COP3710 DATABASE MANAGEMENT SYSTEMS
DATABASE DESIGN EXAMPLE:
(RESTAURANT DB)

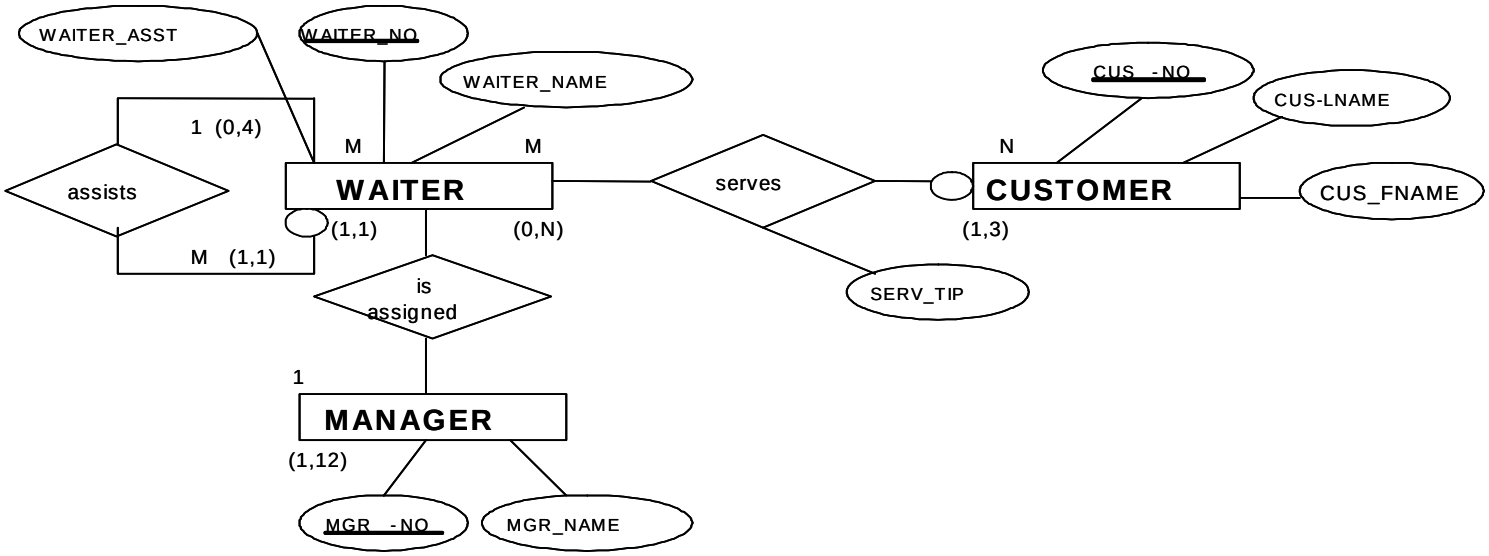
BUSINESS RULES:

Each WAITER may serve any number of CUSTOMERs or may not serve any. (WAITER_NO, WAITER_NAME)
Each CUSTOMER is served by at least one WAITER and may be served by up to 3. (CUS_NO, CUS_LNAME, CUS_FNAME)

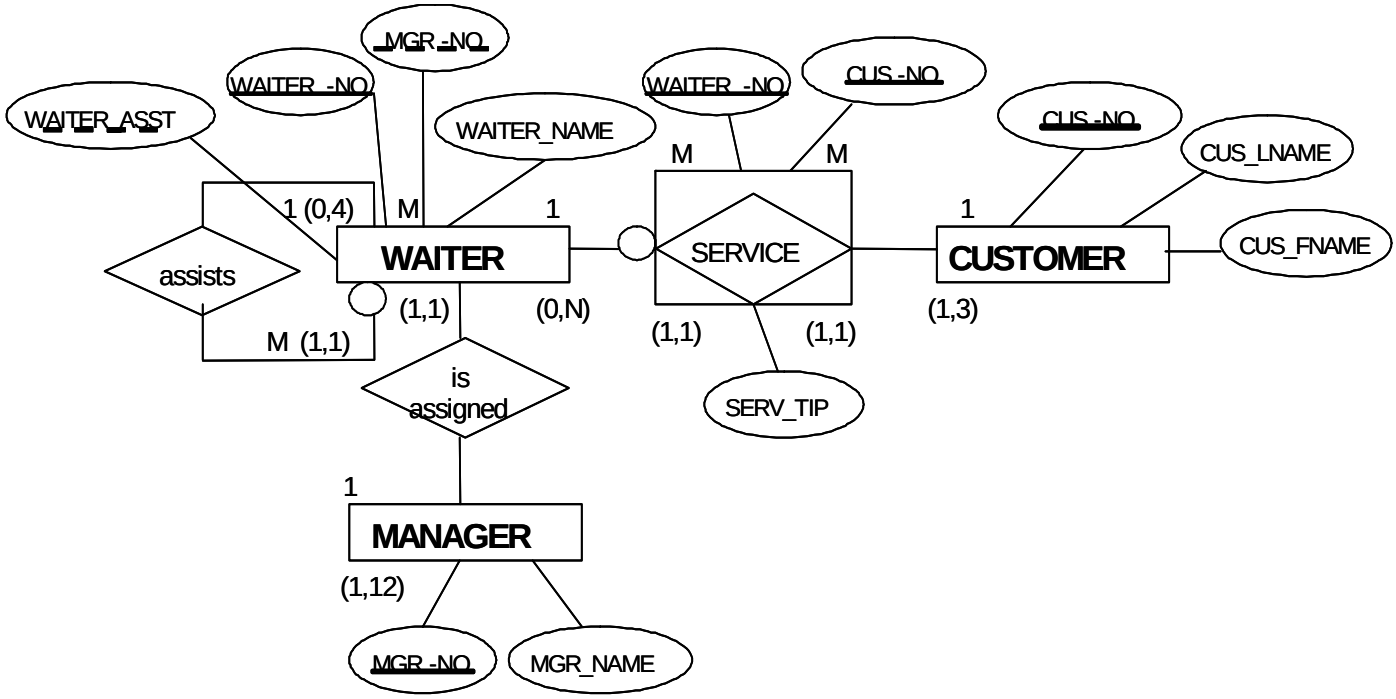
Each WAITER is assigned one and only one MANAGER. (WAITER_NO, WAITER_NAME)
Each MANAGER must have at least one WAITER assigned to him/her and may have as many as 12. (MGR_NAME, MGR_NO)

Each WAITER may assist up to 4 other WAITERs or none at all.
Each WAITER who is assisted by another WAITER is assisted by one and only one.
(WAITER_ASST)

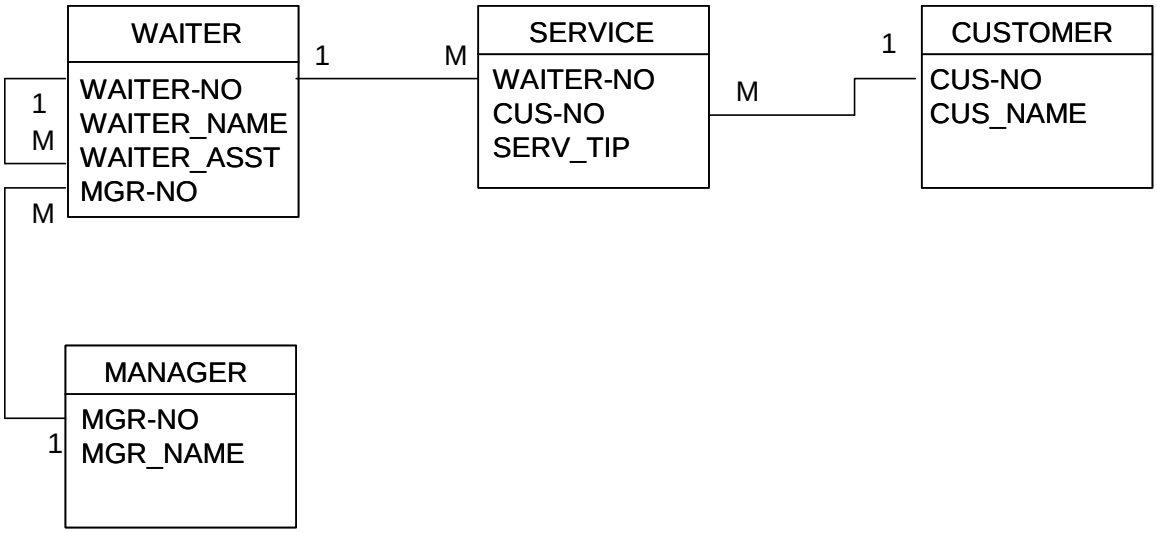
E-R LEVEL 1 DIAGRAM:



E-R LEVEL 2 DIAGRAM:



RELATIONAL SCHEMA:



RELATIONS IN RELATION NOTATION:

WAITER (WAITER-NO, WAITER-ASST, WAITER-NAME, MGR-NO)

SERVICE (WAITER-NO, CUS-NO, SERV_TIP)

CUSTOMER (CUS-NO, CUS-LNAME, CUS-FNAME)

MANAGER (MGR-NO, MGR-NAME)

RELATIONS AS TABLES:

WAITER

WAITER_NO	WAITER_NAME	WAITER_ASST	MGR_NO
155	JOHNSON	101	715
182	MATHEWS		777
101	ANDERSON	182	891
67	SMITH	95	689
81	WALLER	95	689
95	JOHNSON		591

SERVICE

WAITER_NO	CUS_NO	SERV_TIP
155	15536	10.5
95	15536	2.5
101	14586	5.81
95	18935	7.92
81	19521	12.8

CUSTOMER

CUS_NO	CUS_LNAME	CUS_FNAME
15536	SMITH	LARRY
18935	JOHNSON	BETH
19521	JOHNSON	MARY
15632	CAMBRIDGE	MARCUS
14586	JAVON	BELMAN
18325	WILLIAMS	MELVIN

MANAGER

MGR_NO	MGR_NAME
715	CLINTON
891	KERRY
777	BUSH
689	JAMES
591	JEFFERSON

ENTITY MAPS:

TABLE NAME: ___WAITER____

Column Name:	WAITER_NO	WAITER_ASST	WAITER_NAME	MGR_NO
Key Type:	PK	FK		FK
Nulls/Unique:	NN/U		NN	
FK Ref Table:		WAITER		MANAGER
FK Ref Column:		WAITER_NO		MGR_NO
Data Type:	NUMBER	NUMBER	VARCHAR2	NUMBER
Maximum Length:	3	3	25	3
Sample Data:				
(1)	155	101	JOHNSON	715
(2)	182		MATHEWS	777
(3)	101	182	ANDERSON	891
(4)	67	95	SMITH	689
(5)	81	95	WALLER	689
(6)	95		JOHNSON	591

LEGEND:

PK – Primary Key
FK – Foreign Key
NN – Not Null
U – Unique

ENTITY MAPS:

TABLE NAME: ___SERVICE____

Column Name:	WAITER_NO	CUS_NO	SERV_TIP
Key Type:	FK	FK	
Nulls/Unique:			
FK Ref Table:	WAITER	CUSTOMER	
FK Ref Column:	WAITER_NO	CUS_NO	
Data Type:	NUMBER	NUMBER	NUMBER
Maximum Length:	3	4	(5,2)
Sample Data:			
(1)	155	15536	10.50
(2)	95	15536	2.50
(3)	101	14586	5.81
(4)	95	18935	7.92
(5)	81	19521	12.80

PK

LEGEND:

PK – Primary Key
FK – Foreign Key
NN – Not Null
U – Unique

ENTITY MAPS:

TABLE NAME: ___CUSTOMER____

Column Name:	CUS_NO	CUS_LNAME	CUS_FNAME
Key Type:	PK		
Nulls/Unique:	NN/U	NN	NN
FK Ref Table:			
FK Ref Column:			
Data Type:	NUMBER	VARCHAR2	VARCHAR2
Maximum Length:	5	25	25
Sample Data:			
(1)	15536	SMITH	LARRY
(2)	18935	JOHNSON	BETH
(3)	19521	JOHNSON	MARY
(4)	15632	CAMBRIDGE	MARCUS
(5)	14586	JAVON	BELMAN

LEGEND:

PK – Primary Key
FK – Foreign Key
NN – Not Null
U – Unique

ENTITY MAPS:

TABLE NAME: ___MANAGER____

Column Name:	MGR_NO	MGR_NAME
Key Type:	PK	
Nulls/Unique:	NN/U	NN
FK Ref Table:		
FK Ref Column:		
Data Type:	NUMBER	
Maximum Length:	3	
Sample Data:		
(1)	715	CLINTON
(2)	891	KERRY
(3)	777	BUSH
(4)	689	JAMES
(5)	591	JEFFERSON

LEGEND:

PK – Primary Key
FK – Foreign Key
NN – Not Null
U – Unique

SCRIPT FILES FOR RESTAURANT DB: (Create Table Statements)

Script File Name: restaurant.sql

```
DROP TABLE MANAGER;

CREATE TABLE MANAGER (
MGR_NO              NUMBER(3) CONSTRAINT MANAGER_MGR_NO_PK PRIMARY KEY,
MGR_NAME            VARCHAR2(25) CONSTRAINT MANAGER_MNAME_NN NOT NULL)
/

DROP TABLE CUSTOMER;

CREATE TABLE CUSTOMER (
CUS_NO              NUMBER(5) CONSTRAINT CUSTOMER_CUS_NO_PK PRIMARY KEY,
CUS_LNAME           VARCHAR2(25) CONSTRAINT CUSTOMER_CUSLNAME_NN NOT NULL,
CUS_FNAME           VARCHAR2(25) CONSTRAINT CUSTOMER_CUSFNAME_NN NOT NULL)
/

DROP TABLE WAITER;

CREATE TABLE WAITER (
WAITER_NO           NUMBER(3)  CONSTRAINT WAITER_WAITER_NO_PK  PRIMARY KEY,
WAITER_NAME         VARCHAR2(25)  CONSTRAINT WAITER_WAIT_NAME_NN  NOT NULL,
WAITER_ASST         NUMBER(3),
MGR_NO              NUMBER(3),
CONSTRAINT WAITER_WAIT_ASST_FK FOREIGN KEY (WAITER_ASST) REFERENCES WAITER (WAITER_NO),
CONSTRAINT WAITER_MGR_NO_FK  FOREIGN KEY (MGR_NO) REFERENCES MANAGER (MGR_NO))
/

DROP TABLE SERVICE;

CREATE TABLE SERVICE (
WAITER_NO           NUMBER(3),
CUS_NO              NUMBER(5),
SERV_TIP            NUMBER(5,2),
CONSTRAINT SERVICE_WAITER_NO_FK FOREIGN KEY (WAITER_NO) REFERENCES WAITER (WAITER_NO),
CONSTRAINT SERVICE_CUS_NO_FK FOREIGN KEY(CUS_NO) REFERENCES CUSTOMER (CUS_NO),
CONSTRAINT SERVICE_CUSNO_WTNO_PK PRIMARY KEY (WAITER_NO,CUS_NO))
/
```

COMMENTS:

1.) DROP TABLE statement should be executed every time you perform a CREATE TABLE statement.

2.) CREATE TABLE is used to create a database object/table. It lists and defines the datatypes for all of the attributes in the table and can optionally list the constraints placed on each attribute.

3.) CONSTRAINTS can be defined at the Column Level or Table Level, except composite primary keys must be defined at the Table Level. CUSTOMER table is an example of defining Constraints at the Column Level. SERVICE table is an example of defining Constraints at the Table Level. SERVICE table is an example of doing both Column Level and Table Level.

4.) SERVICE table (above) is a Composite Entity. The CREATE TABLE SERVICE statement depicts how to define a Composite Primary Key, which is on the last line of this page.

SCRIPT FILES FOR RESTAURANT DB: (Insert Statements)

Script File Name: rest_insert..sql

```
INSERT INTO MANAGER VALUES (715,'CLINTON');
INSERT INTO MANAGER VALUES (891,'KERRY');
INSERT INTO MANAGER VALUES (777,'BUSH');
INSERT INTO MANAGER VALUES (689,'JAMES');
INSERT INTO MANAGER VALUES (591,'JEFFERSON');

INSERT INTO CUSTOMER VALUES (15536,'SMITH','LARRY');
INSERT INTO CUSTOMER VALUES (18935,'JOHNSON','BETH');
INSERT INTO CUSTOMER VALUES (19521,'JOHNSON','MARY');
INSERT INTO CUSTOMER VALUES (14586,'CAMBRIDGE','MARCUS');
INSERT INTO CUSTOMER VALUES (15632,'JAVON','BELMAN');

INSERT INTO WAITER VALUES (182,'MATHEWS',NULL,777);
INSERT INTO WAITER VALUES (101,'ANDERSON',182,891);
INSERT INTO WAITER VALUES (95,'JOHNSON',101,715);
INSERT INTO WAITER VALUES (81,'WALLER',95,689);
INSERT INTO WAITER VALUES (67,'SMITH',95,689);
INSERT INTO WAITER VALUES (155,'JOHNSON',101,715);

INSERT INTO SERVICE VALUES (155,15536,10.50);
INSERT INTO SERVICE VALUES (95,15536,2.50);
INSERT INTO SERVICE VALUES (101,14586,5.81);
INSERT INTO SERVICE VALUES (95,18935,7.92);
INSERT INTO SERVICE VALUES (81,19521,12.80);
```

COMMENTS:

1.) INSERT command is used to add rows to existing tables.

2.) Specify the data for that table in the VALUES clause, which is enclosed in parentheses.

3.) Must add one row at a time to a table.

4.) Enclose non-numeric data in single quotes. Ex. ‘JOHNSON’

5.) Use NULL keyword to omit datavalue for a column.

EXAMPLE QUERIES FOR RESTAURANT DB:

SELECT STATEMENT:

1. The sum in tips that the customer by the name of SMITH leaves her waiters.

```
SELECT SUM(SERV_TIP)
FROM SERVICE
WHERE CUS_NO =
        (SELECT CUS_NO
         FROM CUSTOMER
         WHERE CUS_LNAME = 'SMITH');
```

OUTPUT:

SUM(SERV_TIP)
13

2. The waiter names of the waiters that JONES assists.

```
SELECT WAITER_NAME
FROM WAITER
WHERE WAITER_ASST IN
        (SELECT WAITER_NO
        FROM WAITER
        WHERE WAITER_NAME = 'JOHNSON')
```

OUTPUT:

WAITER_NAME
WALLER
SMITH

3. The name of MATHEWS' manager.

```
SELECT MGR_NAME
FROM MANAGER
WHERE MGR-NO =
      (SELECT MGR-NO
      FROM WAITER
      WHERE WAITER_NAME = 'MATHEWS');
```

OUTPUT:

MGR_NAME
BUSH

4. The name of MATHEWS' manager.
   WRITE AS A JOIN WITH ALL LABLES.

```
SELECT M.MGR_NAME
FROM MANAGER M, WAITER W
WHERE M.MGR_NO = W.MGR_NO
AND W.WAITER_NAME = 'MATHEWS';
```

OUTPUT:

MGR_NAME
BUSH

5. The waiter names and their tips from the customer SMITH.

```
SELECT WAITER_NAME, SERV_TIP
FROM WAITER, SERVICE
WHERE WAITER.WAITER_NO =
        SERVICE.WAITER_NO
AND CUS_NO =
        (SELECT CUS_NO
         FROM CUSTOMER
         WHERE CUS_LNAME = 'SMITH')
```

OUTPUT:

WAITER_NAME	SERV_TIP
JOHNSON	2.50
JOHNSON	10.50

6. The waiter names and their tips from the customer SMITH.
   WRITE AS A JOIN WITH ALL LABLES.

```
SELECT W.WAITER_NAME, S.SERV_TIP
FROM WAITER W, SERVICE S, CUSTOMER C
WHERE S.CUS_NO = C.CUS_NO
AND S.WAITER_NO = W.WAITER_NO
AND C.CUS_LNAME = 'SMITH';
```