

2. Übung "PL/SQL: Kontrollstrukturen"

1. Schleifen

- Erzeugen Sie eine Tabelle MESSAGES, die nur aus einer Spalte RESULTS (VARCHAR2(2)) besteht.
- Schreiben Sie einen PL/SQL-Block (Schleife) zum Einfügen der Zahlen 1 bis 10 (ausgenommen 6 und 8). Setzen Sie vor dem Ende des Blockes ein COMMIT ab.
- Überprüfen Sie die MESSAGES-Tabelle nach Ausführung des Blockes, ob dieser erfolgreich gearbeitet hat.

2. Alternativen

Schreiben Sie einen PL/SQL-Block, der die Kommission für einen gegebenen Angestellten der Tabelle EMP, basierend auf dessen Gehalt, berechnet. Die Änderung ist mit COMMIT in der Datenbank zu bestätigen.

- Erzeugen Sie zuerst eine Tabelle EMP_KOPIE als Kopie der Tabelle EMP.
- Fügen Sie in diese Tabelle einen neuen Angestellten mit folgenden Daten ein:
EMPNO=8000, ENAME='DOE', JOB='CLERK', MGR=7698, DEPTNO=10

- Das Programm soll wie folgt (auf der Tabelle EMP_KOPIE) arbeiten:

Die Angestellten-Nr. ist über eine SQL*Plus-Substitutionsvariable einzugeben.

- Wenn das Gehalt des Angestellten weniger als 1.000\$ beträgt, ist der Betrag der Kommission auf 10% des Gehalts festzulegen.
- Wenn das Gehalt des Angestellten zwischen 1.000\$ und 1.500\$ beträgt, so ist der Betrag der Kommission auf 15% des Gehalts festzusetzen.
- Wenn das Gehalt des Angestellten 1.500\$ übersteigt, so ist der Betrag der Kommission auf 20% des Gehalts festzulegen.
- Wenn das Gehalt des Angestellten NULL ist, so ist die Kommission auf 0 festzulegen.

Prüfen Sie, ob Ihr Programm funktioniert, indem Sie folgende Testfälle verwenden, um den Wert der Kommission zu überprüfen:

EMPNO	SALARY	COMMISSION
7369	800	80
7934	1300	195
7499	1600	320
8000	NULL	0

3. Arbeit mit Cursors

Schreiben Sie einen PL/SQL-Block, der die Top-Angestellten der Tabelle EMP mit Bezug auf ihr Gehalt ermittelt.

- Für die Speicherung der Top-Leute erzeugen Sie eine neue Tabelle TOP_D0GS mit zwei Spalten:
NAME(VARCHAR2(25)), SALARY(NUMBER(11,2)).
- Es soll zur Laufzeit eine Anzahl n eingegeben werden, für die ein SQL*Plus-Substitutionsparameter genutzt werden kann.

- c. Ermitteln Sie in einer Schleife Nachname und Gehalt der n Top-Angestellten mit Bezug auf ihr Gehalt in der EMP-Tabelle.
- d. Speichern Sie die Namen und Gehälter in der neuen Tabelle TOP_DOGS.
- e. Es gelte die Annahme, dass es keine zwei Angestellten mit dem gleichen Gehalt gibt.
- f. **Testen Sie alle möglichen Fälle, so auch $n=0$, $n > \text{Anzahl der Angestellten in der Tabelle}$.**
Leeren Sie die TOP_DOGS-Tabelle nach jedem Test.
- g. Schreiben Sie das Programm so um, dass es berücksichtigt, dass es **nach** dem n-ten Angestellten mögliche weitere Angestellte mit dem n-ten Gehalt geben kann (und diese somit auch ausgegeben werden müssen).