

**COMP 3005**  
**Assignment #1**  
**Krystian Wojcicki**  
**January 28th**

**Part 1 Concepts**

1. Data A value that represents known facts with an implicit meaning
2. Mini World Some part of the real world for which the database system is developed with its data stored in the database
3. Database A collection of related data stored on a computer
4. Database System the database and the applications developed for the users on top of DBMS
5. DBA Responsible for acquiring software and hardware resources, controlling its use and monitoring efficiency of operations and authorizing access to the database, for coordinating and monitoring its use
6. End User Use the database in day to day basis. Don't know how the DB is structured
7. Data Model specifies how data is structured and operated. Consists of three parts; a set of concepts to describe the structure of the DB, a set of operations for manipulating these structures and a set of constraints that the database should obey.
8. Database Schema the description of data. Elements, data types and constraints on the database.
9. Relational Data Model all data is represented in terms of tuples (records), grouped into relations (files)
10. Atomic Value values that are not divisible
11. Domain A domain has three parts: a name, a data-type/format and a set of atomic values.
12. Key minimal set of attributes that uniquely identifies that tuples in the relation
13. Foreign Key an attribute that references a primary key of the same or different relation
14. Relation A relation contains a set of rows (tuples).
15. Attribute a column name of the relation indicating the meaning of the data items in that column.
16. Tuple a row in the relation. Data elements in each tuple represent certain facts that correspond to a real-world entity or relationship
17. Referential Integrity Rule the value in the foreign key column (or columns) of the referencing relation R1 can either be: a value of an existing primary key value of a corresponding primary key in the referenced relation R2 or null. If null the FK in R1 should not be apart of its own primary key
18. Physical Data Independence the capacity to change the internal schema without having to change the conceptual schema. The internal schema may be changed when certain file structures are reorganized or new indexes are created to improve database performance

19. Data Definition Language (DDL) a language used by the database designers and DBA to specify the conceptual schema of a database. In many DBMSs, the DDL is also used to define internal and external schemas (views).
20. Query Language (QL) a language used to specify database retrievals

## Part 2

```
create table part2 (  
    object char(10) primary key,  
    dependent char(10),  
    foreign key (dependent) references part2 (object));
```

```
insert into part2 values ('o1', null);  
insert into part2 values ('o3', 'o1');  
insert into part2 values ('o2', 'o3');  
update part2 set dependent='o2' where object='o1';
```

```
Connected to:  
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production  
  
SQL> create table part2 (  
    object char(10) primary key,  
    dependent char(10),  
    foreign key (dependent) references part2 (object));  
2      3      4  
Table created.  
  
SQL> insert into part2 values ('o1', null);  
insert into part2 values ('o3', 'o1');  
insert into part2 values ('o2', 'o3');  
update part2 set dependent='o2' where object='o1';  
  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row updated.  
  
SQL> select * from part2;  
  
OBJECT      DEPENDENT  
-----  
o1          o2  
o3          o1  
o2          o3  
  
SQL> █
```

### Part 3

**Bank**

<u>B#</u>	Name	City
B1	England	London
B2	America	New York
B3	Royal	Toronto
B4	France	Paris

**Customer**

<u>C#</u>	Name	Age	City
C1	Adams	20	London
C2	Blake	30	Paris
C3	Clark	25	Paris
C4	Your Lastname	20	Ottawa
C5	Smith	30	Toronto

**Account**

<u>C#</u>	<u>B#</u>	Balance
C1	B1	1000
C1	B2	2000
C1	B3	3000
C1	B4	4000
C2	B1	2000
C2	B2	3000
C2	B3	4000
C3	B1	3000
C3	B2	4000
C4	B1	4000
C4	B2	5000

CREATE TABLE Bank

(B# char (2),

Name char (8),

CITY char (8),

**PRIMARY KEY (B#),**

**CHECK( B# in ('B1','B2','B3','B4')),**

**CHECK(Name in ('England', 'America','Royal','France')),**

**CHECK(City in ('London', 'New York', 'Toronto', 'Paris')));**

CREATE TABLE Customer

(C# char (2),

Name char (8),

Age number (2),

City char (7),

**PRIMARY KEY (C#),**

**CHECK( C# in ('C1','C2','C3','C4','C5')),**

**CHECK (Name in ('Adams', 'Blake', 'Clark', 'Wojcicki', 'Smith')),**

**CHECK (Age between 0 and 99),**

**CHECK(City in ('London', 'Paris', 'Ottawa', 'Toronto')));**

CREATE TABLE Account

(C# char (2),

B# char (2),

Balance     number (4),  
**PRIMARY KEY (C#, B#),**  
**FOREIGN KEY (C#) REFERENCES Customer (C#) ON DELETE CASCADE,**  
**FOREIGN KEY (B#) REFERENCES Bank (B#) ON DELETE CASCADE,**  
**CHECK (Balance between 0 and 9999));**

INSERT INTO Bank VALUES('B1', 'England', 'London');  
INSERT INTO Bank VALUES('B2', 'America', 'New York');  
INSERT INTO Bank VALUES('B3', 'Royal', 'Toronto');  
INSERT INTO Bank VALUES('B4', 'France', 'Paris');  
INSERT INTO Customer VALUES('C1', 'Adams', 20, 'London');  
INSERT INTO Customer VALUES('C2', 'Blake', 30, 'Paris');  
INSERT INTO Customer VALUES('C3', 'Clark', 25, 'Paris');  
INSERT INTO Customer VALUES('C4', 'Wojcicki', 20, 'Ottawa');  
INSERT INTO Customer VALUES('C5', 'Smith', 30, 'Toronto');  
INSERT INTO Account VALUES('C1', 'B1', 1000);  
INSERT INTO Account VALUES('C1', 'B2', 2000);  
INSERT INTO Account VALUES('C1', 'B3', 3000);  
INSERT INTO Account VALUES('C1', 'B4', 4000);  
INSERT INTO Account VALUES('C2', 'B1', 2000);  
INSERT INTO Account VALUES('C2', 'B2', 3000);  
INSERT INTO Account VALUES('C2', 'B3', 4000);  
INSERT INTO Account VALUES('C3', 'B1', 3000);  
INSERT INTO Account VALUES('C3', 'B2', 4000);  
INSERT INTO Account VALUES('C4', 'B1', 4000);  
INSERT INTO Account VALUES('C4', 'B2', 5000);

```
fedora@OracleVM:~
SQL>
SQL> CREATE TABLE Bank
(B#      char (2),
 Name    char (8),
 CITY    char (8),
 PRIMARY KEY (B#),
 CHECK( B# in ('B1','B2','B3','B4')),
 CHECK(Name in ('England', 'America','Royal','France')),
 CHECK(City in ('London', 'New York', 'Toronto', 'Paris')));
      2      3      4      5      6      7      8
Table created.

SQL> CREATE TABLE Customer
(C#      char (2),
      2      3      Name    char (8),
 Age      number (2),
 City     char (7),
 PRIMARY KEY (C#),
 CHECK( C# in ('C1','C2','C3','C4','C5')),
 CHECK (Name in ('Adams', 'Blake', 'Clark', 'Wojcicki', 'Smith')),
 CHECK (Age between 0 and 99),
 CHECK(City in ('London', 'Paris', 'Ottawa', 'Toronto')));
      4      5      6      7      8      9      10
Table created.

SQL> CREATE TABLE Account
(C#      char (2),
      2      3      B#      char (2),
 Balance  number (4),
 PRIMARY KEY (C#, B#),
 FOREIGN KEY (C#) REFERENCES Customer (C#) ON DELETE CASCADE,
      4      5      6      7      FOREIGN KEY (B#) REFERENCES Bank (B#) ON DELETE CASCADE,
 CHECK (Balance between 0 and 9999));
      8
Table created.

SQL> INSERT INTO Bank VALUES('B1', 'England', 'London');
INSERT INTO Bank VALUES('B2', 'America', 'New York');

1 row created.

SQL> INSERT INTO Bank VALUES('B3', 'Royal', 'Toronto');
INSERT INTO Bank VALUES('B4', 'France', 'Paris');
INSERT INTO Customer VALUES('C1','Adams',20,'London');
INSERT INTO Customer VALUES('C2','Blake',30,'Paris');
INSERT INTO Customer VALUES('C3','Clark',25,'Paris');
INSERT INTO Customer VALUES('C4','Wojcicki',20,'Ottawa');
INSERT INTO Customer VALUES('C5','Smith',30,'Toronto');
INSERT INTO Account VALUES('C1','B1',1000);
INSERT INTO Account VALUES('C1','B2',2000);
INSERT INTO Account VALUES('C1','B3',3000);
INSERT INTO Account VALUES('C1','B4',4000);
INSERT INTO Account VALUES('C2','B1',2000);
INSERT INTO Account VALUES('C2','B2',3000);
INSERT INTO Account VALUES('C2','B3',4000);
INSERT INTO Account VALUES('C3','B1',3000);
INSERT INTO Account VALUES('C3','B2',4000);
INSERT INTO Account VALUES('C4','B1',4000);
INSERT INTO Account VALUES('C4','B2',5000);

1 row created.
```

—      □      ×

B#	NAME	CITY
---	-----	-----

```
fedora@OracleVM:~  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL>  
1 row created.  
  
SQL> select * from Bank;  
  
B# NAME      CITY  
--  
B1 England  London  
B2 America  New York  
B3 Royal    Toronto  
B4 France   Paris  
  
SQL> select * from Customer;  
  
C# NAME      AGE CITY  
--  
C1 Adams     20 London  
C2 Blake     30 Paris  
C3 Clark     25 Paris  
C4 Wojcicki  20 Ottawa  
C5 Smith     30 Toronto  
  
SQL> select * from Account;  
  
C# B#      BALANCE  
--  
C1 B1      1000  
C1 B2      2000  
C1 B3      3000  
C1 B4      4000  
C2 B1      2000  
C2 B2      3000  
C2 B3      4000  
C3 B1      3000  
C3 B2      4000  
C4 B1      4000  
C4 B2      5000  
  
11 rows selected.  
  
SQL> 
```