

COMP 3005 – Assignment 1

NAME: Nem Zutkovic

STUDENT#: 101085982

PART 1 (CONCEPTS)

| TERM | DEFINITION |
|----------------------------|---|
| Mini World | Some part of the real world for which the database system is developed with its data stored in the database. |
| Data Model | Used to hide storage details and present the users with conceptual view of the database. Specifies how data is structured and operated. Consists of concepts, operations and constraints. |
| Relational Data Model | A data model where the data is represented in terms of tuples (records), grouped into relations (files). |
| Database | A collection of related data stored on a computer. |
| Database Management System | Software to facilitate the creation and maintenance of a computerized database. |
| Database System | The database and the application programs developed on top of the DBMS (Database Management System). |
| Database Schema | Description of data at some abstraction level. Each level has its own schema. Three levels of schema include physical, conceptual and external. |
| Atomic Value | Values that are indivisible. |
| Attribute | Designates the role played by a domain in a relation. |
| Tuple | Ordered set of values (enclosed in angled brackets '<...>'), where each value is derived from an appropriate domain. |
| Domain | Consists of a name, data-type/format and set of atomic values. |
| Relation | A table of values that contains a set of rows (tuples). |
| Key | Each row has an attribute (or a set of attributes) that uniquely identifies that row in the table. Keys are underlined. |
| Primary Key | A chosen key. Primary key attributes are underlined. |
| Foreign Key | An attribute that references a primary key of the same or different relation. |
| 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 use. |
| End User | Use the database on a day-to-day basis and do not know how the DB is structured. |
| Entity Integrity Rule | The primary key attributes of each relation schema R in S cannot have null values in any tuple of r(R). |
| Logical Data Independence | The capacity to change the conceptual schema without having to change the external schemas and their associated application programs. |
| Query Language (QL) | Used to specify database retrievals. |

PART 2

```
Create table PART2(  
    entity varchar(6) primary key,  
    dependent varchar(9),  
    foreign key (dependent) references PART2(entity));
```

```
insert into PART2 values('E1',null);  
insert into PART2 values('E2',null);  
insert into PART2 values('E3','E1');
```

```
update PART2 set dependent = 'E3' where entity = 'E2';  
update PART2 set dependent = 'E2' where entity = 'E1';  
select * from PART2;
```

```
SQL> create table PART2(  
  2  entity varchar(6) primary key,  
  3  dependent varchar(9),  
  4  foreign key (dependent) references PART2(entity));
```

Table created.

```
SQL> insert into PART2 values ('E1',null);
```

1 row created.

```
SQL> insert into PART2 values ('E2',null);
```

1 row created.

```
SQL> insert into PART2 values ('E3','E1');
```

1 row created.

```
SQL> update PART2 set dependent = 'E3' where entity = 'E2';
```

1 row updated.

```
SQL> update PART2 set dependent = 'E2' where entity = 'E1';
```

1 row updated.

```
SQL> select * from PART2;
```

| ENTITY | DEPENDENT |
|--------|-----------|
| E1 | E2 |
| E2 | E3 |
| E3 | E1 |

PART 3

```
create table SAILER(  
    S# char(2) primary key,  
    Name varchar(8) default 'Lastname',  
    Age integer check(Age > 0),  
    check(Name in ('Smith','Jones','Blake','Zutkovic','Adams')));
```

```
create table BOAT(  
    B# char(2) primary key,  
    Name varchar(8) not null unique,  
    Color varchar(6),  
    check(Name in ('Freedom','Paradise','Miracle','Splendor')),  
    check(Color in ('Blue','Green','Red','Yellow')));
```

```
create table RESERVATION(  
    S# char(2),  
    B# char(2),  
    Day date not null,  
    foreign key(S#) references SAILER(S#) on delete cascade,  
    foreign key(B#) references BOAT(B#) on delete cascade);
```

```
SQL> create table SAILER(  
  2  S# char(2) primary key,  
  3  Name varchar(8) default 'Lastname',  
  4  Age integer check(Age > 0),  
  5  check(Name in ('Smith','Jones','Blake','Zutkovic','Adams')));
```

Table created.

```
SQL> create table BOAT(  
  2  B# char(2) primary key,  
  3  Name varchar(8) not null unique,  
  4  Color varchar(6),  
  5  check(Name in ('Freedom','Paradise','Miracle','Splendor')),  
  6  check(Color in ('Blue','Green','Red','Yellow')));
```

Table created.

```
SQL> create table RESERVATION(  
  2  S# char(2),  
  3  B# char(2),  
  4  Day date not null,  
  5  foreign key(S#) references SAILER(S#) on delete cascade,  
  6  foreign key(B#) references BOAT(B#) on delete cascade);
```

Table created.

```
insert into BOAT values('B1','Freedom','Blue');
insert into BOAT values('B2','Paradise','Green');
insert into BOAT values('B3','Miracle','Red');
insert into BOAT values('B4','Splendor','Yellow');
```

```
SQL> select * from BOAT;
```

| B# | NAME | COLOR |
|----|----------|--------|
| B1 | Freedom | Blue |
| B2 | Paradise | Green |
| B3 | Miracle | Red |
| B4 | Splendor | Yellow |

```
insert into SAILER values('S1','Smith','20');
insert into SAILER values('S2','Jones','30');
insert into SAILER values('S3','Blake','25');
insert into SAILER values('S4','Zutkovic','20');
insert into SAILER values('S5','Adams','30');
```

```
SQL> select * from SAILER;
```

| S# | NAME | AGE |
|----|----------|-----|
| S1 | Smith | 20 |
| S2 | Jones | 30 |
| S3 | Blake | 25 |
| S4 | Zutkovic | 20 |
| S5 | Adams | 30 |

```
insert into RESERVATION values('S1','B1','1-Jan-15');
insert into RESERVATION values('S1','B2','2-Jan-16');
insert into RESERVATION values('S1','B3','3-Feb-17');
insert into RESERVATION values('S1','B4','4-Feb-18');
insert into RESERVATION values('S2','B1','5-Mar-16');
insert into RESERVATION values('S2','B2','6-Mar-17');
insert into RESERVATION values('S2','B3','7-Apr-18');
insert into RESERVATION values('S3','B1','8-May-17');
insert into RESERVATION values('S3','B2','9-Jul-17');
insert into RESERVATION values('S4','B1','10-Sep-17');
```

```
SQL> select * from RESERVATION;
```

| S# | B# | DAY |
|----|----|-----------|
| S1 | B1 | 01-JAN-15 |
| S1 | B2 | 02-JAN-16 |
| S1 | B3 | 03-FEB-17 |
| S1 | B4 | 04-FEB-18 |
| S2 | B1 | 05-MAR-16 |
| S2 | B2 | 06-MAR-17 |
| S2 | B3 | 07-APR-18 |
| S3 | B1 | 08-MAY-17 |
| S3 | B2 | 09-JUL-17 |
| S4 | B1 | 10-SEP-17 |

10 rows selected.