Anonymní procedura Výjimky

**DECLARE**

v\_variable VARCHAR(20);

**BEGIN**

SELECT … INTO v\_variable

FROM…

**END;**

//v\_variable Student.name%TYPE

//v\_variable student%ROWTYPE

DECLARE

BEGIN

**EXCEPTION**

**WHEN** excep\_name **THEN** something

END;

//WHEN OTHERS THEN – pro všechny

Transakce Vyvolání výjimky

DECLARE

**name\_of\_excep EXCEPTION;**

BEGIN

**RAISE name\_of\_excep;**

EXCEPTION

WHEN **name\_of\_excep** THEN something

END;

//WHEN OTHERS THEN – pro všechny

BEGIN

…

**COMMIT**

EXCEPTION

WHEN OTHERS THEN **ROLLBACK;**

Pojmenována procedura

**CREATE [OR REPLACE] PROCEDURE proc\_name(param\_name VARCHAR2)**

**AS**

v\_variable student.name%TYPE;

BEGIN

…

END;

EXECUTE proc\_name(‚slovo‘);

//EXEC proc\_name(‚slovo‘);

Funkce

CREATE [OR REPLACE] **FUNCTION func\_name(param\_name VARCHAR2)**

**RETURN VARCHAR2**

AS

v\_variable student.name%TYPE

BEGIN

…

**RETURN v\_varchar;**

END;

//CREATE [OR REPLACE] FUNCTION func\_name(param\_name **IN/OUT** VARCHAR2)

Volání funkcí

SELECT func\_name(‚slovo‘)

FROM DUAL;

EXEC func\_name(‚slovo‘);

BEGIN

func\_name(‚slovo‘);

END;

Triggery

CREATE [OR REPLACE] **TRIGGER** trig\_name

**BEFORE | INSTEAD OF | AFTER**

**INSERT | UPDATE | DELETE** // INSERT **OR** UPDATE

**ON table\_name**

**[REFERENCING OLD AS stara\_hodnota NEW AS nova\_hodnota]**

**[FOR EACH ROW [WHEN (podmínka)]]**

BEGIN

…

END;

//bez REFERENCING hodnoty pojmenované jako :OLD :NEW //:OLD.name

Dynamické SQL

DECLARE

v\_var VARCHAR2(50);

BEGIN

v\_var := ‘DROP TABLE Book‘;

**EXECUTE IMMEDIATE v\_var;**

END;

DECLARE

BEGIN

**EXECUTE IMMEDIATE** ‘CREATE TABLE …‘;

END;

Proměnné (nedoporučená verze)

CREATE [OR REPLACE] PROCEDURE proc\_name(param\_name VARCHAR2)

AS

BEGIN

EXECUTE IMMEDIATE ‘SELECT \* FROM Student

WHERE login = ‘ || param\_name;

END;

Proměnné (doporučená verze)

CREATE [OR REPLACE] PROCEDURE proc\_name(param\_name VARCHAR2)

AS

BEGIN

EXECUTE IMMEDIATE ‘SELECT \* FROM Student

WHERE login = **:name**‘ **USING param\_name**

END;

//EXECUTE IMMEDIATE v\_var USING param\_name

Větvení IF CASE

**IF podmínka**

**THEN …**

**[ELSIF podmínka THEN …]**

**[ELSE …]**

**END IF;**

**CASE**

**WHEN … THEN …**

**WHEN … THEN …**

**ELSE …**

**END CASE;**

Cykly

**LOOP**

**…**

**EXIT WHEN podmínka**

**…**

**END LOOP;**

**WHILE podmínka**

LOOP

…

END LOOP;

**FOR v\_i IN 0..5**

LOOP

…

END LOOP;

Kurzory (varianta 1)

DECLARE

**CURSOR c\_name IS** SELECT \* FROM Student;

v\_record Student%ROWTYPE

BEGIN

**OPEN c\_name**

LOOP

**FETCH c\_name INTO v\_record;**

EXIT WHEN **c\_name%NOTFOUND;**

END LOOP;

**CLOSE c\_name;**

END;

//%NOTFOUND => true pokud není další záznam

//%ROWCOUNT => počet záznamů získaných příkazem fetch

Kurzory (varianta 2)

BEGIN

FOR **name\_temp IN** (SELECT \* FROM …)

LOOP

…

END LOOP;

Balíky (Package)

CREATE [OR REPLACE] **PACKAGE pack\_name**

AS

**//deklarace proměnných, podprogramů, typů…**

END;

CREATE [OR REPLACE] **PACKAGE BODY pack\_name**

AS

**//kód balíku**

END;