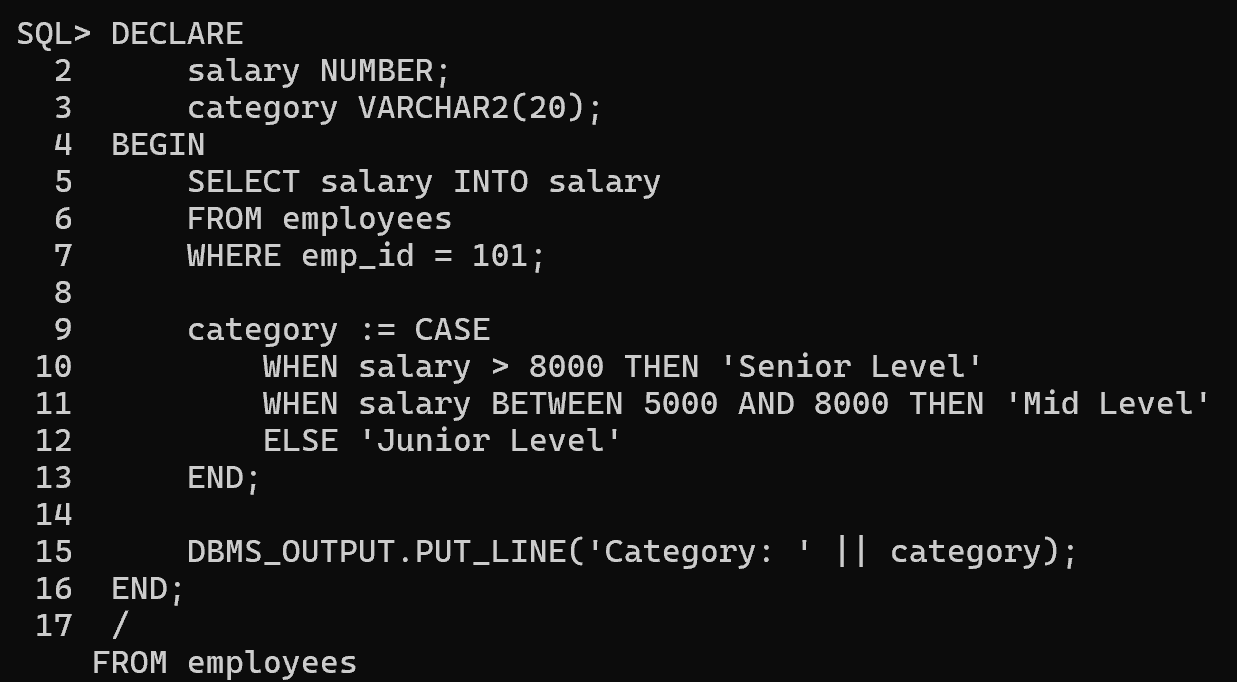
**Task 7:** Fetch the department of an employee based on emp\_id and print: ● "IT Department" if in **IT**,

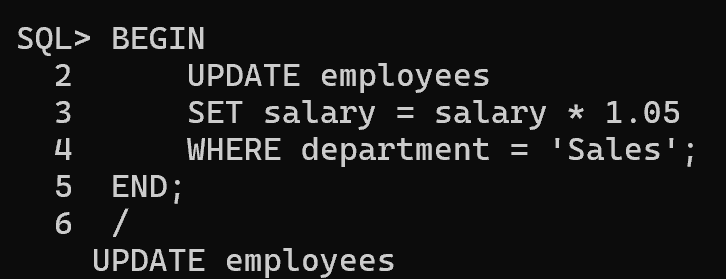
● "HR Department" if in **HR**,

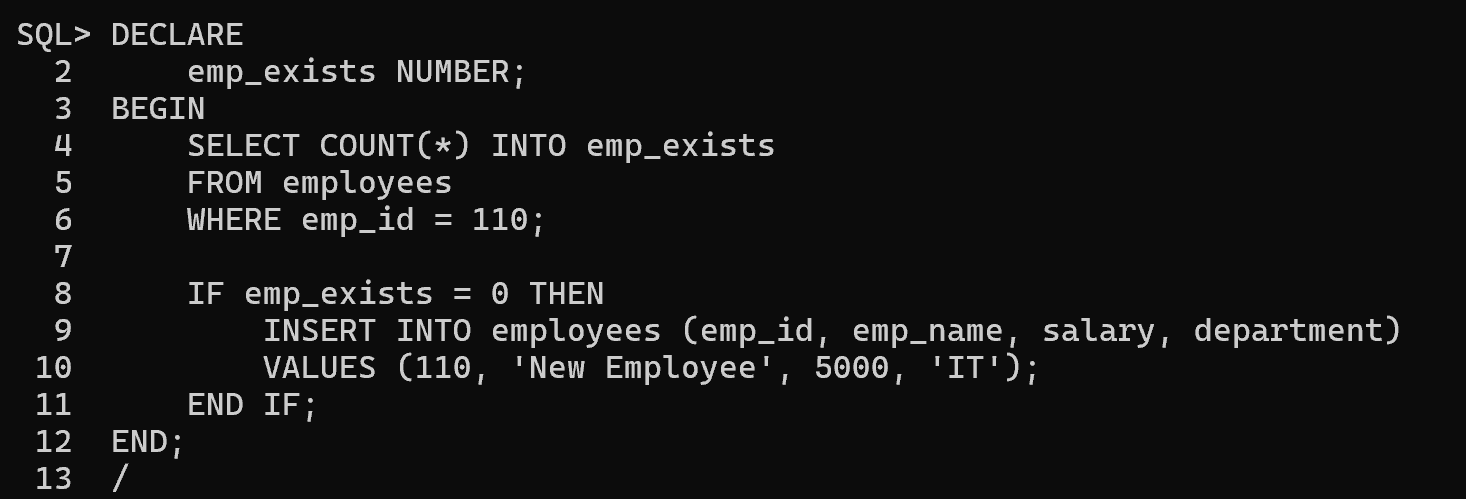
● "Other Department" otherwise. 

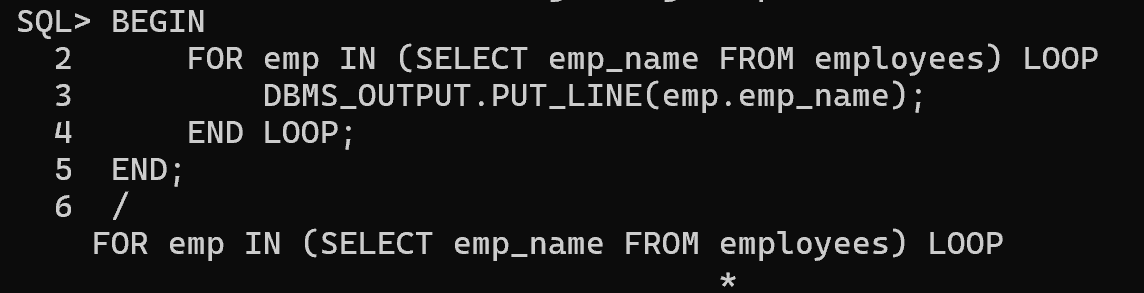
**Task 8:** Use a CASE statement to categorize employees based on salary: ● **Above 8000 → "Senior Level"**

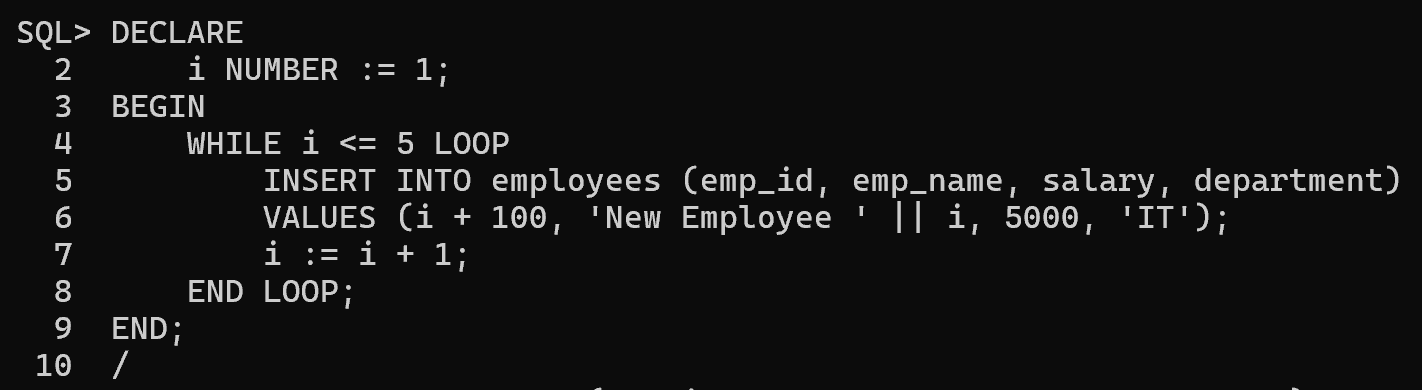
● **5000-8000 → "Mid Level"**

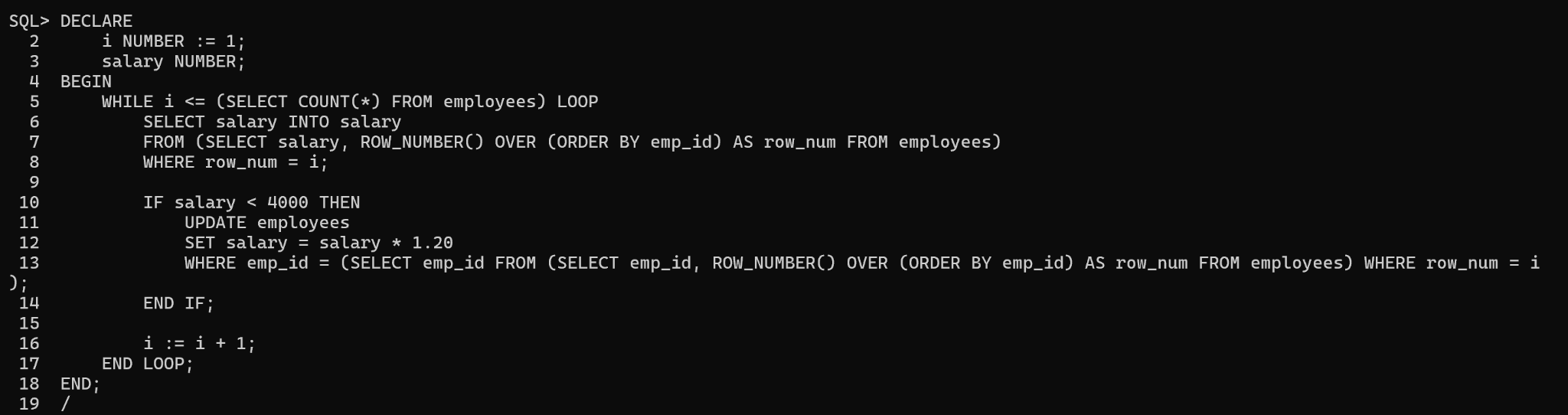
● **Below 5000 → "Junior Level"**

**Task 9:** If an employee's department is **Sales**, increase their salary by **5%**. **Task 10:** Check if an employee with emp\_id = 110 exists. If not, insert a new record. 

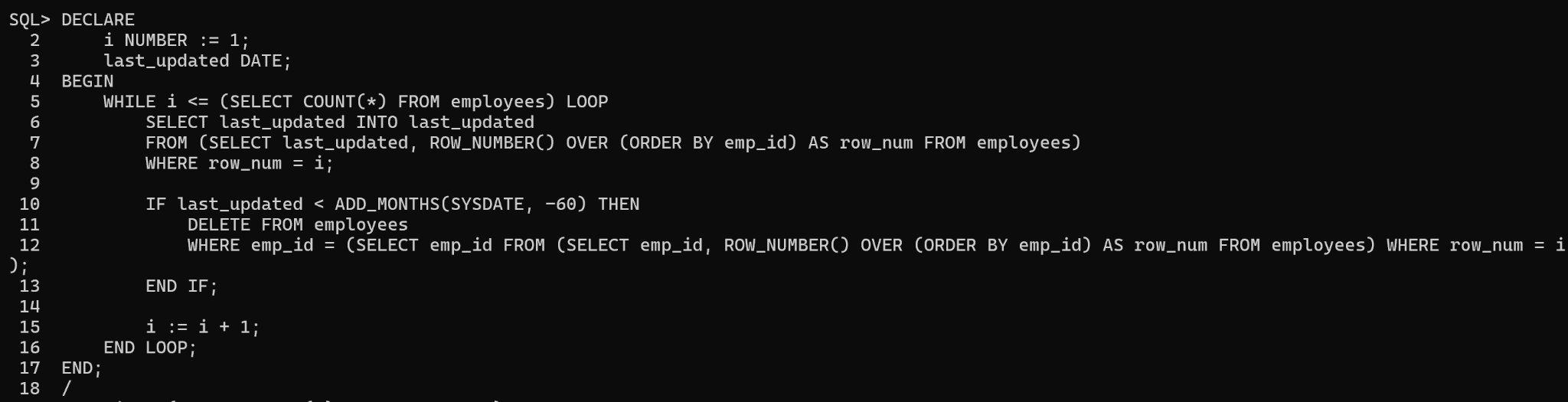
**Tasks on Loops with Database **

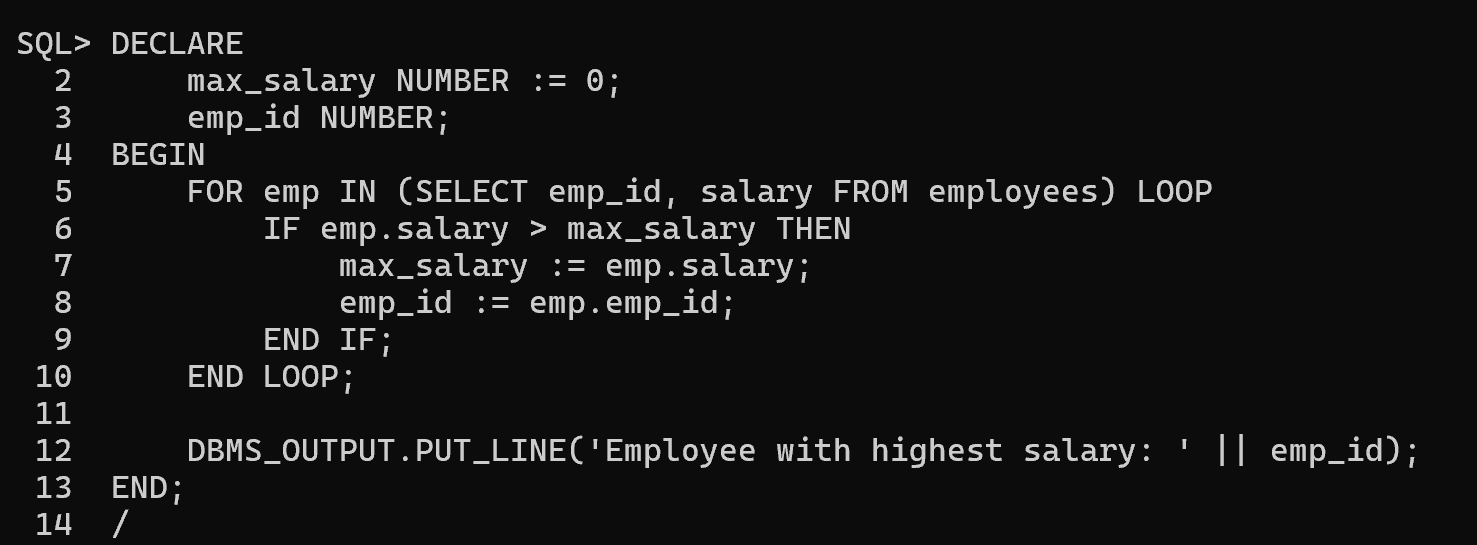
**Task 11:** Use a **FOR LOOP** to print all employees' names from the employees table.

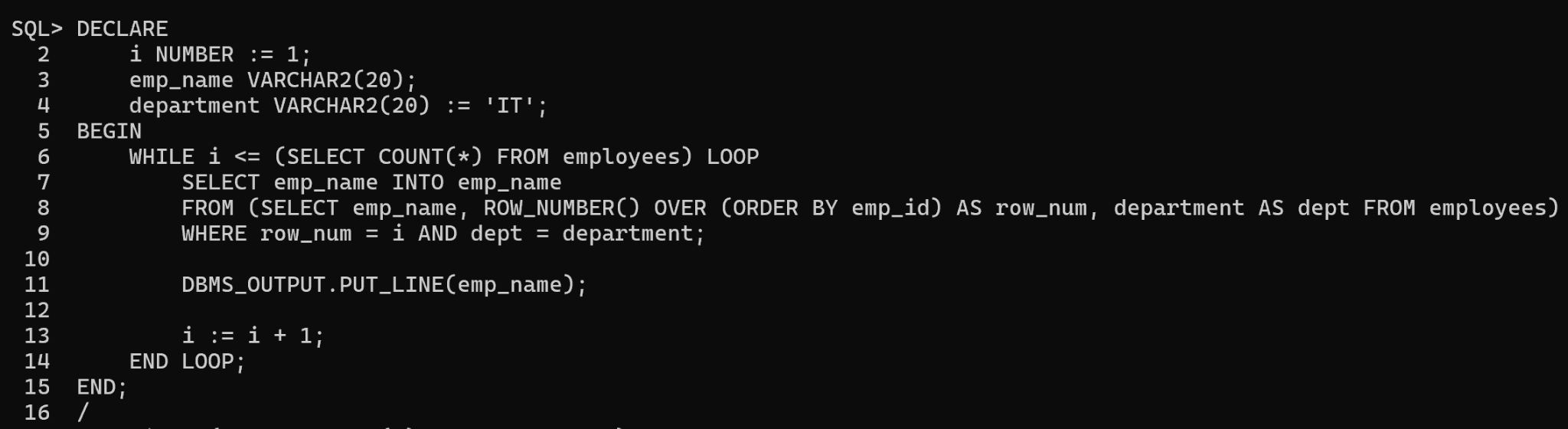
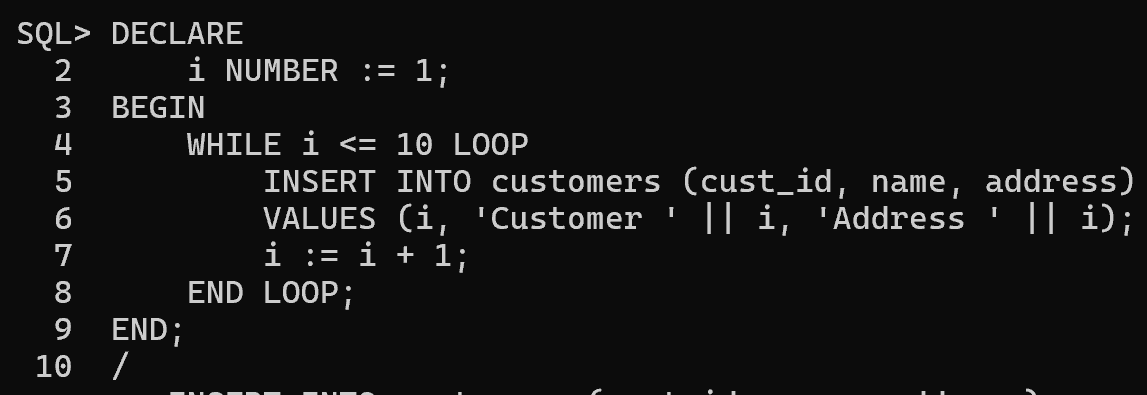
**Task 12:** Write a **LOOP** to insert **5 new employees** into the employees table. 

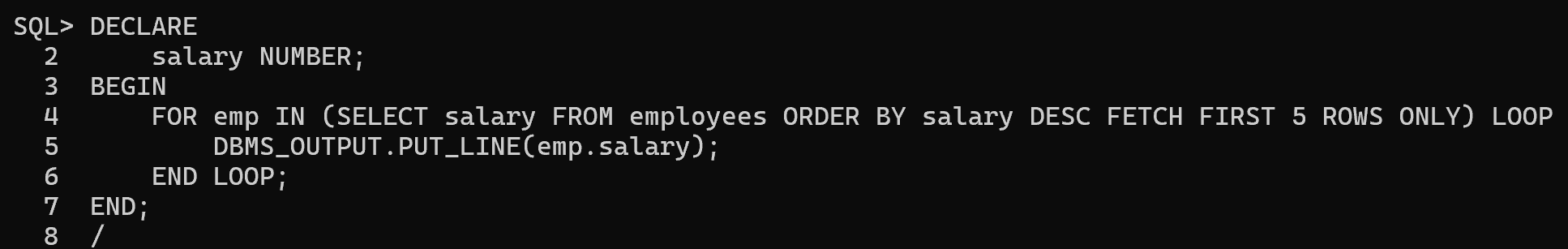
**Task 13:** Use a **WHILE LOOP** to increase the salary of all employees earning less than **4000** by **20%**. 

**Task 14:** Create a **FOR LOOP** that prints the first **3 departments** from the departments table. 

**Task 15:** Write a **LOOP** to delete employees who have not updated their records in the last **5 years** (assuming there's a last\_updated column). 

**Task 16:** Use a **LOOP** to find the employee with the highest salary in the employees table. 

**Task 17:** Fetch and display all employees in a specific department using a **WHILE LOOP**. **Task 18:** Write a **LOOP** to insert **10 new customers** into a customers table. 

**Task 19:** Use a **FOR LOOP** to display the top **5 highest-paid employees** from the employees table. 

**Task 20:** Write a **LOOP** to find and delete duplicate employee records in the employees table.