Introduction to PL/SQL

OBJECTIVES

- After completing this lesson, you should be able to do the following:
  - Explain the need for PL/SQL
  - Explain the benefits of PL/SQL
  - Identify the different types of PL/SQL blocks
  - Use *i*SQL*Plus as a development environment for PL/SQL
  - Output messages in PL/SQL

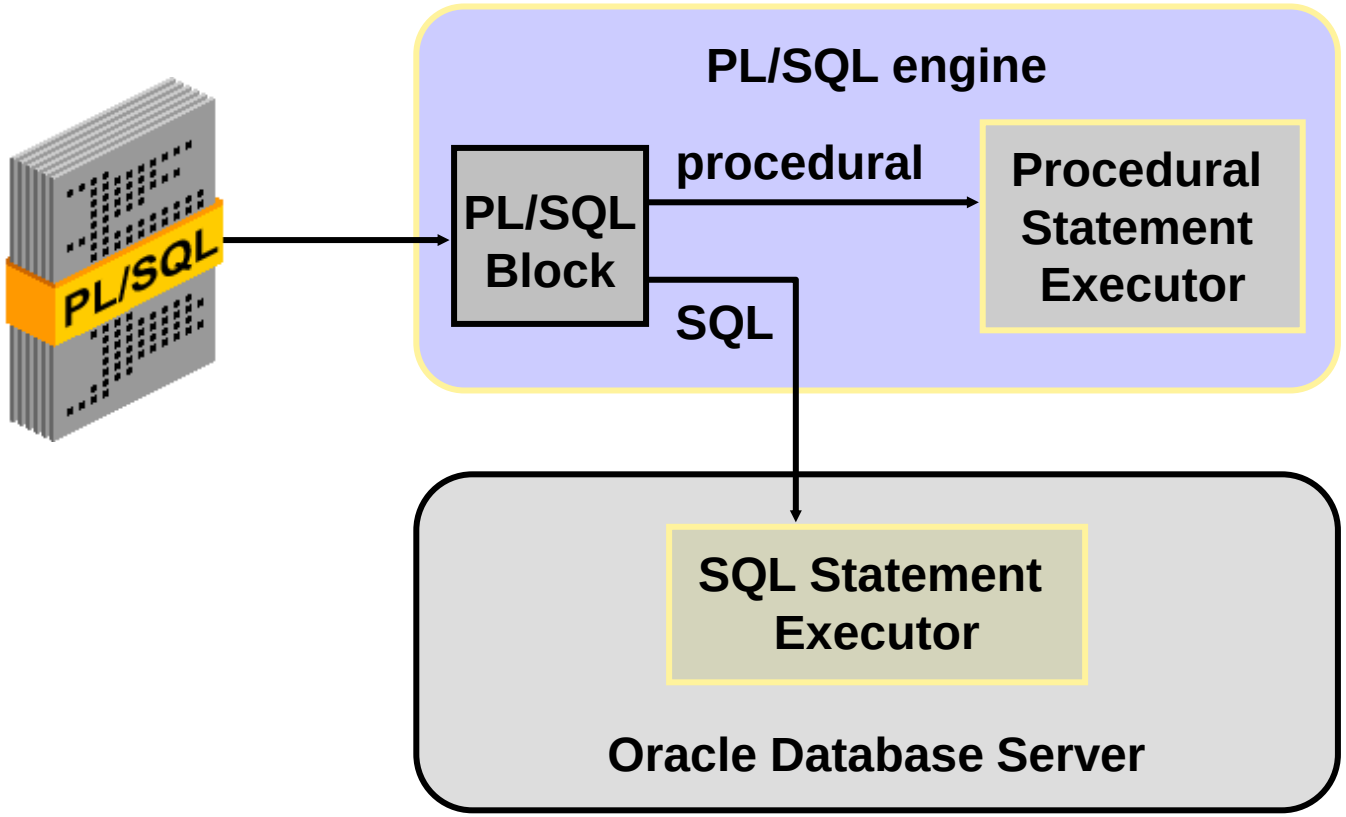
What Is PL/SQL?

- PL/SQL:
  - Stands for Procedural Language extension to SQL
  - Is Oracle Corporation's standard data access language for relational databases
  - Seamlessly integrates procedural constructs with SQL

About PL/SQL

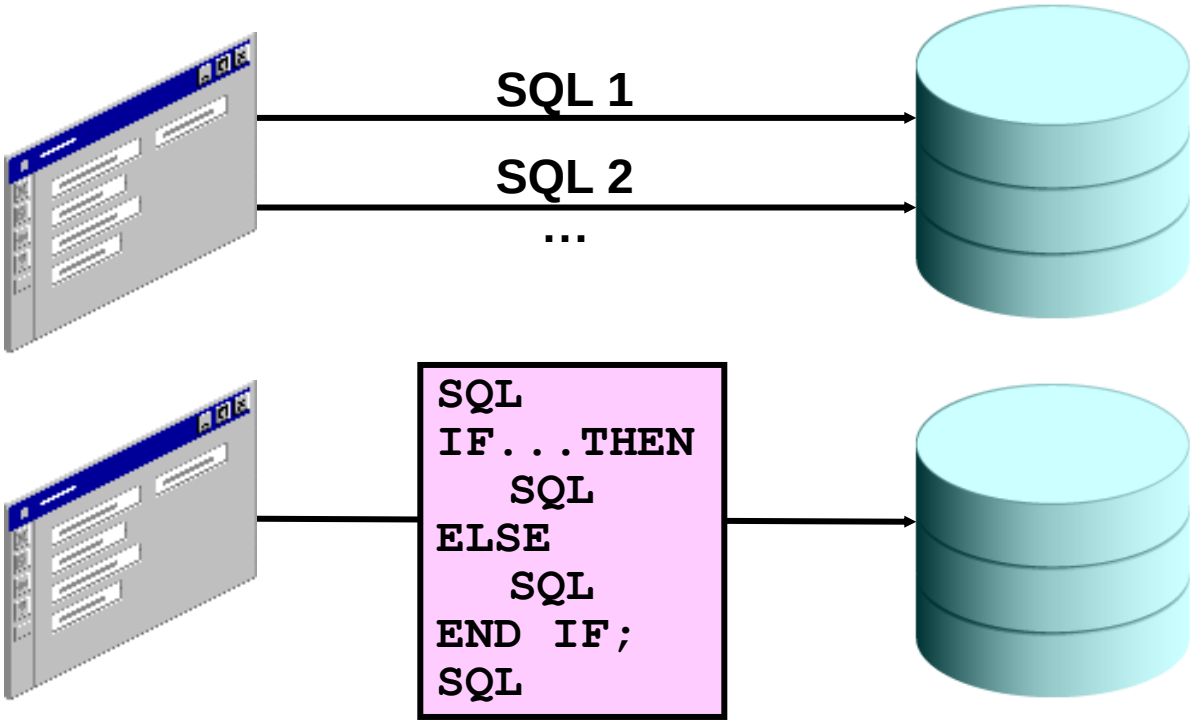
- PL/SQL:
  - Provides a block structure for executable units of code. Maintenance of code is made easier with such a well-defined structure.
  - Provides procedural constructs such as:
    - Variables, constants, and types
    - Control structures such as conditional statements and loops
    - Reusable program units that are written once and executed many times

PL/SQL Environment



Benefits of PL/SQL

- Integration of procedural constructs with SQL
- Improved performance

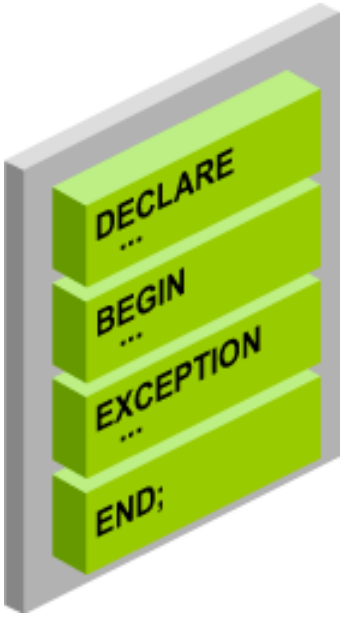


Benefits of PL/SQL

- Modularized program development
- Integration with Oracle tools
- Portability
- Exception handling

PL/SQL Block Structure

- `DECLARE` (optional)
  - Variables, cursors, user-defined exceptions
- `BEGIN` (mandatory)
  - SQL statements
  - PL/SQL statements
- `EXCEPTION` (optional)
  - Actions to perform when errors occur
- `END;` (mandatory)



Block Types

- Anonymous

```
[DECLARE]


BEGIN
   --statements

[EXCEPTION]

END;
```

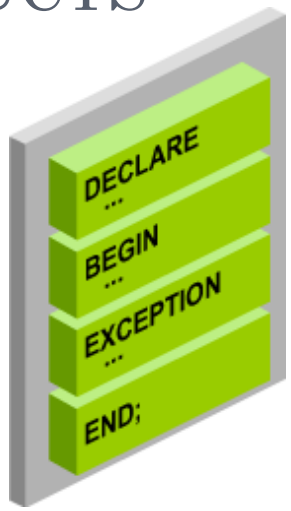
Function

```
PROCEDURE name
IS

BEGIN
   --statements

[EXCEPTION]

END;
```

Procedure

```
FUNCTION name
RETURN datatype
IS
BEGIN
   --statements
   RETURN value;
[EXCEPTION]

END;
```

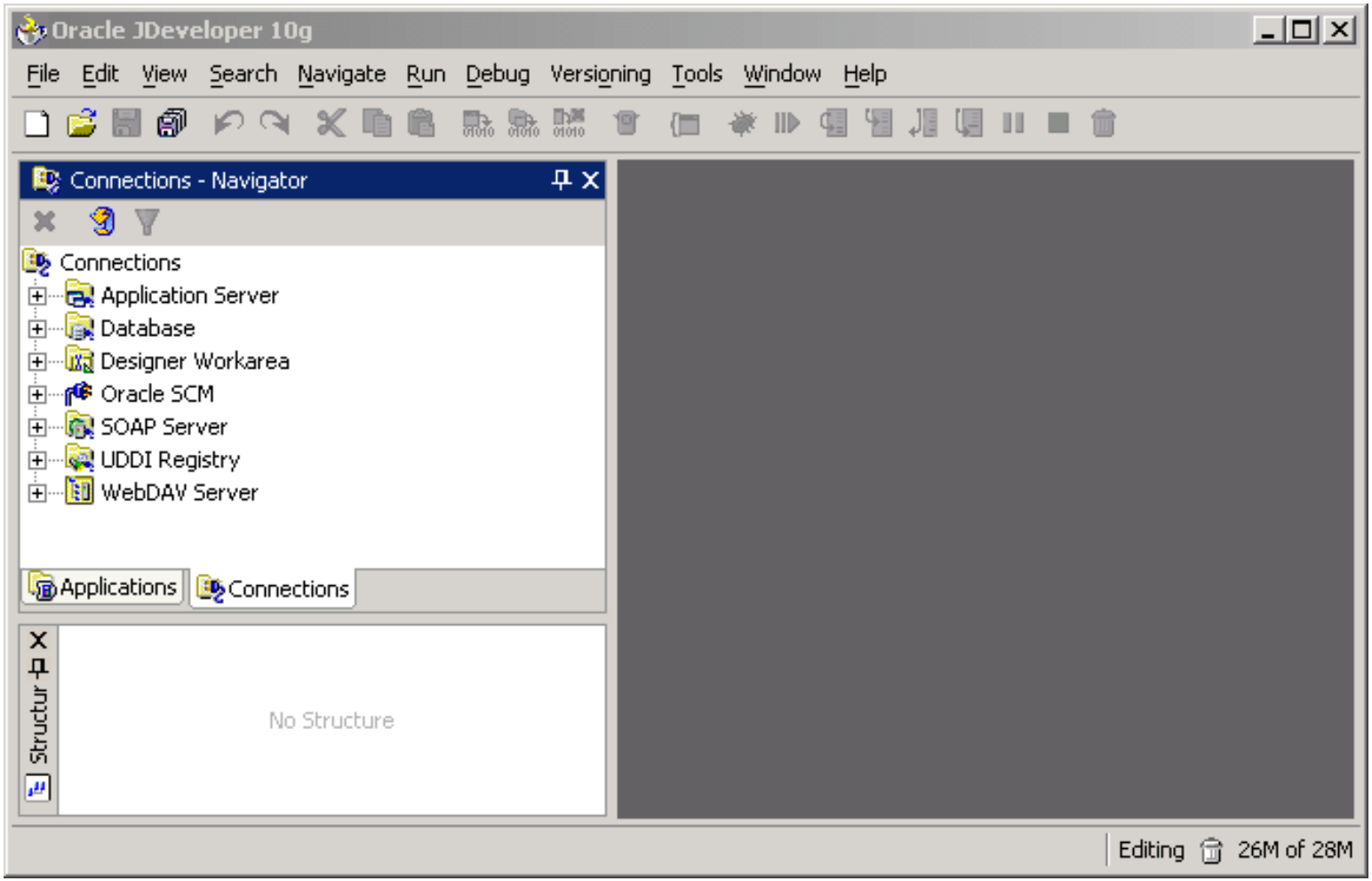
Program Constructs



Tools Constructs
Anonymous blocks
Application procedures or functions
Application packages
Application triggers
Object types

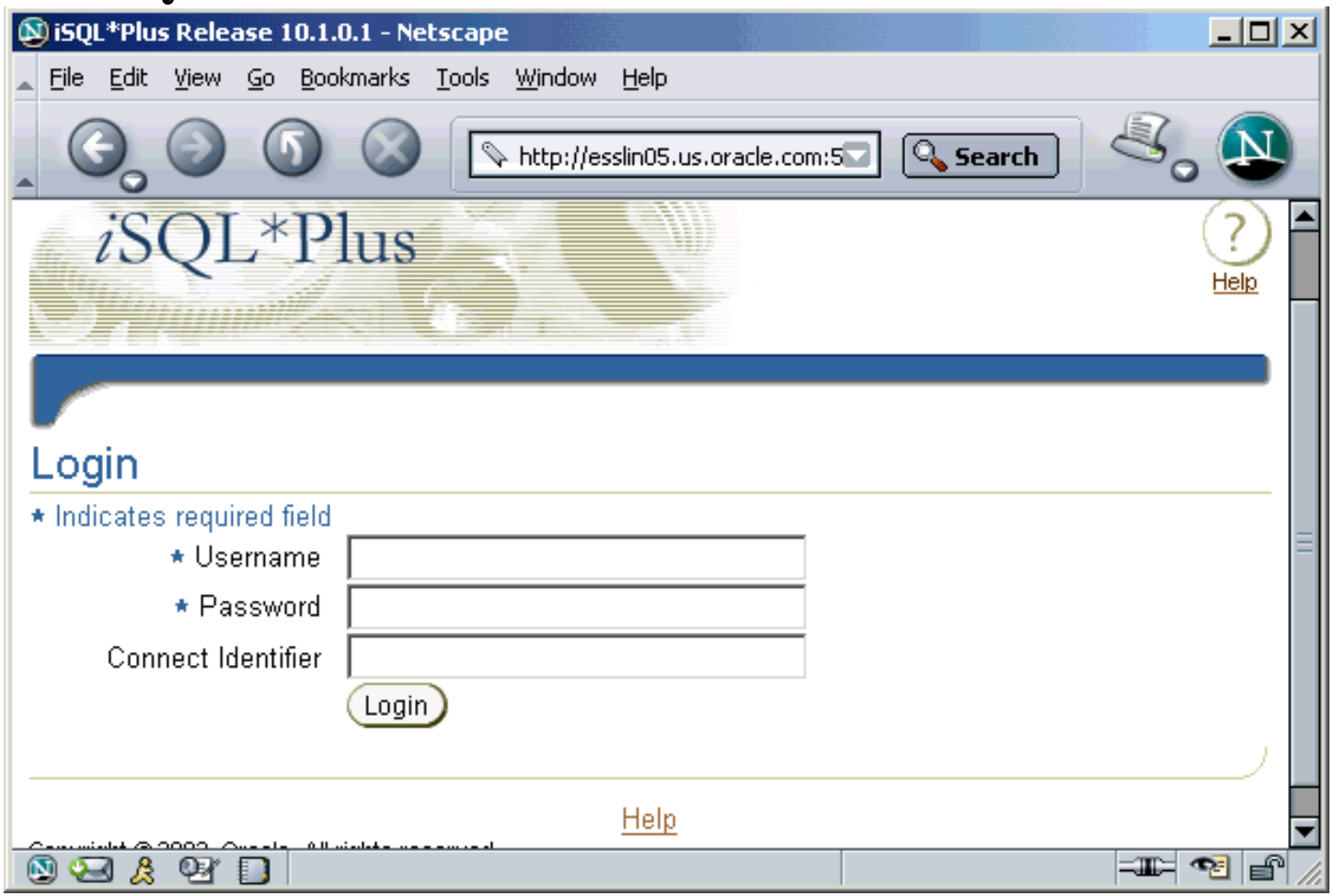
Database Server Constructs
Anonymous blocks
Stored procedures or functions
Stored packages
Database triggers
Object types

PL/SQL Programming Environments

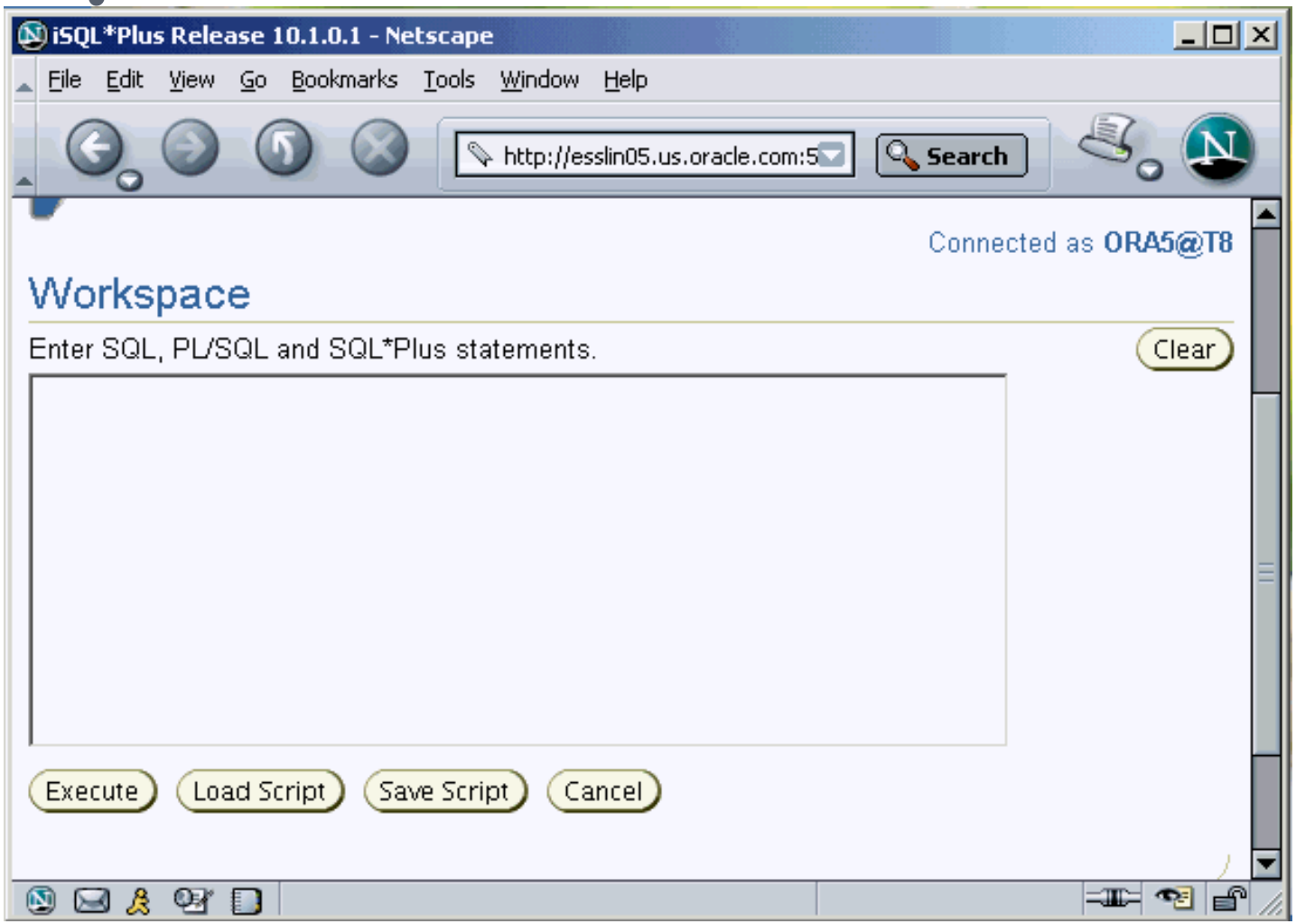


PL/SQL Programming Environments

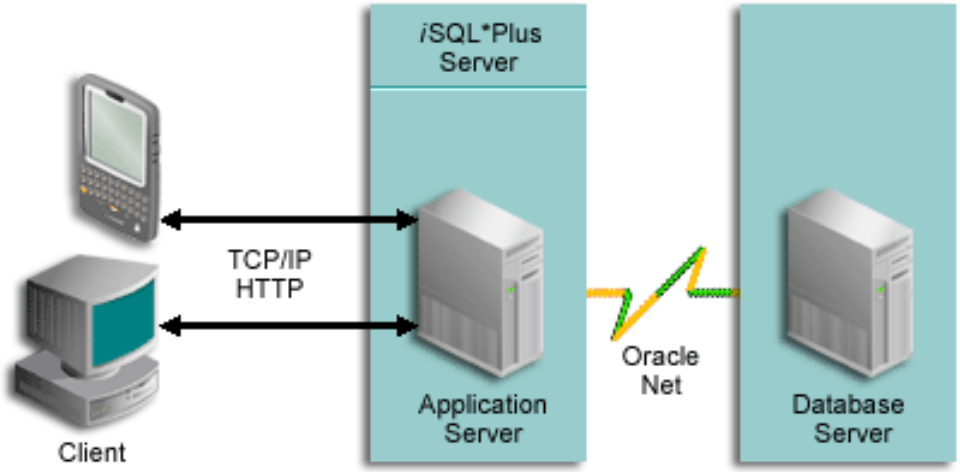
- *i*SQL*Plus



PL/SQL Programming Environments

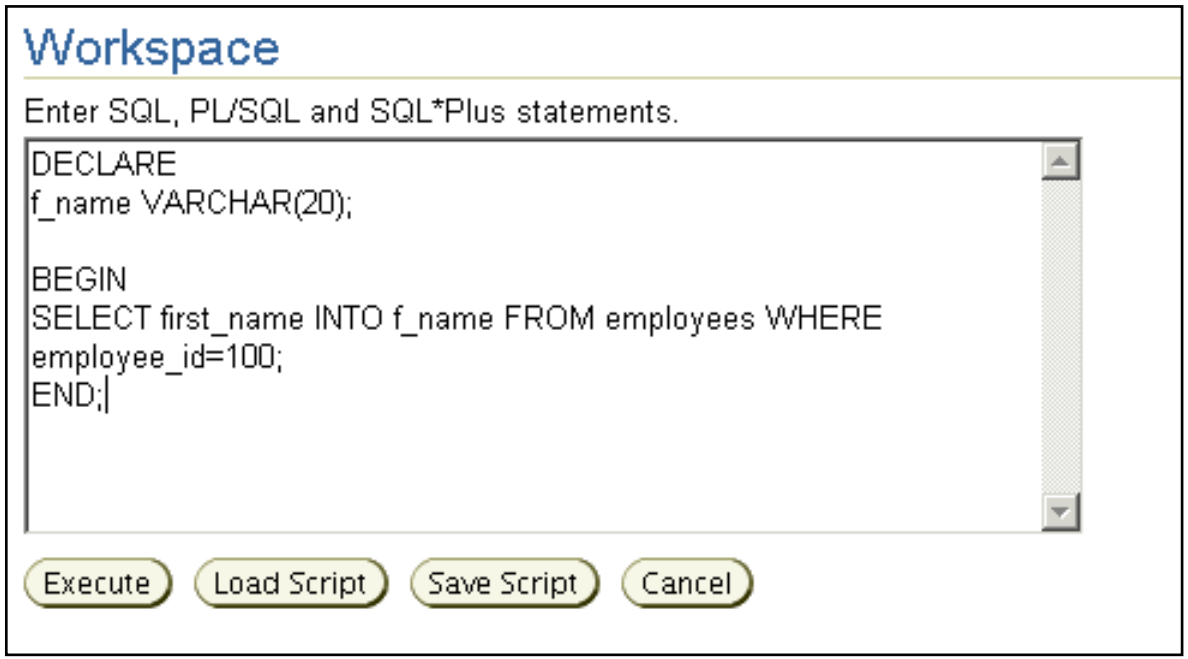


*i*SQL*Plus Architecture



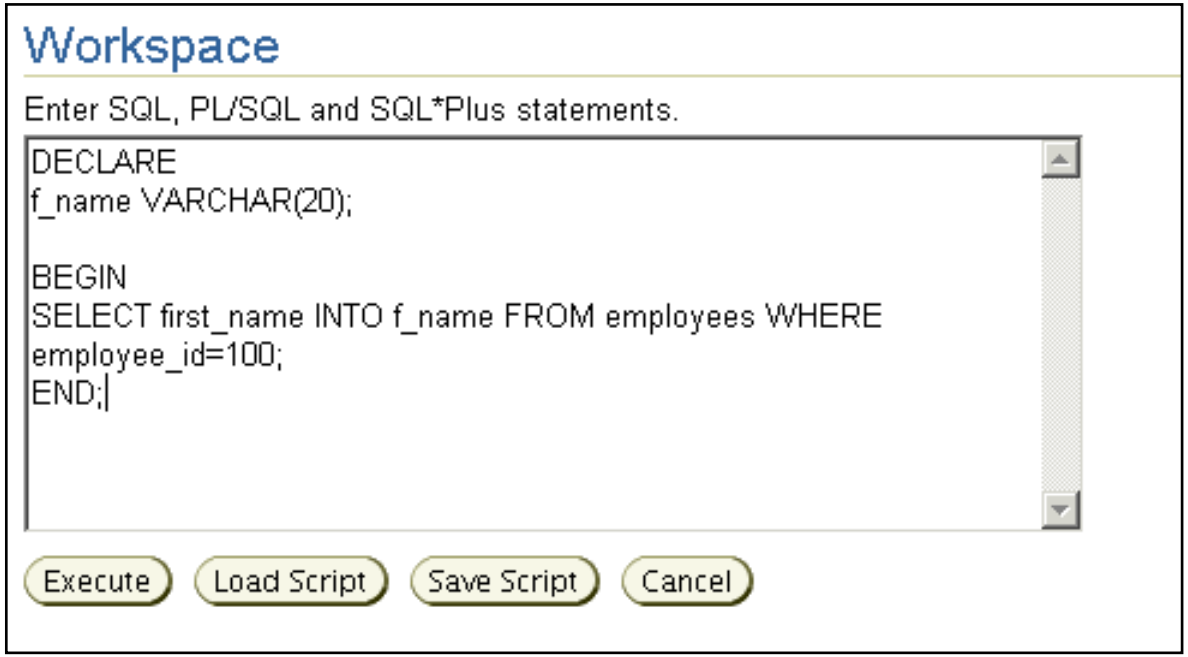
Create an Anonymous Block

- Type the anonymous block in the *i*SQL*Plus workspace:



Execute an Anonymous Block

- Click the Execute button to execute the anonymous block:



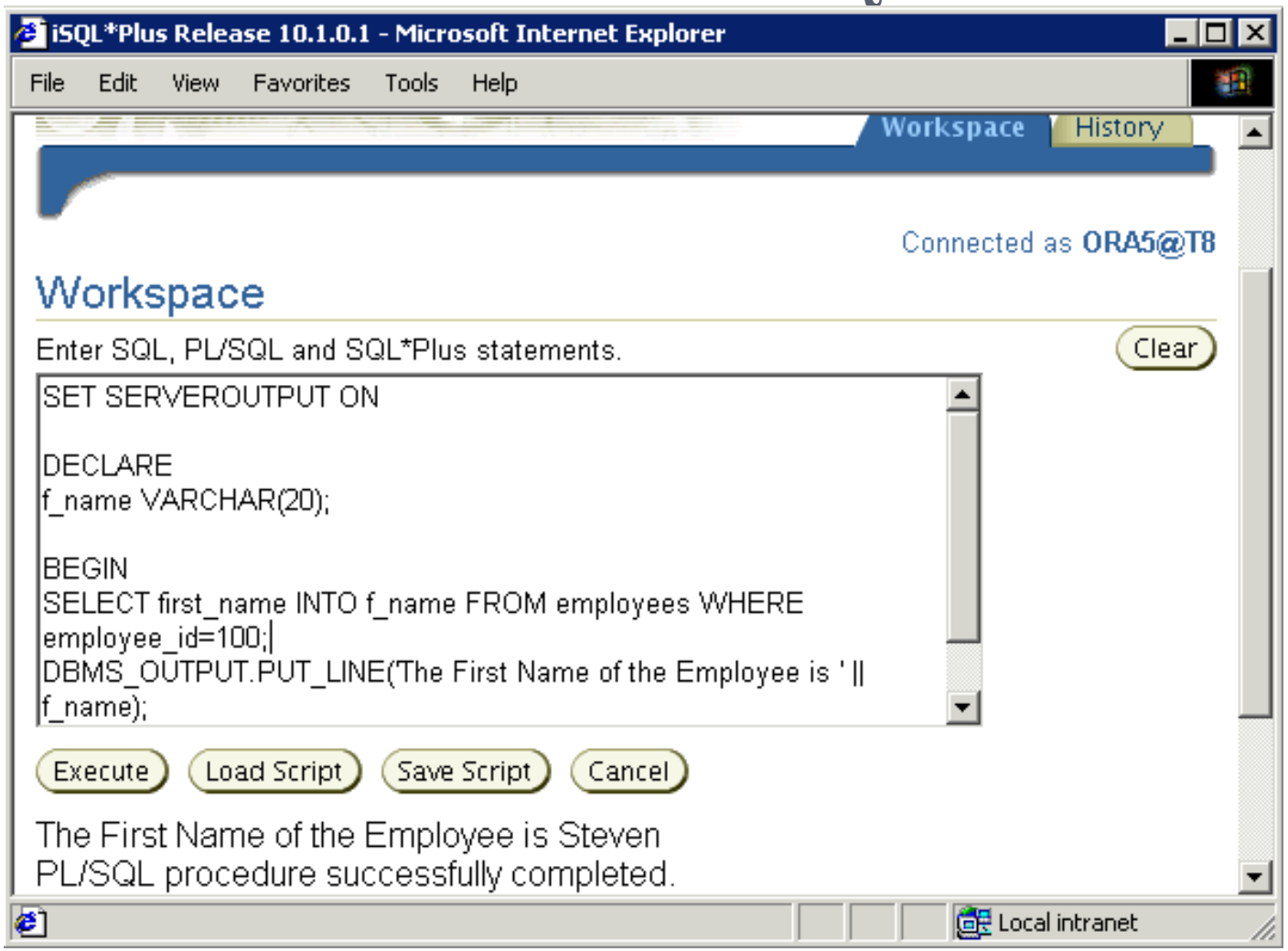
PL\SQL procedure successfully completed.

Test the Output of a PL/SQL Block

- Enable output in *i*SQL*Plus with the following command:
  `SET SERVEROUTPUT ON`
- Use a predefined Oracle package and its procedure:
  - `DBMS_OUTPUT.PUT_LINE`

```
SET SERVEROUTPUT ON
…
DBMS_OUTPUT.PUT_LINE(' The First Name of the
Employee is ' || f_name);
…
```

TEST THE OUTPUT OF A PL/SQL BLOCK



SUMMARY

- In this lesson, you should have learned how to:
  - Integrate SQL statements with PL/SQL program constructs
  - Identify the benefits of PL/SQL
  - Differentiate different PL/SQL block types
  - Use *i*SQL*Plus as the programming environment for PL/SQL
  - Output messages in PL/SQL

Practice 1: Overview

- This practice covers the following topics:
  - Identifying which PL/SQL blocks execute successfully
  - Creating and executing a simple PL/SQL block