

# PL/SQL Temelleri

Oturum -2/3

PL/SQL'e Giriş

# PL/SQL

- Procedural Language / Structured Query Language
- Veritabanı programlama için özelleşmiş bir dil.

[www.canburaktumer.com/istanbulhs/index.html](http://www.canburaktumer.com/istanbulhs/index.html)

# Blocks

- Unnamed, Named
- Can be nested

```
DECLARE  
[variable&cursor def]  
BEGIN  
[commands]  
END [block_name];
```

# Variables

- Char, varchar, varchar2, number, boolean
- Variable\_name data\_type [:= value];

```
First_name varchar2(20) := 'Canburak';
```

# Procedure

```
CREATE [OR REPLACE] PROCEDURE procedure_name
    [ (parameter [,parameter]) ]
IS
    [declaration_section]
BEGIN
    executable_section
[EXCEPTION
    exception_section]
END [procedure_name];

Exec procedure_name([parameter]);
```

# Function

```
CREATE [OR REPLACE] FUNCTION function_name
    [ (parameter [,parameter]) ]
    RETURN return_datatype
IS | AS
    [declaration_section]
BEGIN
    executable_section
[EXCEPTION
    exception_section]
END [function_name];
```

# Conditional Statement (IF – CASE)

- `IF condition THEN statement  
ELSIF condition THEN statement  
ELSE statement  
END IF;`
- `CASE WHEN condition THEN statement  
WHEN condition THEN statement  
ELSE statement  
END;`
- `CASE expression WHEN condition THEN  
statement  
WHEN condition THEN statement  
ELSE statement  
END;`

# Cursor

- **Cursor cursor\_name is select\_statement;**

```
Cursor c1 is select * from employees where department_id =  
80;
```

```
Open c1;
```

```
Fetch c1 into [variable_list];
```

```
Close c1;
```



# Loops (WHILE – FOR)

- LOOP  
statements  
EXIT WHEN condition  
END LOOP;
- FOR value IN lower\_limit..upper\_limit  
LOOP  
statements  
END LOOP;
- WHILE condition  
LOOP  
statements  
END LOOP;

# FOR with Cursor

- FOR index IN cursor  
LOOP  
statements  
END LOOP;

# Dynamic SQL

- EXECUTE IMMEDIATE dyn\_sql;

```
V_dept_id :=80;
```

```
V_dyn_sql := `select * from employees where department_id  
= `|| v_dept_id;
```

```
Execute immediate v_dyn_sql;
```

# Exception Handling

- EXCEPTION WHEN exception\_type  
THEN handling\_code
- System Exceptions :  
[http://www.techonthenet.com/oracle/exceptions/named\\_system.php](http://www.techonthenet.com/oracle/exceptions/named_system.php)
- User Exceptions : RAISE exception\_name
- Worst Practice :  
EXCEPTION WHEN OTHERS  
THEN NULL;

# Package

- Package

```
CREATE PACKAGE pkg_name IS  
[function OR procedure definitions]  
END pkg_name;
```

- Package Body

```
CREATE PACKAGE BODY pkg_name IS  
[function OR procedure implementations]  
END pkg_name;
```