

Datenbank und Informationssysteme

DI. Hilbe Klaus, MBA

Organisatorisches

- Tools mit denen wir arbeiten
 - Oracle Datenbank
 - in einer virtuellen Maschine
 - (Oracle) SQL Developer

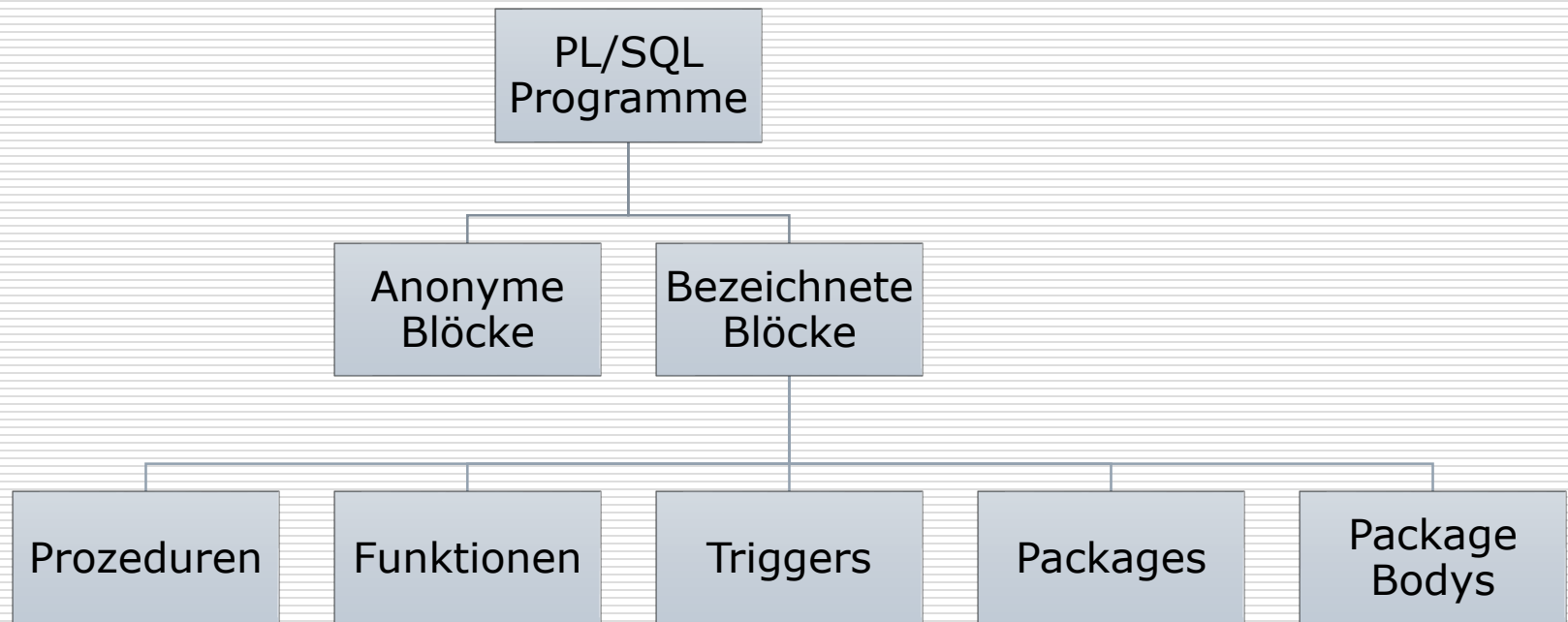
Was werden wir machen

- Programmieren von Datenbanken mit PL/SQL
 - SQL steht für „Structured Query Language“
 - PL steht für „Procedural Language“
- Prozedurale Programmierung – Was ist das?
 - Algorithmen werden in überschaubare Teile zerlegt, die anhand einer definierten Schnittstelle aufrufbar sind
 - Diese überschaubaren Teile werden in sogenannten Funktionen oder Prozeduren umgesetzt
 - Im Gegensatz zu objektorientierter Programmierung werden Daten und Funktionen nicht zu Objekten zusammengefasst

PL/SQL

- ❑ Seit 1991, ausgelegt um mit Datenbanken zu arbeiten
- ❑ Ist eine proprietäre Programmiersprache für Oracle Datenbanken (ähnliche Sprachen gibt es bei anderen Datenbanksystemen bspw. TSQL für Microsoft SQL Server oder PL/pgSQL für PostgreSQL)
- ❑ Sprache ist nicht case sensitive
- ❑ SQL Statements werden in den Programmcode integriert und zur Kompilierung überprüft

PL/SQL Programme



Struktur eines PL/SQL Programms

- PL SQL besteht aus Programmblöcken, wobei jeder Block aus folgenden Teilen besteht:

DECLARE -- optional
 <Deklarationsteil>

BEGIN -- verpflichtend
 <Eigentlicher Programmcode>

EXCEPTION -- optional
 <Fehler Behandlung>

END;

Ein einfaches PL/SQL Programm

Ausgabe von „Hello World!“



'--' Kommentarzeichen, alternativ dazu kann auch
'/* */' für Zeichenblöcke verwendet werden

```
-- Gibt "Hello World!" aus
```

```
BEGIN
```

```
    dbms_output.Put_line('Hello World!');
```

```
END;
```

Eventuell muss zusätzlich vorher noch die
BildschirmAusgabe aktiviert werden

```
Set serveroutput ON;
```

Ein einfaches PL/SQL Programm Mit Deklarationsteil

Ausgabe von „Hello World!“

```
-- Gibt 'Hello World' aus
Set serveroutput ON;
DECLARE
    v_test VARCHAR2(20) := 'Hello World!';
BEGIN
    dbms_output.Put_line(v_test);
END;
```

':=' ist der Zuweisungsoperator



';' markiert das
Ende eines Befehls



Deklarationsteil

Variablen

- ❑ Der Deklarationsteil startet mit dem Schlüsselwort **DECLARE**
- ❑ Variablen, Cursor, Konstanten **müssen** im Deklarationsteil deklariert werden
- ❑ Variablen haben eine Bezeichnung, einen Datentyp und speichern „variablen“ Inhalt

Variablen_Name [CONSTANT] *datentyp*
[NOT NULL] [:= | DEFAULT *initialer_Wert*]

CONSTANT ist die Variable eine Konstante

NOT NULL Variable darf nicht Null sein

DEFAULT Default Wert für die Variable, stattdessen kann auch der Zuweisungsoperator „:=“ verwendet werden

Deklarationsteil

Die wichtigsten Datentypen

- Numerischen Datentypen
 - Number: Gleitkommazahlen
 - Decimal: Fixkommazahlen
 - Integer: Ganze Zahlen
- Zeichen Datentypen
 - Char: Zeichenkette mit fixer Länge
 - Varchar2: Zeichenkette mit variabler Länge
 - NVarchar2: Zeichenkette mit variabler Länge von Unicode Zeichen (nationale Zeichen wie ü,ä,ß,...)*
- Datum
 - Date
- Wahrheitswert
 - Boolean: True, False or Null als mögliche Werte

* Siehe dazu auch: <https://de.wikipedia.org/wiki/Unicode>

Ausführungsteil

- ❑ Beinhaltet den ausführbaren Programmcode
- ❑ Beginnt mit dem Schlüsselwort **BEGIN** und ende mit **END**
- ❑ Ist der einzige **verpflichtende Block**
- ❑ SQL und DML Befehle können hier ausgeführt werden

Ausführungsteil

Package – dbms_output

- ❑ „dbms_output“ ist ein vordefiniertes Package
- ❑ die Ausgabe wird in einen Puffer geschrieben und nach der Ausgabe lesen die Programme dies aus
- ❑ Ausgabe ist nur in SQL*Plus oder SQL Worksheet (Oracle SQL Developer) möglich

Befehle:

```
dbms_output.put_line (...); -- Ausgabe mit Zeilenumbruch
dbms_output.put (...);     -- Ausgabe ohne Zeilenumbruch
dbms_output.enable (...);  -- max. Puffergrösse in Byte
dbms_output.new_line;      -- Zeilenumbruch
```

Ausführungsteil

Package – „dbms_output“

- ❑ Wird zum entwickeln verwendet
- ❑ Zur Ausgabe von Strings müssen diese in einfachen Hochkommas übergeben werden
- ❑ Variablen die keine Strings sind sollten mit `to_char()` umgewandelt werden bevor Sie ausgegeben werden
- ❑ `||` fügt zwei Zeichenketten zusammen
- ❑ will man ein Hochkomma ausgegeben werden zwei Hochkomma's hintereinander ausgegeben (`''`)

Fehlerbehandlung

- ❑ Tritt zur Laufzeit ein Fehler auf, so wird eine **Exception** ausgelöst
- ❑ Jeder Exception ist eine Datenbank - **Fehlernummer** zugeordnet
- ❑ Im Exception Teil kann auf das Auslösen einer Exception reagiert

```
DECLARE
```

```
    v_zaeher  INTEGER := 5;
```

```
    v_divisor INTEGER := 0;
```

```
BEGIN
```

```
    dbms_output.Put_line ('Das Ergebnis der Division ist: '  
                          || v_zaeher / v_divisor);
```

```
EXCEPTION
```

```
    WHEN zero_divide THEN
```

```
        dbms_output.Put_line ('Division durch 0 ist nicht erlaubt');
```

```
END;
```

Aufgaben

- Schreiben Sie einen PL/SQL Block welches die Fläche und den Umfang von einem Kreis mit einem Radius von 10 ausgibt
- Was müssen Sie tun um die Werte mit einem Radius von 20 auszugeben