|  |  |
| --- | --- |
|  |  |
| Centralized College Database |  |
|  |  |
|  |  |
|  | Saikrishna B A |

PL/SQL (Procedural Language extensions to the Structured Query Language)

PL/SQL was developed by Oracle Corporation in the early ’90s to enhance the capabilities of SQL. It is a database-oriented programming language that is a powerful extension of SQL with procedural capabilities. The key strength of PL/SQL is its tight integration with the Oracle database.

Some of its procedural capabilities include

♣ Variable Definition and Assignment

♣ Conditional Processing

♣ Loop Constructs

♣ Error Handling

♣ Unique Capabilities of Subprograms

So in this report of ours which is concerning the Centralised College Database, we have endeavored to use certain features of Advance SQL such as procedures, Case, If-Else statement, While loops, Cursor, and Trigger.

ER DIAGRAM

Diagram

Description automatically generated

WHAT ARE PL/SQL PROCEDURES?

The PL/SQL stored procedure or simply a procedure is a PL/SQL block that performs one or more specific tasks. It is just like procedures in other programming languages.

The procedure contains a header and a body.

* **Header:** The header contains the name of the procedure and the parameters or variables passed to the procedure.
* **Body:** The body contains a declaration section, execution section, and exception section similar to a general PL/SQL block.

## **How to pass parameters in procedure:**

When you want to create a procedure or function, you have to define parameters. There are three ways to pass parameters in procedure:

1. **IN parameters:**The IN parameter can be referenced by the procedure or function. The value of the parameter cannot be overwritten by the procedure or the function.
2. **OUT parameters:**The OUT parameter cannot be referenced by the procedure or function, but the value of the parameter can be overwritten by the procedure or function.
3. **INOUT parameters:**The INOUT parameter can be referenced by the procedure or function and the value of the parameter can be overwritten by the procedure or function.

Graphical user interface, text, application, email

Description automatically generated

CASE STATEMENT

1. The CASE statement is a compact way to evaluate a single condition and choose between many alternative actions.

2. This statement is very useful when we have multiple conditions as well as the expected input is also multiple and changeable according to the situation.

3. It makes sense to use CASE when there are three or more alternatives to choose from.

Graphical user interface, text, application, email

Description automatically generated

STORED PROCEDURES

1. A stored procedure or simple subroutine or a proc or a subprogram is a named PL/SQL block that performs one or more specific tasks.
2. The stored procedures are written in advance and compiled before their use. This improves the speed of execution.
3. This is like a procedure or function in other programming languages.
4. A procedure has a header and a body.
5. The header consists of the name of the procedure and the parameters or variables passed to the procedure.
6. The body consists of a declaration section, execution section, and exception section like a general PL/SQL Block.
7. A procedure is like an anonymous PL/SQL Block, but it is named for repeated usage.
8. An exception is an error that disrupts the normal flow of program instructions.
9. PL/SQL provides us the exception block which raises the exception thus helping the programmer to find out the fault and resolve it.



CURSORS

1. When an SQL statement is processed, Oracle creates a memory area known as the context area.
2. A cursor is a pointer to this context area. It contains all information needed for processing the statement. In PL/SQL, the context area is controlled by Cursor.
3. A cursor contains information on a select statement and the rows of data accessed by it.

A cursor is used to refer to a program to fetch and process the rows returned by the SQL statement, one at a time. There are two types of cursors:

* Implicit Cursors
* Explicit Cursors

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

TRIGGERS

* The triggers play an important role while validating the SQL and PLSQL queries on an automatic basis depending on some conditions.
* It executes or fired like a definite event every time.
* A trigger can be defined as automatic code execution on an event of a database.
* A trigger is a PL/SQL block structure or a subprogram that is fired or executed when a DML statement like Insert, Delete, Update is executed on a database table.
* A trigger is triggered automatically at predefined timing and even when an associated DML statement is executed.
* Triggers are physically stored in a database.

Text, timeline

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

WHILE-LOOP

* The WHILE loop statement continues to execute the statements between the LOOP and END LOOP as long as the condition in the WHILE clause evaluates to TRUE.
* PL/SQL evaluates the condition in the WHILE clause before each loop iteration. If the condition is TRUE, then the loop body executes. In case it is FALSE or NULL, the loop terminates.
* If the condition is FALSE before entering the loop, the WHILE loop does not execute at all. This behaviour is different from the LOOP statement whose loop body always executes once.

Graphical user interface, text, application, email

Description automatically generated

IF-ELSE CONDITION

* In PL/SQL, the IF-THEN-ELSE statement is used to execute code when a condition is TRUE or execute a different code if the condition evaluates to FALSE.

Graphical user interface, text, application, email

Description automatically generated

APPLICATIONS OF CENTRALIZED COLLEGE DATABASE

A centralized database is stored at a single location such as a mainframe computer. It is maintained and modified from that location only and usually accessed using an internet connection such as a LAN or WAN. The centralized database is used by organizations such as colleges, companies, banks, etc.

The potential benefits of using the Centralized College Database are:

* One-stop Solution

Manage all activities and tasks of campus from a single CCDB software platform.

* Better & Faster decisions

Provides precise & accurate information to end-users for a better decision.

* Reduced Workload

Automated tasks enable teachers & faculty to focus more on teaching.

* Centralized Database

Organize & store all key education institution data in the unified database of CCDB.

* Streamlined Processes

The deployed workflow-based system ensures streamlined processes.

* Reduced Cost

Automated tasks reduce man-hours which reduces the cost of operations.

* No Queues

The students can apply online for admission & pay fees online 24\*7 instead of 3 hours in a queue.

* Paperless Operations

Since all operations are carried out through centralized campus management software, the need for paper documentation is eliminated.

REFLECTIONS

While working on this project of developing a simplified model of a college database, I started to ponder over the actual database systems used in colleges and at a university level. This helped me appreciate the beauty of PLSQL as it helps in swift data retrieval and making quick changes. The utility of Triggers struck a chord with me; this helps in conditional matching, for instance, we can use the After-Delete Trigger to view the record of changes that have been made in the database.

This project gave me great insights into the nitty-gritty of the College Database. The use of control statements enthralled me, as these are conditional statements that help in retrieving specific portions of data from the database. For instance, suppose an admin member wants to view the number of students who have failed, in such a scenario control statement comes handy as conditions can easily be passed to see the number of students who have passed or failed.

I have always been curious to know about the backend work of a database. Working on this project gave me a very good idea of how a college database works. Though our project is a simplified model of a centralized database, the learnings are invaluable. This project gave me hands-on experience in the usage of CASE. Case statements are useful when we have multiple conditions and each condition is followed by certain actions or statements. For instance, a Case can be used for grading as multiple conditions can be passed on the marks obtained and grades can be passed as actions or statements corresponding to the conditions.

THANK YOU