

〈10주차 실습〉

# PL/SQL 실습

---

Database Programming



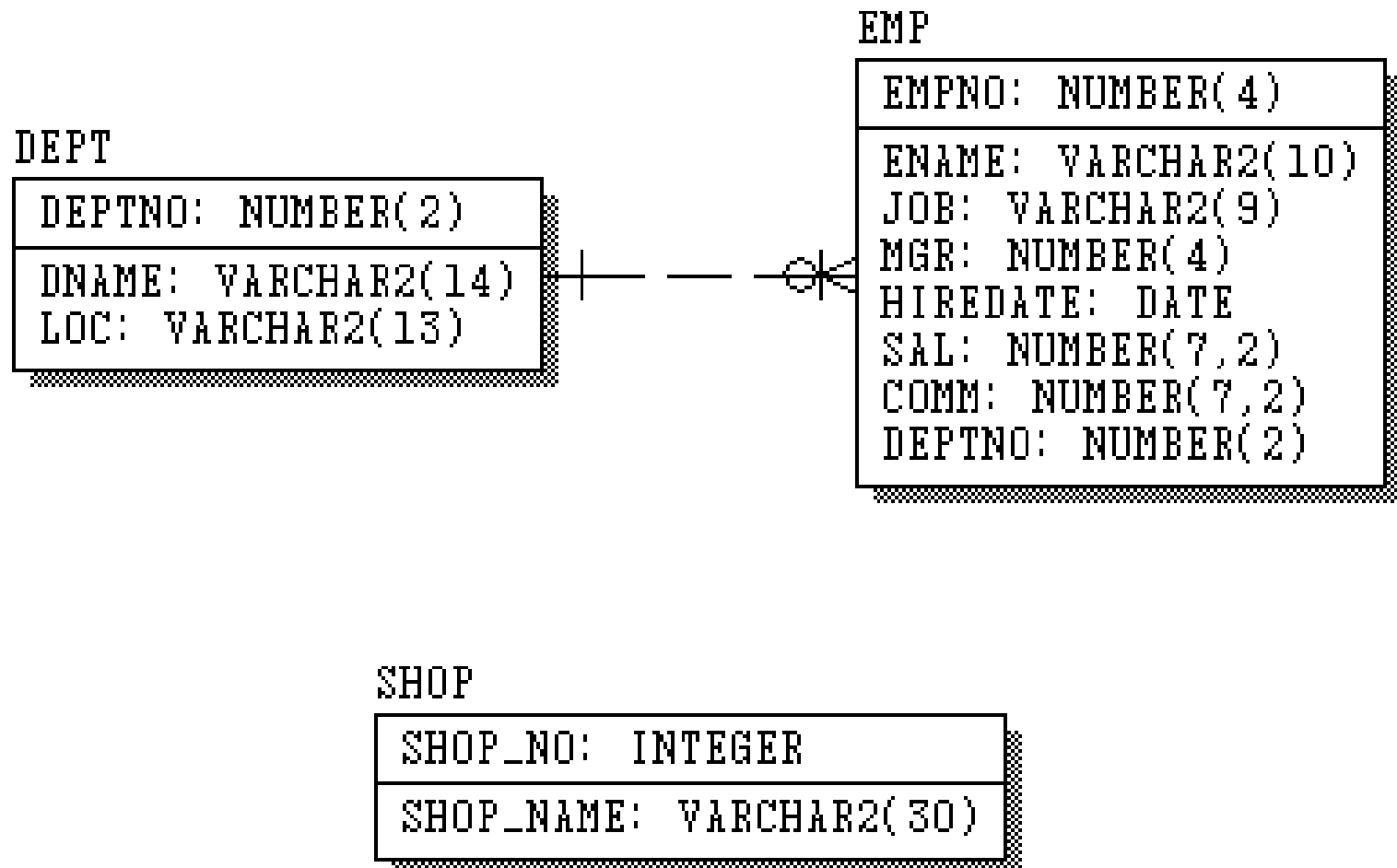
PL/SQL 데이터 타입



PL/SQL 제어 구조



실습 과제



- PL/SQL의 결과를 확인하기 위해 출력하는 방법
  - Set ServerOutput On;
  - DBMS\_OUTPUT.PUT\_LINE(...);

```
SQL> Set ServerOutput On;
SQL> BEGIN
  2  DBMS_OUTPUT.PUT_LINE('HELLO');
  3  END;
  4  /
HELLO

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
2      v_empno emp.empno%TYPE := 7900;
3      v_ename emp.ename%TYPE;
4      v_sal emp.sal%TYPE;
5  BEGIN
6      SELECT empno, ename, sal
7      INTO v_empno, v_ename, v_sal
8      FROM emp
9      WHERE empno = v_empno;
10     DBMS_OUTPUT.PUT_LINE('사원번호 : ' || v_empno);
11     DBMS_OUTPUT.PUT_LINE('사원번호 : ' || v_ename);
12     DBMS_OUTPUT.PUT_LINE('사원번호 : ' || v_sal);
13 END;
14 /
사원번호 : 7900
사원번호 : JAMES
사원번호 : 950

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
  2      v_emp emp%ROWTYPE;
  3  BEGIN
  4      SELECT empno, ename
  5      INTO v_emp.empno, v_emp.ename
  6      FROM emp
  7      WHERE empno = 7900;
  8      DBMS_OUTPUT.PUT_LINE('번호 : ' || v_emp.empno);
  9      DBMS_OUTPUT.PUT_LINE('이름 : ' || v_emp.ename);
 10  END;
 11  /
```

```
번호 : 7900
이름 : JAMES
```

PL/SQL 처리가 정상적으로 완료되었습니다.

```
SQL> DECLARE
2      TYPE empno_table IS TABLE OF emp.empno%TYPE INDEX BY BINARY_INTEGER;
3      TYPE ename_table IS TABLE OF emp.ename%TYPE INDEX BY BINARY_INTEGER;
4      empno_tab empno_table;
5      ename_tab ename_table;
6      i BINARY_INTEGER := 0;
7  BEGIN
8      FOR emp_list IN(SELECT empno, ename FROM emp WHERE deptno = 20) LOOP
9          i := i + 1;
10         empno_tab(i) := emp_list.empno;
11         ename_tab(i) := emp_list.ename;
12         DBMS_OUTPUT.PUT_LINE('번호 : ' || empno_tab(i));
13         DBMS_OUTPUT.PUT_LINE('이름 : ' || ename_tab(i));
14     END LOOP;
15 END;
16 /
번호 : 7369
이름 : SMITH
번호 : 7566
이름 : JONES
번호 : 7788
이름 : SCOTT
번호 : 7876
이름 : ADAMS
번호 : 7902
이름 : FORD
```

PL/SQL 처리가 정상적으로 완료되었습니다.

```
SQL> DECLARE
  2      TYPE emp_record IS RECORD(
  3          v_empno NUMBER,
  4          v_ename VARCHAR2(30));
  5      emp_rec emp_record;
  6  BEGIN
  7      SELECT empno, ename
  8      INTO emp_rec.v_empno, emp_rec.v_ename
  9      FROM emp
 10      WHERE empno = 7698;
 11      DBMS_OUTPUT.PUT_LINE('번호 : ' || emp_rec.v_empno);
 12      DBMS_OUTPUT.PUT_LINE('이름 : ' || emp_rec.v_ename);
 13  END;
 14  /
번호 : 7698
이름 : BLAKE

PL/SQL 처리가 정상적으로 완료되었습니다.
```



```
SQL> DECLARE
  2      sMonth CHAR(2);
  3  BEGIN
  4      SELECT TO_CHAR(SYSDATE, 'MM')
  5      INTO sMonth
  6      FROM DUAL;
  7
  8      IF (sMonth >= '03' and sMonth <= '08') THEN
  9          DBMS_OUTPUT.PUT_LINE('1학기');
 10      ELSE
 11          DBMS_OUTPUT.PUT_LINE('2학기');
 12      END IF;
 13  END;
 14  /
2학기

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
2      i NUMBER := 0;
3      nSum NUMBER := 0;
4  BEGIN
5      LOOP
6          i := i + 1;
7          nSum := nSum + i;
8          EXIT WHEN i >= 100;
9      END LOOP;
10
11      DBMS_OUTPUT.PUT_LINE('1~100까지의 합 : ' || TO_CHAR(nSum));
12  END;
13  /
1~100까지의 합 : 5050

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
2      i NUMBER := 0;
3      nSum NUMBER := 0;
4  BEGIN
5      FOR i IN 1..100 LOOP
6          nSum := nSum + i;
7      END LOOP;
8
9      DBMS_OUTPUT.PUT_LINE('1~100까지의 합 : ' || TO_CHAR(nSum));
10 END;
11 /
1~100까지의 합 : 5050

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
2       i NUMBER := 0;
3       nSum NUMBER := 0;
4 BEGIN
5       WHILE i < 100 LOOP
6           i := i + 1;
7           nSum := nSum + i;
8       END LOOP;
9
10      DBMS_OUTPUT.PUT_LINE('1~100까지의 합 : ' || TO_CHAR(nSum));
11 END;
12 /
1~100까지의 합 : 5050

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
2      i NUMBER := 0;
3  BEGIN
4      FOR i IN 1..50 LOOP
5          IF i = 30 THEN
6              GOTO my_label;
7          END IF;
8      END LOOP;
9
10     <<my_label>>
11     DBMS_OUTPUT.PUT_LINE('i = 30');
12 END;
13 /
i = 30

PL/SQL 처리가 정상적으로 완료되었습니다.
```

```
SQL> DECLARE
2      i NUMBER := 0;
3      nValue NUMBER := 0;
4  BEGIN
5      FOR i IN 1..100 LOOP
6          nValue := 1000/i;
7      END LOOP;
8  EXCEPTION
9      WHEN ZERO_DIVIDE THEN
10         NULL;
11  END;
12  /
```

PL/SQL 처리가 정상적으로 완료되었습니다.

**(%TYPE, MAX , MIN 사용)**

**Q2> 아래 그림과 같이 출력하는 SCRIPT를 작성하시오. (FOR문 사용)**

```
*  
**  
***  
****  
*****  
******  
*******  
********  
*********  
*****
```

PL/SQL 처리가 정상적으로 완료되었습니다.

**Q3> SALGRADE 테이블을 참조하여 사원 번호가 '7698' 인 급여의 등급을 출력하는 SCRIPT를 작성하시오. (FOR문, IF문 사용)**

- 제출 방식 : E-Class를 통하여 제출
- 제출 내용 : spool file
- 제출 형식 : 학번\_이름\_주차
  - Ex) 학번\_홍길동\_10주차.sql
- 제출 기한 : 수업 시작 시간으로 부터 24시간 이내 제출
  - 제출 기한 위반 시 감점 기준
    - 지각 제출 시 과제 점수에서 40% 감점
    - 1일 초과 당 10% 추가 감점 (단, 7일 이후 제출 불가)