

Datenbanken 2

Dokumentation zu Übung 2

Fabian Uhlmann
Diana Irmischer

30. Mai 2016

Aufgabe 2

1. Möglichkeit

— *DDL for Type ADRESSE_T*

```
CREATE OR REPLACE TYPE "ADRESSE_T" AS OBJECT(  
  Strasse  VARCHAR2(30) ,  
  HNR  INTEGER,  
  PLZ  INTEGER,  
  Ort  VARCHAR2(20)) NOT FINAL;
```

/

— *DDL for Type KONTENLISTE*

```
CREATE OR REPLACE TYPE "KONTENLISTE" as table of REF Konto_t;
```

/

— *DDL for Type KONTENLISTE*

```
CREATE OR REPLACE TYPE "KONTENLISTE" as table of REF Konto_t;
```

/

— *DDL for Type KONTO_T*

```
CREATE OR REPLACE TYPE "KONTO_T" AS OBJECT(  
  KNR  INTEGER,  
  Kontostand  NUMBER(10,2) ,  
  Typ  INTEGER(1)) NOT FINAL;
```

/

— *DDL for Type PERSON_T*

```
CREATE OR REPLACE TYPE "PERSON_T" AS OBJECT(  
  Kd_NR  INTEGER,  
  Nachname  VARCHAR2(20) ,  
  Vorname  VARCHAR2(20) ,  
  Adresse  REF  Adresse_t ,  
  Status  INTEGER(1) ,  
  Konten  Kontenliste) NOT FINAL;
```

/

— *DDL for Type ZWEIGSTELLE_T*

CREATE OR REPLACE TYPE "ZWEIGSTELLE_T" **AS** OBJECT(
ZName **VARCHAR2**(20),
Adresse REF Adresse_t ,
Leiter **Integer**,
Konten Kontenliste) **NOT FINAL**;

/

— *DDL for Table ADRESSEN*

CREATE TABLE "ADRESSEN" OF "ADRESSE_T"

— *DDL for Table KONTEN*

CREATE TABLE "KONTEN" OF "KONTO_T"

— *DDL for Table KUNDEN*

CREATE TABLE "KUNDEN" OF "PERSON_T"
NESTED TABLE "KONTEN" **STORE AS** "KONTEN_KD"

— *DDL for Table ZWEIGSTELLEN*

CREATE TABLE "ZWEIGSTELLEN" OF "ZWEIGSTELLE_T"
NESTED TABLE "KONTEN" **STORE AS** "KONTEN_ZWEIGST"

— *Inserts in Table ADRESSEN*

```

Insert into ADRESSEN (STRASSE,HNR,PLZ,ORT) values ( 'Simseestr.', '3', '80800', 'Musterhausen' );
Insert into ADRESSEN (STRASSE,HNR,PLZ,ORT) values ( 'Hochstr.', '1', '80000', 'Muenchhausen' );
Insert into ADRESSEN (STRASSE,HNR,PLZ,ORT) values ( 'Eschenweg', '12', '80335', 'Muenchen' );
Insert into ADRESSEN (STRASSE,HNR,PLZ,ORT) values ( 'Muenchnerstr.', '33', '80801', 'Muenchen' );
Insert into ADRESSEN (STRASSE,HNR,PLZ,ORT) values ( 'Schellingstr.', '42', '53620', 'Hasenbuettel' );

```

— *Inserts in Table KONTEN*

```

Insert into KONTEN (KNR,KONTOSTAND,TYP) values ( '120768', '234,56', '0' );
Insert into KONTEN (KNR,KONTOSTAND,TYP) values ( '348973', '12567,56', '1' );
Insert into KONTEN (KNR,KONTOSTAND,TYP) values ( '678453', '-456,78', '1' );
Insert into KONTEN (KNR,KONTOSTAND,TYP) values ( '987654', '789,65', '1' );
Insert into KONTEN (KNR,KONTOSTAND,TYP) values ( '745363', '-23,67', '0' );

```

— *Inserts in Table KUNDEN*

```

Insert into KUNDEN values (PERSON_T( '2345', 'Fach', 'Hans', (SELECT REF(adr) FROM ADRESSEN adr WHERE adr.ST
Insert into TABLE (SELECT KONTEN FROM KUNDEN WHERE KD_NR = '2345')(SELECT REF(k) FROM KONTEN k WHERE k.K
Insert into KUNDEN values (PERSON_T( '7654', 'Meier', 'Bernd', (SELECT REF(adr) FROM ADRESSEN adr WHERE adr.
Insert into KUNDEN values (PERSON_T( '8764', 'Wiesner', 'Jan', (SELECT REF(adr) FROM ADRESSEN adr WHERE adr.
Insert into TABLE (SELECT KONTEN FROM KUNDEN WHERE KD_NR = '2345')(SELECT REF(k) FROM KONTEN k WHERE k.K

```

— *Inserts in Table ZWEIGSTELLEN*

```

Insert into ZWEIGSTELLEN values ( 'Bachdorf', (SELECT REF(adr) FROM ADRESSEN adr WHERE adr.strasse = 'Hoc

```

Insert into TABLE (SELECT KONTEN FROM ZWEIGSTELLEN WHERE ZNAME = 'Bachdorf')(SELECT REF(k) FROM KONTEN k WHERE k.ZNAME = 'Bachdorf')
 Insert into ZWEIGSTELLEN values ('Riedering',(SELECT REF(adr) FROM ADRESSEN adr WHERE adr.strasse = 'Simseestr. 3'))
 Insert into TABLE (SELECT KONTEN FROM ZWEIGSTELLEN WHERE ZNAME = 'Riedering')(SELECT REF(k) FROM KONTEN k WHERE k.ZNAME = 'Riedering')

— *SELECT Statement gibt "alle Kontonummern mit Kontostand, Art und Adresse der Zweigstelle" aus*

SELECT z.ZNAME **AS** Zweigstellennamen, Deref(z.ADRESSE).STRASSE **AS** Strasse, Deref(z.ADRESSE).HNR **AS** Hausnummer, z.TYP

—> *Result*

ZWEIGSTELLENNAME	STRASSE	HAUSNUMMER	PLZ	ORT	KONTONUMMER	KONTOSTAND	TYP
Bachdorf	Hochstr.	1	80000	Muenchhausen	120768	234,56	0
Bachdorf	Hochstr.	1	80000	Muenchhausen	348973	12567,56	1
Bachdorf	Hochstr.	1	80000	Muenchhausen	678453	-456,78	1
Riedering	Simseestr.	3	80800	Musterhausen	987654	789,65	1
Riedering	Simseestr.	3	80800	Musterhausen	745363	-23,67	0

— *SELECT Statement gibt "alle Kontonummern mit Adresse aller Kontoinhaber" aus*

SELECT kd.VORNAME **AS** Vorname, kd.NACHNAME **AS** Nachname, Deref(k.COLUMN.VALUE).KNR **AS** Kontonummer, Deref(k.COLUMN.VALUE).ZNAME

—> *Result*

VORNAME	NACHNAME	KONTONUMMER	STRASSE	HAUSNUMMER	PLZ	ORT
Bernd	Meier	987654	Eschenweg	12	80335	Muenchen
Hans	Fach	348973	Muenchnerstr.	33	80801	Muenchen
Hans	Fach	120768	Muenchnerstr.	33	80801	Muenchen
Jan	Wiesner	745363	Schellingstr.	42	53620	Hasenbuettel
Jan	Wiesner	348973	Schellingstr.	42	53620	Hasenbuettel
Jan	Wiesner	678453	Schellingstr.	42	53620	Hasenbuettel

```
--  SELFJOIN
```

```
select kd1.vorname as VORNAME, kd1.nachname as NACHNAME, kd2.vorname, kd2.nachname, deref(k1.COLUMN_VALU
```

2. Möglichkeit

— *DDL for Type ADRESSE_T2*

```
CREATE OR REPLACE TYPE "ADRESSE_T2" AS OBJECT(  
  Strasse  VARCHAR2(30) ,  
  HNR INTEGER,  
  PLZ INTEGER,  
  Ort VARCHAR2(20)) NOT FINAL;
```

/

— *DDL for Type PERSON_T2*

```
CREATE OR REPLACE TYPE "PERSON_T2" AS OBJECT(  
  KdNR INTEGER,  
  Nachname  VARCHAR2(20) ,  
  Vorname   VARCHAR2(20) ,  
  Adresse REF  Adresse_t2 ,  
  Status    INTEGER(1)) NOT FINAL;
```

/

— *DDL for Type KUNDENLISTE2*

```
CREATE OR REPLACE TYPE "KUNDENLISTE2" as table of REF PERSON_t2;%
```

/

— *DDL for Type KONTO_T2*

```
CREATE OR REPLACE TYPE "KONTO_T2" AS OBJECT(  
  KNR INTEGER,  
  Kontostand NUMBER(10,2) ,  
  Typ INTEGER(1) ,  
  Kunden2 Kundenliste2) NOT FINAL;
```

/

— *DDL for Type KONTENLISTE2*

```
CREATE OR REPLACE TYPE "KONTENLISTE2" as table of Konto_t2;
```

/

— *DDL for Type ZWEIGSTELLE_T2*

CREATE OR REPLACE TYPE "ZWEIGSTELLE_T2" **AS OBJECT**(
ZName **VARCHAR2**(20),
Adresse2 **REF** Adresse_t2 ,
Leiter **Integer**,
Konten2 Kontenliste2) **NOT FINAL**;

/

— *DDL for Table ADRESSEN2*

CREATE TABLE "ADRESSEN2" **OF** "ADRESSE_T2"

— *DDL for Table KUNDEN2*

CREATE TABLE "KUNDEN2" **OF** "PERSON_T2"

— *DDL for Table ZWEIGSTELLEN*

CREATE TABLE "ZWEIGSTELLEN2" **OF** "ZWEIGSTELLE_T2"
NESTED TABLE "KONTEN2" **STORE AS** "KONTEN_ZWEIGST2" (**NESTED TABLE** "KUNDEN2" **STORE AS** "KUNDEN_ZWEIGST2")

— *Inserts in Table ADRESSEN2*

```

Insert into ADRESSEN2 (STRASSE,HNR,PLZ,ORT) values ( 'Simseestr.', '3', '80800', 'Musterhausen' );
Insert into ADRESSEN2 (STRASSE,HNR,PLZ,ORT) values ( 'Hochstr.', '1', '80000', 'Muenchhausen' );
Insert into ADRESSEN2 (STRASSE,HNR,PLZ,ORT) values ( 'Eschenweg', '12', '80335', 'Muenchen' );
Insert into ADRESSEN2 (STRASSE,HNR,PLZ,ORT) values ( 'Muenchnerstr.', '33', '80801', 'Muenchen' );
Insert into ADRESSEN2 (STRASSE,HNR,PLZ,ORT) values ( 'Schellingstr.', '42', '53620', 'Hasenbuettel' );

```

— *Inserts in Table KUNDEN2*

```

Insert into KUNDEN2 values ( '2345', 'Fach', 'Hans', (SELECT REF(adr) FROM ADRESSEN2 adr WHERE adr.STRASSE = 'H
Insert into KUNDEN2 values ( '7654', 'Meier', 'Bernd', (SELECT REF(adr) FROM ADRESSEN2 adr WHERE adr.STRASSE
Insert into KUNDEN2 values ( '8764', 'Wiesner', 'Jan', (SELECT REF(adr) FROM ADRESSEN2 adr WHERE adr.STRASSE

```

— *Inserts in Table ZWEIGSTELLEN*

```

Insert into ZWEIGSTELLEN2 values ( 'Bachdorf', (SELECT REF(adr) FROM ADRESSEN2 adr WHERE adr.STRASSE = 'H
—> Bemerkung: INSERT into table (SELECT z.KONTEN2 FROM ZWEIGSTELLEN2 z WHERE z.ZNAME = 'BACHDORF') values
Insert into ZWEIGSTELLEN2 values ( 'Riedering', (SELECT REF(adr) FROM ADRESSEN2 adr WHERE adr.STRASSE = 'S

```

— *befuellen der jeweiligen NESTED TABLE Kundenliste2*

```

Insert into TABLE (SELECT k.KUNDEN2 FROM ZWEIGSTELLEN2 z, TABLE(z.KONTEN2) k WHERE z.ZNAME = 'Bachdorf'
Insert into TABLE (SELECT k.KUNDEN2 FROM ZWEIGSTELLEN2 z, TABLE(z.KONTEN2) k WHERE z.ZNAME = 'Bachdorf'
Insert into TABLE (SELECT k.KUNDEN2 FROM ZWEIGSTELLEN2 z, TABLE(z.KONTEN2) k WHERE z.ZNAME = 'Bachdorf'

```

— *Alternativ wenn SELECT nicht direkt in Kundenliste2()*

```

—> Insert into TABLE (SELECT k.KUNDEN2 FROM ZWEIGSTELLEN2 z, TABLE(z.KONTEN2) k WHERE z.ZNAME = 'Riedering'
—> Insert into TABLE (SELECT k.KUNDEN2 FROM ZWEIGSTELLEN2 z, TABLE(z.KONTEN2) k WHERE z.ZNAME = 'Riedering'

```

— *SELECT Statement gibt "alle Kontonummern mit Konostand, Art und Adresse der Zweigstelle" aus*

SELECT z.ZNAME **AS** Filiale , Deref(z.ADRESSE2).STRASSE **AS** Strasse , Deref(z.ADRESSE2).HNR **AS** Hausnummer , DE

—> *Result:*

FILIALE	STRASSE	HAUSNUMMER	PLZ	ORT	KONTONUMMER	KONTOSTAND	TYP
Bachdorf	Hochstr.	1	80000	Muenchhausen	120768	234,56	0
Bachdorf	Hochstr.	1	80000	Muenchhausen	348973	12567,56	1
Bachdorf	Hochstr.	1	80000	Muenchhausen	678453	-456,78	1
Riedering	Simseestr.	3	80800	Musterhausen	987654	789,65	1
Riedering	Simseestr.	3	80800	Musterhausen	745363	-23,67	0

— *SELECT Statement gibt "alle Kontonummern mit Adresse aller Kontoinhaber" aus*

SELECT Deref(kd.COLUMN_VALUE).VORNAME **AS** Vorname , Deref(kd.COLUMN_VALUE).NACHNAME **AS** Nachname , Deref(DE

—> *Result:*

VORNAME	NACHNAME	STRASSE	HAUSNUMMER	PLZ	ORT	KONTONUMMER	KONTOSTAND
Hans	Fach	Muenchnerstr.	33	80801	Muenchen	120768	234,56
Hans	Fach	Muenchnerstr.	33	80801	Muenchen	348973	12567,56
Jan	Wiesner	Schellingstr.	42	53620	Hasenbuettel	348973	12567,56
Jan	Wiesner	Schellingstr.	42	53620	Hasenbuettel	678453	-456,78
Jan	Wiesner	Schellingstr.	42	53620	Hasenbuettel	987654	789,65
Bernd	Meier	Eschenweg	12	80335	Muenchen	745363	-23,67

Aufgabe 5

— *DDL for Type PHONENUMBERLIST*

CREATE OR REPLACE TYPE "PHONENUMBERLIST" as table of VARCHAR2(12);

/

— *DDL for Type CONTACT_T*

**CREATE OR REPLACE TYPE "CONTACT_T" AS OBJECT(
Street VARCHAR2(30),
HNUMBER INTEGER,
Code VARCHAR2(5),
City VARCHAR2(20),
Phone_Numbers PHONENUMBERLIST,
Fax VARCHAR2(12)) NOT FINAL;**

— *DDL for Type SHAREHOLDER_T*

**CREATE OR REPLACE TYPE "SHAREHOLDER_T" AS OBJECT(
Sholder_ID INTEGER,
Sholder_Name VARCHAR2(30),
Sholder_Contact REF Contact_T) NOT FINAL;**

/

— *DDL for Type OWN_SHARES_T*

**CREATE OR REPLACE TYPE "OWN_SHARES_T" AS OBJECT(
Sholder_ID REF Shareholder_t,
Share_Amount INTEGER) NOT FINAL;**

/

— *DDL for Type SHAREHOLDERLIST*

CREATE OR REPLACE TYPE "SHAREHOLDERLIST" as table of Own_Shares_t;

/

— *DDL for Type DIRECTORS*

```

    CREATE OR REPLACE TYPE "DIRECTORS" as table of INTEGER;

/

```

```

—   DDL for Type MANAGER_T

```

```

    CREATE OR REPLACE TYPE "MANAGER_T" AS OBJECT(
    Manag_Type VARCHAR2(30) ,
    Yearly_Salary VARCHAR2(10)) NOT FINAL;

/

```

```

—   DDL for Type MANAGER

```

```

    CREATE OR REPLACE TYPE "MANAGERLIST" as table of Manager_t;

/

```

```

—   DDL for Type MANAGEMENT_T

```

```

    CREATE OR REPLACE TYPE "MANAGEMENT_T" AS OBJECT(
    Manag_ID INTEGER,
    Manag_Name VARCHAR2(30) ,
    Manag_Contact REF Contact_T ,
    Manag_Typ_Director DIRECTORS,
    Manag_Typ_Manager MANAGERLIST) NOT FINAL;

/

```

```

—   DDL for Type MANAGEMENTLIST

```

```

    CREATE OR REPLACE TYPE "MANAGEMENTLIST" as table of Management_t

/

```

```

—   DDL for Type EMPLOYEE_T

```

```

    CREATE OR REPLACE TYPE "EMPLOYEE_T" AS OBJECT(
    Emp_ID VARCHAR2(5) ,
    Emp_Name VARCHAR2(30) ,
    Emp_Contact REF Contact_T) NOT FINAL;

/

```

— *DDL for Type EMPLOYEE_FULLTIME_T*

CREATE OR REPLACE TYPE "EMPLOYEE_FULLTIME_T" UNDER "EMPLOYEE_T" (
Annual_Wage **INTEGER**,
Emp_Bonus **INTEGER**) **NOT FINAL**;

/

— *DDL for Type EMPLOYEE_PARTTIME_T*

CREATE OR REPLACE TYPE "EMPLOYEE_PARTTIME_T" UNDER "EMPLOYEE_T" (
Weekly_Wage **INTEGER**) **NOT FINAL**;

/

— *DDL for Type EMPLOYEE_CASUAL_T*

CREATE OR REPLACE TYPE "EMPLOYEE_CASUAL_T" UNDER "EMPLOYEE_T" (
Hourly_Wage **INTEGER**) **NOT FINAL**;

/

— *DDL for Type EMPLOYEEELIST*

CREATE OR REPLACE TYPE "EMPLOYEEELIST" **as table** of Employee_t;

/

— *DDL for Type DEPARTMENT_T*

CREATE OR REPLACE TYPE "DEPARTMENT_T" **AS OBJECT**(
Dept_ID **INTEGER**,
Dept_Name **VARCHAR2**(20) ,
Dept_Head **VARCHAR2**(25) ,
Dept_Employees EMPLOYEEELIST) **NOT FINAL**;

/

— *DDL for Type DEPARTMENTS*

CREATE OR REPLACE TYPE "DEPARTMENTLIST" **as table** of Department_t;

/

— *DDL for Type TRANSACTION_T*

CREATE OR REPLACE TYPE "TRANSACTION_T" AS OBJECT(
Trans_ID **INTEGER**,
Trans_Date **DATE**,
Store_Contact REF Contact_T ,
Quantity **INTEGER**) **NOT FINAL**;

/

— *DDL for Type TRANSACTIONLIST*

CREATE OR REPLACE TYPE "TRANSACTIONLIST" as table of REF Transaction_t ;

/

— *DDL for Type ITEM_T*

CREATE OR REPLACE TYPE "ITEM_T" AS OBJECT(
Item_ID **VARCHAR2**(5) ,
Item_Name **VARCHAR2**(20) ,
Item_Desc **VARCHAR2**(20) ,
Item_Cost **VARCHAR2**(4) ,
Item_Price **VARCHAR2**(5) ,
Item_Trans TRANSACTIONLIST) **NOT FINAL**;

/

— *DDL for Type ITEMLIST*

CREATE OR REPLACE TYPE "ITEMLIST" as table of REF Item_t ;

/

— *DDL for Type AVAILABLE_IN_T*

CREATE OR REPLACE TYPE "AVAILABLE_IN_T" AS OBJECT(
Item_ID REF Item_t ,
Item_Stock **INTEGER**) **NOT FINAL**;

/

— *DDL for Type AVAILABLELIST*

CREATE OR REPLACE TYPE "AVAILABLELIST" as table of Available_In_t;

/

— *DDL for Type STORE_T*

CREATE OR REPLACE TYPE "STORE_T" AS OBJECT(
Store_ID **VARCHAR2**(5),
Store_Location **VARCHAR2**(20),
Store_Contact REF Contact_T,
Store_Manage **VARCHAR2**(30),
Store_Depts DEPARTMENTLIST,
Store_Items AVAILABLELIST) **NOT FINAL**;

/

— *DDL for Type STORELIST*

CREATE OR REPLACE TYPE "STORELIST" as table of Store_t

/

— *DDL for Type CUSTOMER_T*

CREATE OR REPLACE TYPE "CUSTOMER_T" AS OBJECT(
Cust_ID **VARCHAR2**(5),
Cust_Name **VARCHAR2**(20),
Cust_Contact REF Contact_t,
Cust_Gender **INTEGER**(1),
Cust_DOB **DATE**,
Cust_Bonus_Points **INTEGER**,
Cust_Trans TRANSACTIONLIST) **NOT FINAL**;

/

— *DDL for Type MAKER_T*

CREATE OR REPLACE TYPE "MAKER_T" AS OBJECT(
Maker_ID **VARCHAR2**(5),
Maker_Name **VARCHAR2**(20),
Maker_Contact REF Contact_T,
Items ITEMLIST) **NOT FINAL**;

/

— *DDL for Type COMP_TYPE_T*

**CREATE OR REPLACE TYPE "COMP_TYPE_T" AS OBJECT(
Typ_Desc **VARCHAR2**(30)) NOT FINAL;**

/

— *DDL for Type COMP_TYPE1_T*

**CREATE OR REPLACE TYPE "COMP_TYPE1_T" UNDER COMP_TYPE_T(
Area **VARCHAR2**(5)) NOT FINAL;**

/

— *DDL for Type COMP_TYPE2_T*

**CREATE OR REPLACE TYPE "COMP_TYPE2_T" UNDER COMP_TYPE_T(
Market **VARCHAR2**(10)) NOT FINAL;**

/

— *DDL for Type COMPANY_T*

**CREATE OR REPLACE TYPE "COMPANY_T" AS OBJECT(
Comp_ID **INTEGER**,
Comp_Name **VARCHAR2**(20),
Comp_Contact REF Contact_T,
Comp_Type COMP_TYPE_T,
Comp_Sholders SHAREHOLDERLIST,
Comp_Manag MANAGEMENTLIST,
Comp_Stores STORELIST) NOT FINAL;**