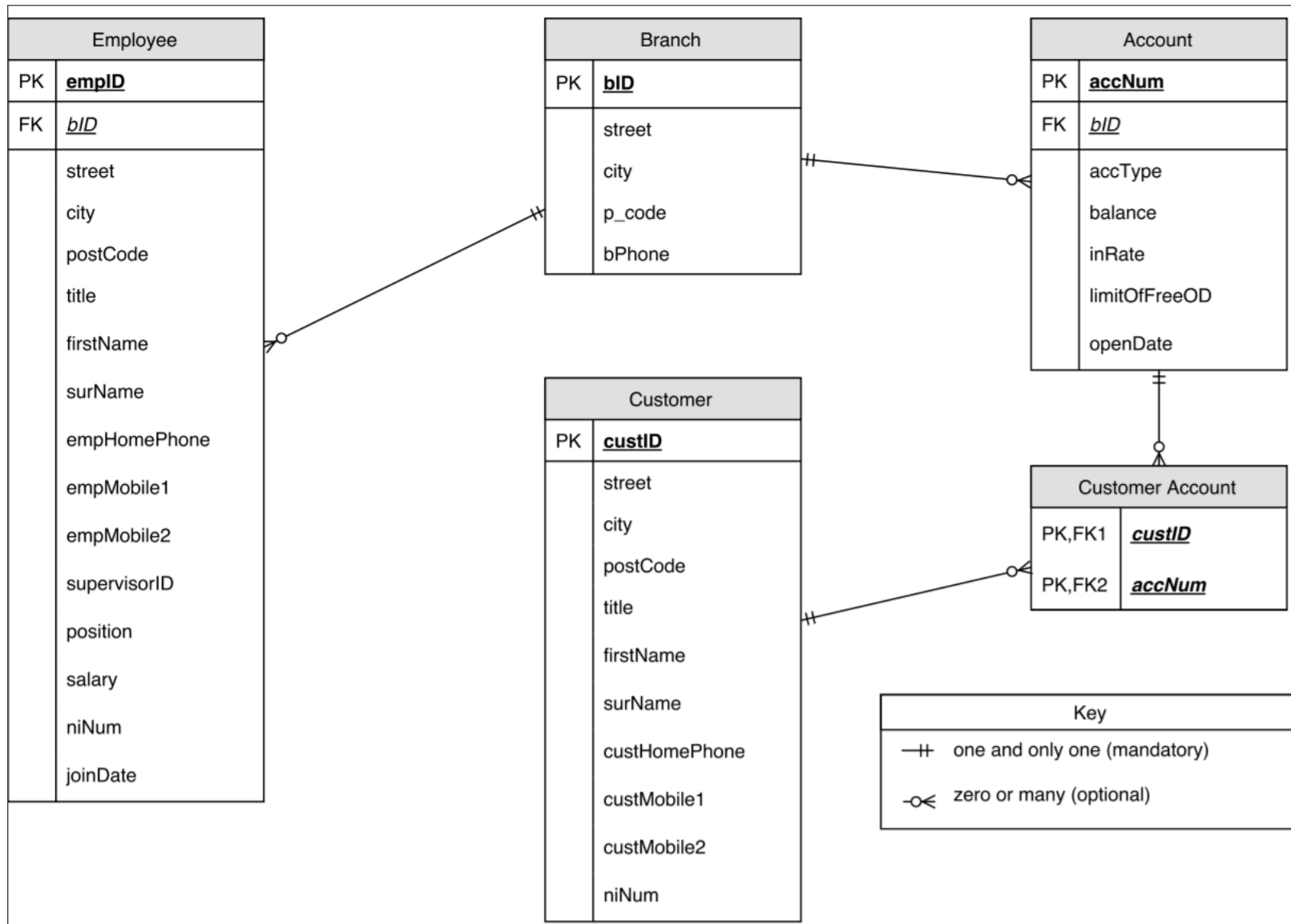


Task 1 - ER Diagram



Task 2 - Proposed Object-Relational Database Redesign

Structured Types

| | |
|-----------------------|---|
| Type | (attribute DATA_TYPE [CONSTRAINT]) |
| Name | (title VARCHAR2(8) [IS NOT NULL], firstName VARCHAR2(30) [IS NOT NULL], surName VARCHAR2(30) [IS NOT NULL]) |
| Address | (street VARCHAR2(30) [IS NOT NULL], city VARCHAR2(30) [IS NOT NULL], p_code VARCHAR2(8) [IS NOT NULL]) |
| Phone | (homePhone VARCHAR2(20) [IS NOT NULL], mobilePhone mobilePhones_nested) |
| Branch | (bID NUMBER [PRIMARY KEY], bAddress Address, bPhone VARCHAR2(20) [IS NOT NULL]) |
| Job | (position VARCHAR2(30) [CHECK IN ("Head", "Manager", "Accountant", "Leader", "Cashier")], salary NUMBER [IS NOT NULL], bID ref Branch, joinDate DATE [IS NOT NULL]) |
| Person | (pName Name, pAddress Address, pPhone Phone, niNum VARCHAR2(20) [UNIQUE]) |
| Customer UNDER Person | (custID NUMBER [PRIMARY KEY]) |
| Employee UNDER Person | (empID NUMBER [PRIMARY KEY], supervisorID NUMBER [IS NOT NULL], eJob Job) |
| Account | (accNum NUMBER [PRIMARY KEY], accType VARCHAR2(30) [CHECK IN ("current", "savings")], balance NUMBER [IS NOT NULL], bID ref Branch, inRate NUMBER [IS NOT NULL], limitOfFreeOD NUMBER [IS NOT NULL], openDate DATE [IS NOT NULL]) |
| CustomerAccount | (custID ref Customer, accNum ref Account) |

Nested Table

mobilePhones_nested (AS TABLE OF VARCHAR2(20))

- A Critical Review of the Rationale for the Proposed Object-Relational Database Design

Structured Types Used and Why

The 'Employee' and 'Customer' entities in the relational database share many attributes (*street, city, postCode, title, firstName, surName, niNum*) as well as having some similar attributes (*empHomePhone, custHomePhone, empMobile1, custMobile1, empMobile2, custMobile2*). Therefore, the structured types 'Name', 'Address' and 'Phone' were created with their relevant attributes assigned to them. Further to this a structured type 'Person' was created that used these three structured types and also has the *niNum* attribute. This allowed the final structured types

'Employee' and 'Customer' (used to make their respective tables) to inherit from 'Person' and maintain all those shared attributes as well as the attributes unique to them. The 'Address' type was also able to be used in the 'Branch' structured type for the 'Branch' table as each branch also had a street, city and post code attribute in the original design. Separate to this, the structured type 'Job' was created to further group attributes in the 'Employee' entity.

Data Types Used and Why

For most text-based attributes the data type VARCHAR2(30) was used. VARCHAR2 was picked over VARCHAR as recommended by Oracle as it can store a larger number of bytes of characters. A maximum of 30 characters was chosen as a sensible number of characters for these attributes. Exceptions to this were for the *title*, *p_code*, and all phone number attributes where a different maximum was chosen. A maximum of 8 characters was chosen for *titles* and *p_code* to reduce size as they are always 8 or less characters. For the phone number attributes (*homePhone*, *bPhone*), as well as the data type for the nested table of mobile numbers, VARCHAR2(20) was chosen as the international standard can support up to 15 characters, and this allows room for additional characters, such as brackets, that may be used. The data type NUMBER was used for all attributes that would be stored as only numbers and was chosen as it is Oracle's recommended number-based data type. Finally, the DATE data type was chosen for the two date attributes (*joinDate* and *openDate* in the structured types 'Job' and 'Account' respectively).

Inheritance Used and Why

In the redesigned system, the structured subtypes 'Employee' and 'Customer' inherit from the supertype 'Person'. The reason for this is that they shared many attributes in the original relational model and some of these were grouped into 'Person'. If new attributes are required they can be added to the 'Person' supertype once (and therefore inherited by the two subtypes) so that time will be saved when updating the database.

References Used and Why

There are 5 references used in the proposed object-relational database: in the 'CustomerAccount' structured type there are references to *custID* in the 'Customer' type and to *accNum* in the 'Account' type, as well as two references made to *bID* in the 'Branch' type by the 'Job' type and the 'Account' type. These references were chosen as replacements to the foreign key attributes in the original relational database, allowing tuples to be referred to by separate structured types. The final reference was for *supervisorID* used as a self reference to the 'Employee' type to represent an employee's supervisor's *empID*; this reference helped answer query 4g.

Constraints Used and Why

On most attributes the only constraint was the 'NOT NULL' constraint. This is to ensure that the fields are not empty so that each query returns actual data. There are, however, a few exceptions to this. 4 of the tables had a single 'PRIMARY KEY' constraint that corresponds to the primary keys used in the original relational database. In the 'Account' table the attribute *accType* is

also constrained by a CHECK to ensure it is either a “current” or “savings” account. The *niNum* attributes in both the ‘Employee’ and ‘Customer’ tables have the ‘UNIQUE’ constraint to ensure that there are no duplicate values for that attribute within its respective table. Also in the ‘Employee’ table is the *position* attribute from the ‘Job’ type which uses the CHECK constraint to ensure it is one of the specified positions as described in the scenario: Head, Manager, Leader, Accountant, Cashier.

Collections Used and Why

In both the ‘Employee’ and ‘Customer’ entities in the relational database there were two attributes to store mobile phone numbers. For the object-relation database a nested table was created to store multiple mobile phone numbers and is used by the ‘Phone’ type; a nested table was chosen over a varray as it allows more mobile numbers to be added later if needed. A varray’s size must be specified when the attribute is defined and not changed later, this would cause issues if an employee or customer wished to add more mobile numbers than the varray could store.

Task 3 - SQL Statements for Creating and Populating the Database

--Create name type

```
CREATE OR REPLACE TYPE name_typ AS OBJECT (  
    title VARCHAR2(8),  
    firstName VARCHAR2(30),  
    surName VARCHAR2(30));
```

/

--Create address type

```
CREATE OR REPLACE TYPE address_typ AS OBJECT (  
    street VARCHAR2(30),  
    city VARCHAR2(30),  
    p_code VARCHAR2(8));
```

/

--Create nested table of mobile phones

```
CREATE OR REPLACE TYPE mobilePhones_nested AS TABLE OF VARCHAR2(20)
```

/

--Create phone type

```
CREATE OR REPLACE TYPE phone_typ AS OBJECT (  
    homePhone VARCHAR2(20),  
    mobilePhone mobilePhones_nested);
```

/

--Create branch type for Branch table

```
CREATE OR REPLACE TYPE branch_typ AS OBJECT (  
    bID NUMBER,  
    bAddress address_typ,  
    bPhone VARCHAR2(20));
```

/

--Create job type

```
CREATE OR REPLACE TYPE job_typ AS OBJECT (  
    position VARCHAR2(30),  
    salary NUMBER,  
    bID ref branch_typ,  
    joinDate DATE);
```

/

--Create person type to be supertype to Customer and Employee

CREATE OR REPLACE TYPE person_typ AS OBJECT (

 pName name_typ,

 pAddress address_typ,

 pPhone phone_typ,

 niNum VARCHAR(20))

NOT FINAL;

/

--Create customer subtype for Customer table

CREATE OR REPLACE TYPE customer_typ UNDER person_typ (

 custID NUMBER);

/

--Create employee subtype for Employee table

CREATE OR REPLACE TYPE employee_typ UNDER person_typ (

 empID NUMBER,

 supervisorID ref employee_typ,

```

        eJob job_typ);

/

--Create account type for Account table
CREATE OR REPLACE TYPE account_typ AS OBJECT (
    accNum NUMBER,
    accType VARCHAR2(30),
    balance NUMBER,
    bID ref branch_typ,
    inRate NUMBER,
    limitOfFreeOD NUMBER,
    openDate DATE);

/

--Create customerAccount type for CustomerAccount table
CREATE OR REPLACE TYPE customerAccount_typ AS OBJECT (
    custID ref customer_typ,
    accNum ref account_typ);

/

```


--Create Branch table

```
CREATE TABLE branchTable OF branch_typ (  
    bID PRIMARY KEY,  
    CONSTRAINT bStreet_const CHECK(bAddress.street IS NOT NULL),  
    CONSTRAINT bCity_const CHECK(bAddress.city IS NOT NULL),  
    CONSTRAINT bP_code CHECK(bAddress.p_code IS NOT NULL),  
    CONSTRAINT bPhone_const CHECK(bPhone IS NOT NULL));
```

/

--Create Account table

```
CREATE TABLE accountTable OF account_typ (  
    accNum PRIMARY KEY,  
    CONSTRAINT accType_const CHECK(accType IN ('current', 'savings')),  
    CONSTRAINT balance_const CHECK(balance IS NOT NULL),  
    CONSTRAINT inRate_const CHECK(inRate IS NOT NULL),  
    CONSTRAINT limitOfFreeOD_const CHECK(limitOfFreeOD IS NOT NULL),  
    CONSTRAINT openDate_const CHECK(openDate IS NOT NULL));
```

/

--Create CustomerAccount table

CREATE TABLE customerAccountTable OF customerAccount_typ;

/

--Create Customer table

CREATE TABLE customerTable OF customer_typ (

 custID PRIMARY KEY,

 CONSTRAINT cTitle_const CHECK(pName.title IS NOT NULL),

 CONSTRAINT cFirstName_const CHECK(pName.firstName IS NOT NULL),

 CONSTRAINT cSurName_const CHECK(pName.surName IS NOT NULL),

 CONSTRAINT cStreet_const CHECK(pAddress.street IS NOT NULL),

 CONSTRAINT cCity_const CHECK(pAddress.city IS NOT NULL),

 CONSTRAINT cP_Code_const CHECK(pAddress.p_code IS NOT NULL),

 CONSTRAINT cNiNum_const UNIQUE(niNum))

 NESTED TABLE pPhone.mobilePhone STORE AS cMobilePhones_nested_table;

/

--Create Employee table

CREATE TABLE employeeTable OF employee_typ (

 empID PRIMARY KEY,

```

CONSTRAINT eTitle_const CHECK(pName.title IS NOT NULL),
CONSTRAINT eFirstName_const CHECK(pName.firstName IS NOT NULL),
CONSTRAINT eSurName_const CHECK(pName.surName IS NOT NULL),
CONSTRAINT eStreet_const CHECK(pAddress.street IS NOT NULL),
CONSTRAINT eCity_const CHECK(pAddress.city IS NOT NULL),
CONSTRAINT eP_Code_const CHECK(pAddress.p_code IS NOT NULL),
CONSTRAINT eNiNum_const UNIQUE(niNum),
CONSTRAINT ePosition_const CHECK (eJob.position IN ('Head', 'Manager', 'Accountant', 'Leader',
'Cashier')),
CONSTRAINT eSalary_const CHECK(eJob.salary IS NOT NULL),
CONSTRAINT eJoinDate_const CHECK(eJob.joinDate IS NOT NULL))
NESTED TABLE pPhone.mobilePhone STORE AS eMobilePhones_nested_table;

```

/

--Insert Data into Branch Table

```

insert into branchTable values (1, address_typ('Westerfield', 'Zhenghu', 'WN8 OW0'), '86-(411)478-4891');
insert into branchTable values (2, address_typ('Commercial', 'Santo Antônio do Monte', 'BD9 HG2'), '55-(243)128-8960');
insert into branchTable values (3, address_typ('Merry', 'Alexandria', 'ZP6 AI0'), '20-(591)387-2220');
insert into branchTable values (4, address_typ('Scoville', 'Buka', 'IH0 XO9'), '62-(163)129-5169');
insert into branchTable values (5, address_typ('Orin', 'Jitan', 'XI1 PB9'), '86-(976)687-6109');
insert into branchTable values (6, address_typ('Basil', 'Laval', 'ND2 IQ7'), '33-(959)617-6999');

```

```

insert into branchTable values (7, address_typ('Trailsway', 'Yashalta', 'NL5 QW0'), '7-(273)645-9765');
insert into branchTable values (8, address_typ('Magdeline', 'Sumqayıt', 'KF6 RN6'), '994-(567)573-1198');
insert into branchTable values (9, address_typ('Dryden', 'Pittsburgh', 'MB6 KV7'), '1-(412)481-3007');
insert into branchTable values (10, address_typ('Hazelcrest', 'Breu', 'VG7 BD7'), '51-(592)482-5606');
insert into branchTable values (11, address_typ('Linden', 'Dengfang', 'DG6 UT3'), '86-(593)254-5941');
insert into branchTable values (12, address_typ('Armistice', 'Bei', 'CI3 VC0'), '62-(944)990-9309');
insert into branchTable values (13, address_typ('Carioca', 'Sarmanovo', 'DO1 CX0'), '7-(330)964-9719');
insert into branchTable values (14, address_typ('Mariners Cove', 'Föglö', 'RZ4 XA0'), '358-(618)405-9497');
insert into branchTable values (15, address_typ('Corry', 'Charlemagne', 'MD1 VE5'), '1-(940)810-7302');
insert into branchTable values (16, address_typ('Hauk', 'Longjin', 'GU6 FQ8'), '86-(475)236-3250');
insert into branchTable values (17, address_typ('Northfield', 'Rey', 'ZS6 DK4'), '98-(754)579-0060');
insert into branchTable values (18, address_typ('Moland', 'Samashki', 'AG6 JX8'), '7-(317)754-6463');
insert into branchTable values (19, address_typ('Chive', 'Vereya', 'GD0 IQ8'), '7-(840)591-1520');
insert into branchTable values (20, address_typ('Portage', 'San José', 'QU8 KP4'), '506-(220)966-3351');

```

--Insert Data into Account Table

```

insert into accountTable values (101, 'savings', 6376.8, (SELECT REF(b) FROM branchTable b WHERE b.bID = 1),
0.056, 497, '14-Dec-2000');

insert into accountTable values (102, 'savings', 5798.7, (SELECT REF(b) FROM branchTable b WHERE b.bID = 17),
0.197, 599, '19-Nov-2000');

insert into accountTable values (103, 'current', 6922.42, (SELECT REF(b) FROM branchTable b WHERE b.bID = 5),
0.021, 107, '13-Apr-2004');

insert into accountTable values (104, 'savings', 4780.39, (SELECT REF(b) FROM branchTable b WHERE b.bID = 1),
0.191, 123, '12-Jul-2011');

```

```
insert into accountTable values (105, 'current', 213.12, (SELECT REF(b) FROM branchTable b WHERE b.bID = 6),
1.798, 67, '15-May-2005');

insert into accountTable values (106, 'savings', 4047.7, (SELECT REF(b) FROM branchTable b WHERE b.bID = 19),
1.748, 331, '28-Sep-2005');

insert into accountTable values (107, 'savings', 3756.8, (SELECT REF(b) FROM branchTable b WHERE b.bID = 8),
1.573, 676, '02-Mar-2001');

insert into accountTable values (108, 'current', 2336.28, (SELECT REF(b) FROM branchTable b WHERE b.bID = 2),
0.652, 694, '03-Jul-2001');

insert into accountTable values (109, 'current', 9219.34, (SELECT REF(b) FROM branchTable b WHERE b.bID = 3),
0.713, 528, '13-Oct-2006');

insert into accountTable values (110, 'savings', 7693.28, (SELECT REF(b) FROM branchTable b WHERE b.bID = 8),
0.698, 261, '30-Mar-2016');

insert into accountTable values (111, 'current', 2059.5, (SELECT REF(b) FROM branchTable b WHERE b.bID = 15),
1.94, 981, '19-Dec-2003');

insert into accountTable values (112, 'savings', 737.47, (SELECT REF(b) FROM branchTable b WHERE b.bID = 14),
0.052, 696, '10-Aug-2006');

insert into accountTable values (113, 'savings', 7126.53, (SELECT REF(b) FROM branchTable b WHERE b.bID = 18),
0.527, 157, '28-Sep-2012');

insert into accountTable values (114, 'savings', 2849.39, (SELECT REF(b) FROM branchTable b WHERE b.bID = 4),
0.907, 145, '28-Feb-2014');

insert into accountTable values (115, 'current', 7880.63, (SELECT REF(b) FROM branchTable b WHERE b.bID = 6),
1.769, 852, '08-Nov-2011');

insert into accountTable values (116, 'savings', 2178.58, (SELECT REF(b) FROM branchTable b WHERE b.bID = 2),
0.615, 112, '29-Mar-2000');

insert into accountTable values (117, 'current', 6568.43, (SELECT REF(b) FROM branchTable b WHERE b.bID = 13),
0.047, 115, '24-Jun-2011');

insert into accountTable values (119, 'current', 2878.95, (SELECT REF(b) FROM branchTable b WHERE b.bID = 11),
1.952, 927, '08-Oct-2006');
```

```
insert into accountTable values (118, 'savings', 2552.88, (SELECT REF(b) FROM branchTable b WHERE b.bID = 16),
1.901, 51, '11-May-2012');
```

```
insert into accountTable values (120, 'savings', 1109.63, (SELECT REF(b) FROM branchTable b WHERE b.bID = 7),
0.725, 854, '27-Nov-2009');
```

--Insert Data into Customer Table

```
insert into customerTable values (name_typ('Mrs', 'Marie', 'Wood'), address_typ('Clove', 'Nanlü', 'WU2 CV0'),
phone_typ('86-(973)795-1080', mobilePhones_nested('55-(311)508-6350')), 4038409, 201);
```

```
insert into customerTable values (name_typ('Dr', 'Philip', 'Reynolds'), address_typ('Harbort', 'Cluses', 'LW9
AN5'), phone_typ('33-(284)316-9482', mobilePhones_nested('46-(349)353-9587')), 1322817, 202);
```

```
insert into customerTable values (name_typ('Dr', 'Jennifer', 'Moreno'), address_typ('Everett', 'Bitung', 'GH5
OP7'), phone_typ('62-(262)613-9524', mobilePhones_nested('94-(435)933-1310')), 5209023, 203);
```

```
insert into customerTable values (name_typ('Mr', 'Dennis', 'Lynch'), address_typ('Mariners Cove', 'Nevers',
'WB1 TX2'), phone_typ('33-(815)609-5128', mobilePhones_nested('33-(967)281-4015')), 1907303, 204);
```

```
insert into customerTable values (name_typ('Dr', 'Chris', 'Martinez'), address_typ('Erie', 'Norrköping', 'RP6
CD8'), phone_typ('46-(603)719-0431', mobilePhones_nested('84-(272)854-7381')), 2649665, 205);
```

```
insert into customerTable values (name_typ('Mr', 'Ashley', 'King'), address_typ('Sherman', 'Budapest', 'ES9
PV3'), phone_typ('36-(769)694-4995', mobilePhones_nested('1-(312)576-6252')), 3548583, 206);
```

```
insert into customerTable values (name_typ('Mrs', 'Ryan', 'Price'), address_typ('Eastwood', 'Cheongju-si',
'CV0 GO3'), phone_typ('82-(399)549-5861', mobilePhones_nested('86-(849)759-1806')), 9628851, 207);
```

```
insert into customerTable values (name_typ('Dr', 'Joyce', 'Diaz'), address_typ('Farragut', 'Saint-Constant',
'OE7 NO2'), phone_typ('1-(104)381-4381', mobilePhones_nested('63-(960)562-4388')), 5890825, 208);
```

```
insert into customerTable values (name_typ('Dr', 'Michelle', 'Alexander'), address_typ('Arkansas', 'Rungkam',
'QU9 ML4'), phone_typ('62-(969)523-0508', mobilePhones_nested('33-(112)646-3314')), 8503385, 209);
```

```
insert into customerTable values (name_typ('Mr', 'Brian', 'Crawford'), address_typ('Fuller', 'A dos Cunhados',
'FI1 XR3'), phone_typ('351-(855)717-8566', mobilePhones_nested('598-(376)304-2215')), 5597672, 210);
```

```

insert into customerTable values (name_typ('Mr', 'Judith', 'Elliott'), address_typ('Talmadge', 'Ficksburg',
'KX5 BF4'), phone_typ('27-(224)823-8951', mobilePhones_nested('62-(525)253-0854')), 7707824, 211);

insert into customerTable values (name_typ('Rev', 'Diane', 'Hanson'), address_typ('Starling', 'Psary', 'QQ9
LR5'), phone_typ('48-(277)327-9773', mobilePhones_nested('7-(399)116-9795')), 8385099, 212);

insert into customerTable values (name_typ('Ms', 'Rachel', 'Ford'), address_typ('Mandrake', 'Wunat', 'VL3
ZG1'), phone_typ('62-(171)582-9914', mobilePhones_nested('381-(227)925-6646')), 9539650, 213);

insert into customerTable values (name_typ('Ms', 'Kimberly', 'Dixon'), address_typ('7th', 'Chapimarca', 'AV0
ZP7'), phone_typ('51-(902)446-2780', mobilePhones_nested('57-(756)275-3670')), 8099579, 214);

insert into customerTable values (name_typ('Ms', 'Anthony', 'Harris'), address_typ('Surrey', 'Xiaoshanzi',
'ID3 EV1'), phone_typ('86-(240)810-9443', mobilePhones_nested('234-(524)992-3227')), 2098763, 215);

insert into customerTable values (name_typ('Mr', 'Sean', 'Simpson'), address_typ('Westend', 'Paris 19', 'KG7
KU9'), phone_typ('33-(684)131-8188', mobilePhones_nested('47-(237)265-0285')), 3820509, 216);

insert into customerTable values (name_typ('Dr', 'Denise', 'Freeman'), address_typ('Hagan', 'Tunjuk Selatan',
'EH2 GJ1'), phone_typ('62-(933)766-1974', mobilePhones_nested('86-(136)436-1186')), 8056516, 217);

insert into customerTable values (name_typ('Ms', 'Joyce', 'Ruiz'), address_typ('Monica', 'Karangbayat', 'HD4
BG1'), phone_typ('62-(739)668-6560', mobilePhones_nested('57-(748)968-7518')), 6656503, 218);

insert into customerTable values (name_typ('Rev', 'Patricia', 'Perkins'), address_typ('Toban', 'Kariya', 'SW6
XZ6'), phone_typ('81-(281)932-3601', mobilePhones_nested('86-(846)462-5793')), 6035904, 219);

insert into customerTable values (name_typ('Mrs', 'Joan', 'Robertson'), address_typ('Huxley', 'Sumberbatas',
'WC0 VB6'), phone_typ('62-(424)695-9769', mobilePhones_nested('62-(186)635-7895')), 1891046, 220);

```

--Insert Data into CustomerAccount Table

```

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 205), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 101));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 215), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 101));

```

```
insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 213), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 102));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 213), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 103));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 201), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 110));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 202), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 104));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 207), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 105));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 210), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 111));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 211), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 119));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 209), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 118));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 206), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 107));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 203), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 114));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 218), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 113));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 216), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 111));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 220), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 115));

insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 204), (SELECT
REF(a) FROM accountTable a WHERE a.accNum = 109));
```



```
insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 214), (SELECT  
REF(a) FROM accountTable a WHERE a.accNum = 112));
```

```
insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 217), (SELECT  
REF(a) FROM accountTable a WHERE a.accNum = 120));
```

```
insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 208), (SELECT  
REF(a) FROM accountTable a WHERE a.accNum = 116));
```

```
insert into customerAccountTable values ((SELECT REF(c) FROM customerTable c WHERE c.custID = 208), (SELECT  
REF(a) FROM accountTable a WHERE a.accNum = 117));
```

--Insert Data into Employee Table

```
insert into employeeTable values (name_typ('Mr', 'Deborah', 'William'), address_typ('Basil', 'Glasgow', 'KZ6  
QH8'), phone_typ('66-(494)846-9327', mobilePhones_nested('48-(247)342-6376', '86-(719)169-0074')), 2474835,  
301, (SELECT REF(e) FROM employeeTable e WHERE e.empID = 309), job_typ('Manager', 52111, (SELECT REF(b) FROM  
branchTable b WHERE b.bID = 4), '13-Jan-2016'));
```

```
insert into employeeTable values (name_typ('Ms', 'Anna', 'Grant'), address_typ('Loomis', 'Sebadelhe', 'GX9  
AN0'), phone_typ('351-(499)203-9607', mobilePhones_nested('86-(719)169-0074')), 9699748, 302, (SELECT REF(e)  
FROM employeeTable e WHERE e.empID = 318), job_typ('Accountant', 58321, (SELECT REF(b) FROM branchTable b  
WHERE b.bID = 8), '26-Dec-2013'));
```

```
insert into employeeTable values (name_typ('Ms', 'Wayne', 'Jackson'), address_typ('3rd', 'Bayt Ūmmar', 'JH8  
CR7'), phone_typ('970-(774)928-0310', mobilePhones_nested('62-(206)316-4726')), 6090889, 303, (SELECT REF(e)  
FROM employeeTable e WHERE e.empID = 317), job_typ('Head', 51264, (SELECT REF(b) FROM branchTable b WHERE  
b.bID = 11), '17-Aug-2009'));
```

```
insert into employeeTable values (name_typ('Mr', 'Catherine', 'Bradley'), address_typ('Nelson', 'Tobias  
Fornier', 'NL3 LO1'), phone_typ('63-(336)673-6227', mobilePhones_nested('374-(628)941-1447')), 5673173, 304,  
(SELECT REF(e) FROM employeeTable e WHERE e.empID = 316), job_typ('Head', 30726, (SELECT REF(b) FROM  
branchTable b WHERE b.bID = 9), '21-Nov-2001'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Brandon', 'Edwards'), address_typ('Doe Crossing',  
'Glasgow', 'HB5 VH3'), phone_typ('92-(518)754-8423', mobilePhones_nested('86-(180)136-1940')), 1987198, 305,  
(SELECT REF(e) FROM employeeTable e WHERE e.empID = 303), job_typ('Manager', 31439, (SELECT REF(b) FROM  
branchTable b WHERE b.bID = 17), '13-Sep-2016'));
```

```
insert into employeeTable values (name_typ('Rev', 'Henry', 'Morrison'), address_typ('Golf Course',  
'Ngurensiti', 'OG2 DF9'), phone_typ('62-(164)417-1195', mobilePhones_nested('62-(388)423-8395')), 6065277,  
306, (SELECT REF(e) FROM employeeTable e WHERE e.empID = 302), job_typ('Leader', 50809, (SELECT REF(b) FROM  
branchTable b WHERE b.bID = 7), '17-Jun-2000'));
```

```
insert into employeeTable values (name_typ('Mr', 'John', 'Dixon'), address_typ('Valley Edge', 'Bandarlampung',  
'VU7 KH0'), phone_typ('62-(818)248-2657', mobilePhones_nested('58-(366)950-8701')), 2160673, 307, (SELECT  
REF(e) FROM employeeTable e WHERE e.empID = 301), job_typ('Manager', 23300, (SELECT REF(b) FROM branchTable b  
WHERE b.bID = 1), '21-May-2015'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Jean', 'Burns'), address_typ('Anthes', 'Pa Mok', 'HE2  
IO9'), phone_typ('66-(837)911-0680', mobilePhones_nested('374-(277)818-5806')), 5405811, 308, (SELECT REF(e)  
FROM employeeTable e WHERE e.empID = 313), job_typ('Head', 39101, (SELECT REF(b) FROM branchTable b WHERE  
b.bID = 1), '15-Dec-2014'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Willie', 'Smith'), address_typ('Fair Oaks', 'Pingpo', 'MC7  
GI9'), phone_typ('86-(991)265-7988', mobilePhones_nested('48-(731)108-8030')), 1377578, 309, (SELECT REF(e)  
FROM employeeTable e WHERE e.empID = 317), job_typ('Leader', 51307, (SELECT REF(b) FROM branchTable b WHERE  
b.bID = 6), '30-Jul-2013'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Brenda', 'Perez'), address_typ('Norway Maple', 'Rukem',  
'XL3 HP4'), phone_typ('62-(747)429-4929', mobilePhones_nested('63-(534)752-7153')), 7044732, 310, (SELECT  
REF(e) FROM employeeTable e WHERE e.empID = 320), job_typ('Head', 48788, (SELECT REF(b) FROM branchTable b  
WHERE b.bID = 8), '26-Nov-2005'));
```

```
insert into employeeTable values (name_typ('Ms', 'Patricia', 'Gonzales'), address_typ('Green', 'Mariano  
Moreno', 'MG6 QX4'), phone_typ('54-(228)155-4603', mobilePhones_nested('84-(928)223-0082')), 7795373, 311,  
(SELECT REF(e) FROM employeeTable e WHERE e.empID = 309), job_typ('Cashier', 32398, (SELECT REF(b) FROM  
branchTable b WHERE b.bID = 3), '31-Mar-2000'));
```

```
insert into employeeTable values (name_typ('Mr', 'David', 'Wood'), address_typ('Moulton', 'Manalongon', 'OQ2  
EC1'), phone_typ('63-(328)619-7980', mobilePhones_nested('86-(508)521-5726')), 1700622, 312, (SELECT REF(e)  
FROM employeeTable e WHERE e.empID = 308), job_typ('Cashier', 20425, (SELECT REF(b) FROM branchTable b WHERE  
b.bID = 13), '19-Jan-2010'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Janice', 'Howard'), address_typ('Washington', 'Glasgow',  
'ST3 ZA7'), phone_typ('48-(903)634-7572', mobilePhones_nested('51-(781)751-0589')), 4372273, 313, (SELECT  
REF(e) FROM employeeTable e WHERE e.empID = 307), job_typ('Cashier', 59101, (SELECT REF(b) FROM branchTable b  
WHERE b.bID = 16), '18-Nov-2009'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Phillip', 'Day'), address_typ('Mallard', 'Valle de
Ángeles', 'TR0 XZ9'), phone_typ('504-(631)565-2953', mobilePhones_nested('63-(537)842-0332')), 5854119, 314,
(SELECT REF(e) FROM employeeTable e WHERE e.empID = 306), job_typ('Accountant', 22253, (SELECT REF(b) FROM
branchTable b WHERE b.bID = 19), '21-Sep-2010'));
```

```
insert into employeeTable values (name_typ('Dr', 'Denise', 'Ford'), address_typ('Susan', 'Jinchang', 'ZF4
VS0'), phone_typ('86-(948)601-3873', mobilePhones_nested('961-(310)916-5787')), 1946388, 315, (SELECT REF(e)
FROM employeeTable e WHERE e.empID = 316), job_typ('Manager', 34022, (SELECT REF(b) FROM branchTable b WHERE
b.bID = 20), '11-Mar-2006'));
```

```
insert into employeeTable values (name_typ('Mrs', 'Brendon', 'Ramos'), address_typ('Mosinee', 'Ransang', 'ZI9
UC4'), phone_typ('63-(164)440-8426', mobilePhones_nested('46-(347)996-1471')), 4946417, 316, (SELECT REF(e)
FROM employeeTable e WHERE e.empID = 314), job_typ('Leader', 41355, (SELECT REF(b) FROM branchTable b WHERE
b.bID = 2), '29-Mar-2003'));
```

```
insert into employeeTable values (name_typ('Mr', 'Joyce', 'Ray'), address_typ('Pierstorff', 'Wengtian', 'NR8
HB1'), phone_typ('86-(645)715-0586', mobilePhones_nested('507-(963)715-0478')), 7408300, 317, (SELECT REF(e)
FROM employeeTable e WHERE e.empID = 301), job_typ('Accountant', 54884, (SELECT REF(b) FROM branchTable b
WHERE b.bID = 6), '13-Aug-2008'));
```

```
insert into employeeTable values (name_typ('Rev', 'Jason', 'Armstrong'), address_typ('Pearson', 'Glasgow',
'KU3 ER3'), phone_typ('62-(162)819-2961', mobilePhones_nested('502-(289)221-5794')), 4069936, 318, (SELECT
REF(e) FROM employeeTable e WHERE e.empID = 312), job_typ('Head', 20029, (SELECT REF(b) FROM branchTable b
WHERE b.bID = 1), '11-Apr-2004'));
```

```
insert into employeeTable values (name_typ('Dr', 'Jessica', 'Freeman'), address_typ('Susan', 'Mandor', 'IG1
XB6'), phone_typ('62-(804)392-6011', mobilePhones_nested('64-(641)636-9118')), 7743806, 319, (SELECT REF(e)
FROM employeeTable e WHERE e.empID = 301), job_typ('Accountant', 58144, (SELECT REF(b) FROM branchTable b
WHERE b.bID = 14), '29-Jul-2015'));
```

```
insert into employeeTable values (name_typ('Mr', 'Brenda', 'Mcdonald'), address_typ('Sutteridge', 'Masalovka',
'XZ1 UM3'), phone_typ('7-(124)542-0676', mobilePhones_nested('55-(715)873-7883')), 8936400, 320, (SELECT
REF(e) FROM employeeTable e WHERE e.empID = 310), job_typ('Cashier', 33353, (SELECT REF(b) FROM branchTable b
WHERE b.bID = 1), '04-Sep-2009'));
```

Task 4 - Database Queries

Question 4a.

Find employees with 'on' in first name who live in Glasgow:

```
SELECT
    e.pName.firstName AS "First Name",
    e.pName.surName AS "Last Name"
FROM
    employeeTable e
WHERE
    e.pAddress.city = 'Glasgow'
    AND e.pName.firstName LIKE '%on%';
```

Output:

| FIRST NAME | LAST NAME |
|------------|-----------|
| BRADON | Edwards |
| JASON | Armstrong |

Question 4b.

Find the number of savings account at each branch:

```
SELECT
    COUNT(a.accType) AS "Number of Savings Accounts",
    a.bID.bAddress.street AS "Branch Street",
    a.bID.bAddress.city AS "Branch City",
    a.bID.bAddress.p_code AS "Branch Post Code"
FROM
    accountTable a
WHERE
    a.accType = 'savings'
GROUP BY a.bID;
```

Output:

| NUMBER OF SAVINGS ACCOUNTS | BRANCH STREET | BRANCH CITY | BRANCH POST CODE |
|-------------------------------|---------------|------------------------|------------------|
| 2 | Westerfield | Zhenghu | WN8 OW0 |
| 1 | Mariners Cove | Föglö | RZ4 XA0 |
| 1 | Chive | Vereya | GD0 IQ8 |
| 1 | Scoville | Buka | IH0 XO9 |
| 1 | Trailsway | Yashalta | NL5 QW0 |
| 1 | Northfield | Rey | ZS6 DK4 |
| 1 | Commercial | Santo Antonio do Monte | BD9 HG2 |
| 1 | Moland | Samashki | AG6 JX8 |
| 2 | Magdeline | Sumqayit | KF6 RN6 |
| 1 | Hauk | Longjin | GU6 FQ8 |

Question 4g.

Find the number of employees who are supervised by Mr William, who is supervised by Mrs Smith:

```
SELECT

    CONCAT(CONCAT(e.pName.title, ' '), e.pName.surName) AS
    "Supervisor Name",

    (SELECT COUNT(e.supervisorID) FROM employeeTable e WHERE
    e.supervisorID.empID = 301) AS "Number of Employees Supervised",

    (SELECT CONCAT(CONCAT(e.supervisorID.pName.title, ' '),
    e.supervisorID.pName.surName) FROM employeeTable e WHERE
    e.supervisorID.empID = 309) AS "Supervisor's Supervisor"

FROM

    employeeTable e

WHERE

    e.empID = 301;
```

Output:

| SUPERVISOR NAME | NUMBER OF EMPLOYEES SUPERVISED | SUPERVISOR'S SUPERVISOR |
|-----------------|--------------------------------|-------------------------|
| MR WILLIAM | 3 | Mrs Smith |

Task 5 - A Critical Analysis of the Object-Relational Model Compared Against the Relational Model

Relational Model

Advantages

- Fast and simple.
- Tables made from distinguishable objects.
- Clear relationship between entities.
- Easy to backup and recover.

Disadvantages

- Hard to express data that does not translate into succinct entities easily, e.g. employees having supervisors who are also employees, requiring referencing.
- Cannot have nested relationships (needed for storing multiple entries in a particular value, e.g. mobile phones).
- Cannot write types to save on duplication of attributes, e.g. the Name type used in the new object-relational design.

Object-Relational Model

Advantages

- Combines many advantages of relation-model with that of object-orientated databases.
- Allows nested tables and other collections, e.g. 'mobilePhones_nested' table used in the design above.
- Can make use of inheritance to reduce repeated attributes (e.g. 'Person' type).
- User defined types to help reduce complexity.

Disadvantages

- Can be hard to access certain pieces of nested or referenced data (e.g. accessing a specific mobile phone of an employee who has multiple).
- Can be hard to accurately display on an ER or UML diagram due to some tables being made using types, but not having easily displayable relationships with those types.
- Can be difficult to efficiently design and build when dealing with larger databases with many entities with many defined structured types and inherited relationships.

Task 6 - Drop Statements

```
--DROP TYPES

DROP TYPE name_typ FORCE;

/

DROP TYPE address_typ FORCE;

/

DROP TYPE mobilePhones_nested FORCE;

/

DROP TYPE phone_typ FORCE;

/

DROP TYPE branch_typ FORCE;

/

DROP TYPE job_typ FORCE;

/

DROP TYPE person_typ FORCE;

/

DROP TYPE customer_typ FORCE;

/

DROP TYPE employee_typ FORCE;

/

DROP TYPE account_typ FORCE;

/

DROP TYPE customerAccount_typ FORCE;

/

--DROP TABLES

DROP TABLE branchTable PURGE;

/

DROP TABLE accountTable PURGE;

/

DROP TABLE customerAccountTable PURGE;

/

DROP TABLE customerTable PURGE;

/

DROP TABLE employeeTable PURGE;

/
```