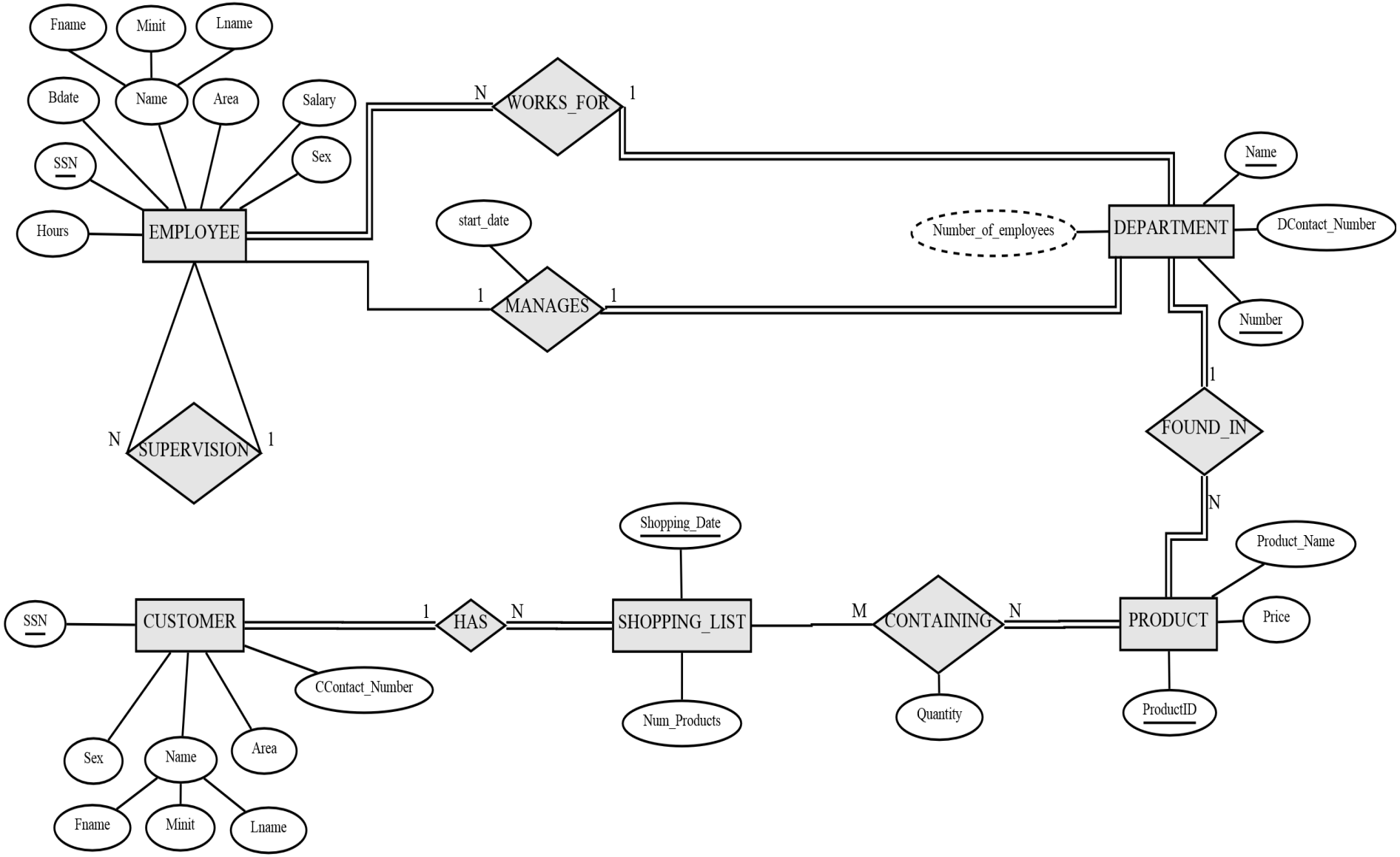
COSC344 ASSIGNMENT 2

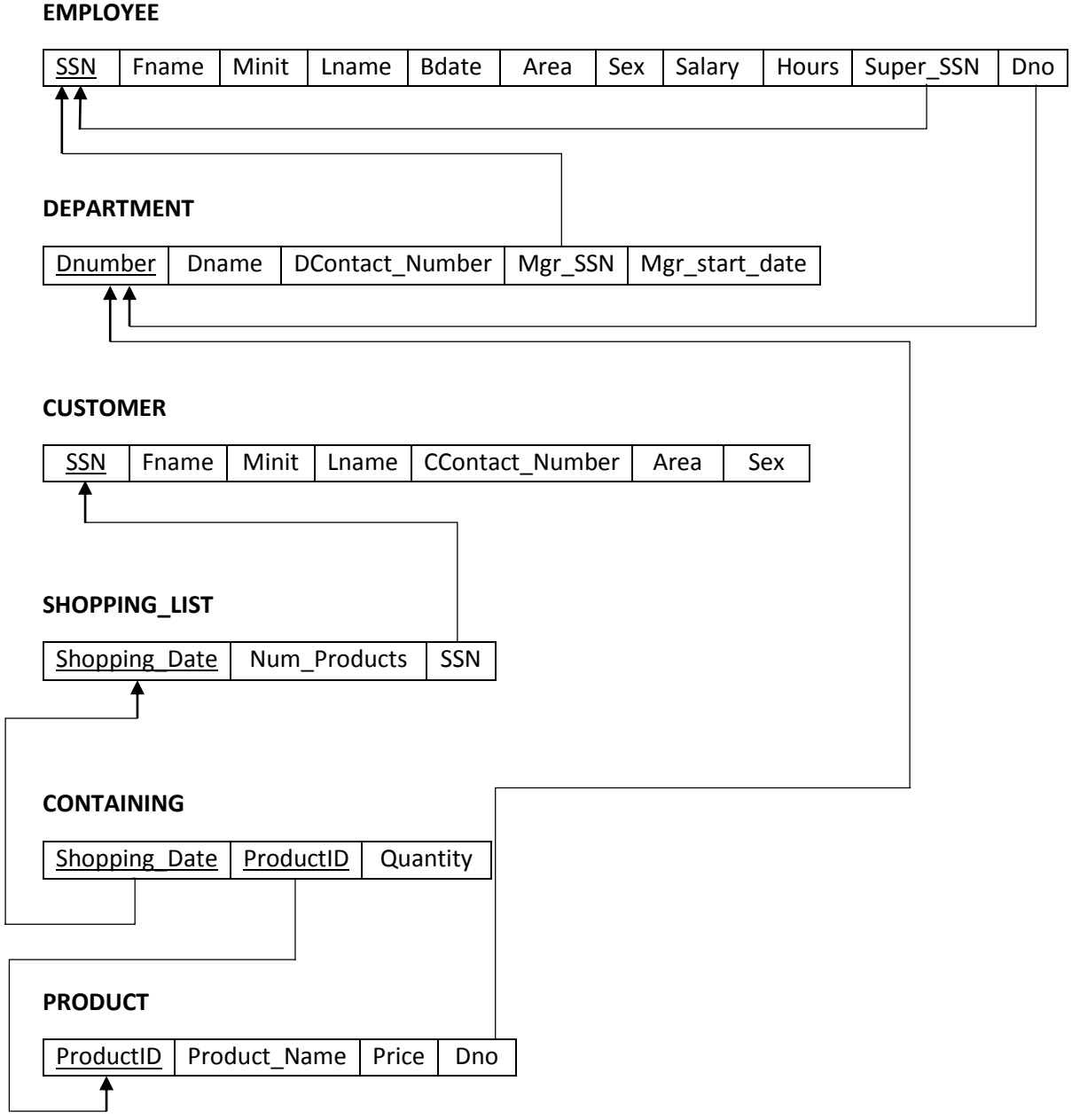
Matthew Brooker, 541670, mbrooker@cs.otago.ac.nz

AUGUST 25, 2014



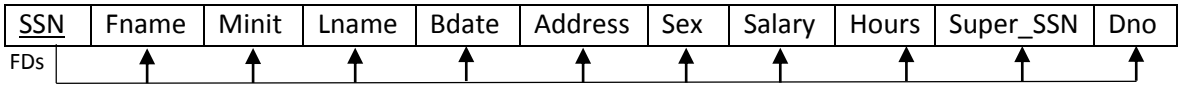
Matthew Brooker, 541670, mbrooker@cs.otago.ac.nz

Relational Schema with Integrity Constraints

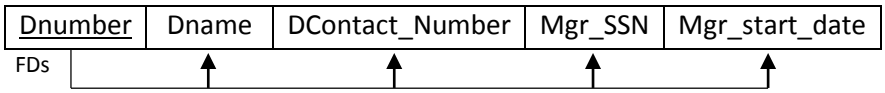


Relational Schema with Functional Dependencies

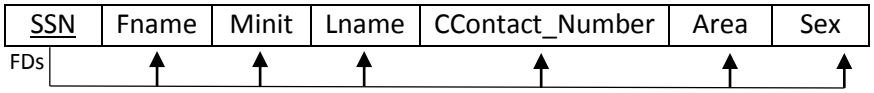
EMPLOYEE



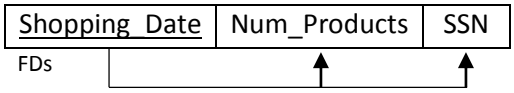
DEPARTMENT



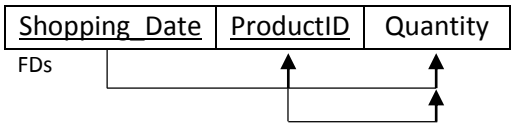
CUSTOMER



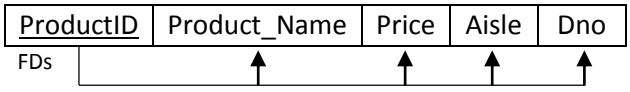
SHOPPING_LIST



CONTAINING



PRODUCT



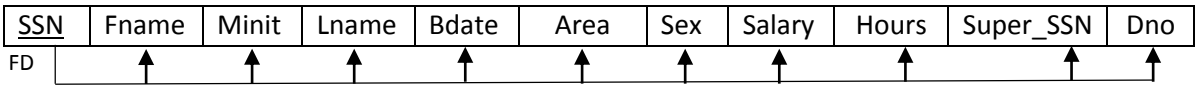
Normalisation

The tables provided in the earlier sections are already in BCNF.

The composite attribute **Name**, found in both **EMPLOYEE** and **CUSTOMER** is broken down into its three sub-attributes **Fname**, **Minit** and **Lname**. These changes make all values single and attribute domains atomic, making the design 1NF.

We can see that it must already be in 2NF because the PKs **Dnumber, SSN, Shopping_Date** and **ProductID** have every non-key attribute dependant on them for their respective relations. For example, the **SSN** is a unique identifier for each employee and therefore other employee attributes will depend on which **SSN** is being referred.

EMPLOYEE



These attributes depend on the PK, **SSN**, as indicated by the arrows.

The design is already in 3NF because for each functional dependency X -> Y in every relation R, X is a PK in R and sometimes Y is a candidate key.

The relations achieve BCNF because every left-hand side of a FD is at least a candidate key.

EMPLOYEE (SSN, Fname, Minit, Lname, Bdate, Area, Sex, Salary, Hours, Super_SSN, Dno)

X is **SSN**, the PK of **EMPLOYEE**, and Y is each of the rest of the attributes for its respective FD.
Therefore this relation is in 3NF.
The PK is on the LHS of each FD so this relation is in BCNF.

DEPARTMENT (Dnumber, Dname, DContact_Number, Mgr_SSN, Mgr_start_date)

X is **Dnumber**, the PK of **DEPARTMENT**, and Y is each of the rest of the attributes for its respective FD. Therefore this relation is in 3NF.
The PK is on the LHS of each FD so this relation is in BCNF.

CUSTOMER (SSN, Fname, Minit, Lname, CContact_Number, Area, Sex)

X is **SSN**, the PK of **CUSTOMER**, and Y is each of the rest of the attributes for its respective FD.
Therefore this relation is in 3NF.
The PK is on the LHS of each FD so this relation is in BCNF.

SHOPPING_LIST (Shopping_Date, Num_Products, SSN)

X is **Shopping_Date**, the PK of **SHOPPING_LIST**, and Y is each of the rest of the attributes for its respective FD. Therefore this relation is in 3NF.
The PK is on the LHS of each FD so this relation is in BCNF.

CONTAINING (Shopping_Date, ProductID, Quantity)

X is **Shopping_Date** and **ProductID** (candidate keys) and Y is **Quantity** and **ProductID**. Therefore this relation is in 3NF. It is in BCNF because all LHSs of the FDs are candidate keys.

PRODUCT (ProductID, Product_Name, Price, Dno)

X is **ProductID**, the PK of **PRODUCT**, and Y is each of the rest of the attributes for its respective FD. Therefore this relation is in 3NF.
The PK is on the LHS of each FD so this relation is in BCNF.

QUERIES.SQL

```
-- q01. Prints out the ssn, first name
-- and contact number of all four customers.

-- vertical subset (All rows, some columns):

SELECT ssn, fname, ccontact_number
FROM customer;

-- horizontal subset (all columns, some rows):


SELECT * FROM employee
WHERE area = 'North Dunedin';


-- q02. Joins customer and shopping list so
--      we know how many products a customer purchased
--      and when.

SELECT c.ssn, s.shopping_date, c.fname, c.lname, s.num_products
FROM customer c, shopping_list s
WHERE c.ssn = s.ssn;

-- q03. Joins customer, shopping_list and containing tables
--      to show us what products they are purchasing.

SELECT c.ssn, s.shopping_date, c.fname, c.lname, o.productid,
o.quantity
FROM customer c, shopping_list s, containing o
WHERE c.ssn = s.ssn
AND s.shopping_date = o.shopping_date;



-- q04. Prints out ssn, first name and lname of all employees
--      have salaries less than 45000.

SELECT ssn, fname, lname
FROM employee
WHERE salary != '45000';

-- q05. Prints out small table with personal
--      details of those who have short names
--      beginning with 'T'.

SELECT fname, ssn, lname, salary FROM employee
WHERE fname LIKE 'T___';

-- q06. Lists all the employees born before the
--      nineties.

SELECT ssn, fname, lname, bdate
FROM employee
WHERE bdate < TO_DATE('01-01-1990','DD-MM-YYYY');

-- q07. Shows how many employees there are in each
```

```
--      department with a salary over 10000.

SELECT dno, COUNT(*)
FROM employee
WHERE salary > 10000
GROUP BY dno;

-- q08. Same thing as before, except it excludes the
--      first two departments.


SELECT dno, COUNT(*)
FROM employee
WHERE salary > 10000
GROUP BY dno
HAVING dno > 2;

-- q09. All customer details in ascending order of ssn.

SELECT *
FROM customer
order by ssn;

-- q10. Product details of those that are higher than the
--      average product price.

SELECT *
FROM product
WHERE price >
      (SELECT AVG(price)
      FROM product);

-- q11. Lists all employees who are not managers.

SELECT fname, lname
FROM employee
WHERE NOT EXISTS
      (SELECT *
      FROM department
      WHERE ssn = mgrssn);

-- q12. Counts the number of recorded shopping lists.

SELECT COUNT(shopping_date)
FROM shopping_list;

-- q13. The price of the cheapest product in the butchery.

SELECT MIN(price)
FROM product
WHERE dno = 2;

-- q14. Deletes product with the id '1000.'

DELETE FROM product
WHERE productid = '1000';

-- q15. Changes an employees living area and department number
```

```
--      given a first name.

UPDATE employee
SET area = 'North Dunedin', dno = 3
WHERE fname = 'Pete';
```

LOAD.SQL

```
DROP TABLE containing;
DROP TABLE product;
DROP TABLE shopping_list;
DROP TABLE customer;
DROP TABLE employee cascade constraints;
DROP TABLE department cascade constraints;



CREATE TABLE department
       (dnumber         INT          PRIMARY KEY,
        dname           VARCHAR2(15) NOT NULL UNIQUE,
        dcontact_number CHAR(9)     NOT NULL UNIQUE,
        mgrssn          CHAR(9)      NOT NULL,
        mgrstartdate    DATE);


INSERT INTO department VALUES
        (1,'Produce','112392348','123456789', TO_DATE('22-05-1988','DD-
MM-YYYY'));
INSERT INTO department VALUES
        (2,'Butchery','124356779','987654321', TO_DATE('01-01-
1995','DD-MM-YYYY'));
INSERT INTO department VALUES
        (3,'Grocery','138556110','888665555', TO_DATE('19-06-1981','DD-
MM-YYYY'));
INSERT INTO department VALUES
        (4,'Chilled Foods','148224661','111100000', TO_DATE('31-12-
2004','DD-MM-YYYY'));
INSERT INTO department VALUES
        (5,'Liquor','158545766','158345766', TO_DATE('31-12-2004','DD-
MM-YYYY'));



CREATE TABLE employee
        (ssn      CHAR(9)       PRIMARY KEY,
        fname    VARCHAR2(10)  NOT NULL,
        minit    CHAR,
        lname    VARCHAR2(20)  NOT NULL,
        bdate    DATE,
        area  VARCHAR2(20),
        sex      CHAR,
```

```
        salary   NUMBER(6),
        hours    NUMBER(2),
        superssn CHAR(9)
        CONSTRAINT superssn_cnst REFERENCES employee(ssn) DISABLE,
        dno      INT           NOT NULL
        CONSTRAINT dno_cnst REFERENCES department(dnumber) DISABLE);

ALTER TABLE employee ENABLE CONSTRAINT dno_cnst;

        INSERT INTO employee VALUES
        ('123456789','Tim','L','Jones',TO_DATE('24-10-1992','DD-MM-
YYYY'),
        'Mornington','M',45000, 40, NULL, 1);
        INSERT INTO employee VALUES
        ('987688888','Rose','F','Petersond',TO_DATE('05-05-1989','DD-
MM-YYYY'),
        'Maori Hill','F', 10000, 15,'123456789', 1);
        INSERT INTO employee VALUES
        ('333445555','Earl','V','Vonstrozzenburger',TO_DATE('16-09-
1977','DD-MM-YYYY'),
        'North Dunedin','M',35000, 40,'123456789', 1);
        INSERT INTO employee VALUES
        ('987654321','Pete','G','Mcgee',TO_DATE('20-09-1965','DD-MM-
YYYY'),
        'South Dunedin','M',45000, 40, NULL,2);
        INSERT INTO employee VALUES
        ('223691415','Doug','M','Glatt',TO_DATE('03-04-1983','DD-MM-
YYYY'),
        'North Dunedin','M',35000, 40, '987654321',2);
        INSERT INTO employee VALUES
        ('239715567','Katie','S','Margaret',TO_DATE('01-07-1995','DD-
MM-YYYY'),
        'North Dunedin','F',13000, 20,'987654321', 2);
        INSERT INTO employee VALUES
        ('888665555','Tom','C','Johnson',TO_DATE('02-06-1991','DD-MM-
YYYY'),
        'Roslyn','M',45000, 40, NULL,3);
        INSERT INTO employee VALUES
        ('303012889','Jessica','B','Stevens',TO_DATE('19-11-1998','DD-
MM-YYYY'),
        'North East Valley','F', 9000, 12,'888665555', 3);
        INSERT INTO employee VALUES
        ('887722669','Sophie','R','Smith',TO_DATE('05-06-1991','DD-MM-
YYYY'),
        'Pine Hill','F', 9000, 40,'888665555',3);
        INSERT INTO employee VALUES
        ('111100000','James','F','Marshall',TO_DATE('03-07-1994','DD-
MM-YYYY'),
        'South Dunedin','M',45000, 40, NULL,4);
        INSERT INTO employee VALUES
        ('999887777','Alicia','J','Zelaya',TO_DATE('19-07-1968','DD-MM-
YYYY'),
        'Caversham','M',25000, 40, '111100000',4);
        INSERT INTO employee VALUES
        ('666884444','Ramesh','K','Narayan',TO_DATE('15-09-1962','DD-
MM-YYYY'),
        'Mornington','M', 35000, 40,'111100000',4);
        INSERT INTO employee VALUES
```

```
        ('158345766','Joyce','A','English',TO_DATE('31-07-1972','DD-MM-
YYYY'),
        'Central Dunedin','M',35000,40,NULL,5);
        INSERT INTO employee VALUES
        ('453453453','Ahmad','V','Jabbar',TO_DATE('29-03-1969','DD-MM-
YYYY'),
        'Central Dunedin','M',35000,40,'158345766',5);
        INSERT INTO employee VALUES
        ('992134455','James','E','Irnbru',TO_DATE('10-11-1937','DD-MM-
YYYY'),
        'Central Dunedin','M',35000,40,'158345766',5);

ALTER TABLE employee ENABLE CONSTRAINT superssn_cnst;


CREATE TABLE customer
        (ssn             CHAR(9) PRIMARY KEY,
         fname           VARCHAR2(10) NOT NULL,
         minit           CHAR,
         lname           VARCHAR2(20) NOT NULL,
         ccontact_number CHAR(10) NOT NULL UNIQUE,
         area            VARCHAR2(20),
         sex             CHAR);


         INSERT INTO customer VALUES
('345876567','Rupert','P','Princeton','0271815678',
                'Roslyn','M');
         INSERT INTO customer VALUES
('119982732','John','R','Bert','0275642231',
                'North Dunedin','M');
         INSERT INTO customer VALUES
('453428890','Joanne','G','Rutherford','0224351677',
                'South Dunedin','F');
         INSERT INTO customer VALUES ('556674390','Sarah','L',
'Edwards', '0221556783',
                'Mornington', 'F');


CREATE TABLE shopping_list
       (shopping_date DATE  PRIMARY KEY,
        num_products  CHAR(12),
        ssn           CHAR(9)   NOT NULL REFERENCES customer(ssn) ON
DELETE SET NULL);

        INSERT INTO shopping_list VALUES
        (TO_DATE('16-05-2014, 5:34 P.M.', 'DD-MM-YYYY, HH:MI
P.M.'),4,'345876567');

        INSERT INTO shopping_list VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI
A.M.'),8,'345876567');

        INSERT INTO shopping_list VALUES
        (TO_DATE('14-06-2014, 7:10 A.M.', 'DD-MM-YYYY, HH:MI
A.M.'),3,'119982732');

        INSERT INTO shopping_list VALUES
```

```
        (TO_DATE('2-07-2014, 12:39 P.M.', 'DD-MM-YYYY, HH:MI
A.M.'),2,'453428890');

        INSERT INTO shopping_list VALUES
        (TO_DATE('22-08-2014, 3.35 P.M.', 'DD-MM-YYYY, HH:MI
A.M.'),5,'556674390');



 CREATE TABLE product
        (productid      CHAR(4)        PRIMARY KEY,
         product_name   VARCHAR(20)    NOT NULL,
         price          NUMBER(8, 2),
         dno            INT            NOT NULL REFERENCES
department(dnumber) ON DELETE CASCADE);

        INSERT INTO product VALUES(1000, 'Flyspray', 7.50, 3);
        INSERT INTO product VALUES(1001, 'Chocolate Cake', 5.30, 3);
        INSERT INTO product VALUES(1002, 'Frozen Pizza', 8.50, 4);
        INSERT INTO product VALUES(1003, '6 Pack Beer', 14.00, 5);
        INSERT INTO product VALUES(1004, 'Avocado', 2.00, 1);
        INSERT INTO product VALUES(1005, 'Stawberries', 6.99, 1);
        INSERT INTO product VALUES(1006, 'Lettuce', 3.49, 1);
        INSERT INTO product VALUES(1007, 'Carrots 1kg', 5.99, 1);
        INSERT INTO product VALUES(1008, 'Chicken Breasts', 13.99, 2);
        INSERT INTO product VALUES(1009, 'Rump Steak', 5.99, 2);
        INSERT INTO product VALUES(1010, 'Pork Sausages 1kg', 9.99, 2);
        INSERT INTO product VALUES(1011, 'Ice Cream', 3.29, 4);
        INSERT INTO product VALUES(1012, 'Hash Browns', 8.50, 4);
        INSERT INTO product VALUES(1013, 'Noodles', 3.50, 3);
        INSERT INTO product VALUES(1014, 'Spagetti', 2.99, 3);
        INSERT INTO product VALUES(1015, 'Beef Jerkey', 4.50, 3);
        INSERT INTO product VALUES(1016, 'Tortillas', 4.99, 3);
        INSERT INTO product VALUES(1017, 'Tomato Sauce', 3.49, 3);
        INSERT INTO product VALUES(1018, 'Ajax Spray and Wipe', 4.00,
3);
        INSERT INTO product VALUES(1019, 'Nutella', 5.30, 3);
        INSERT INTO product VALUES(1020, 'Olive Oil', 4.49, 3);
        INSERT INTO product VALUES(1021, 'Butter', 5.99, 4);
        INSERT INTO product VALUES(1022, 'Yogurt', 4.99, 4);
        INSERT INTO product VALUES(1023, 'Bread', 2.99, 3);
        INSERT INTO product VALUES(1024, 'Cookies', 4.50, 3);
        INSERT INTO product VALUES(1025, 'Doughnut', 3.50, 3);
        INSERT INTO product VALUES(1026, 'Pasta Sauce', 3.00, 3);
        INSERT INTO product VALUES(1027, 'Tea Towels', 2.89, 3);
        INSERT INTO product VALUES(1028, 'Dish Cloth', 2.00, 3);
        INSERT INTO product VALUES(1029, 'Container', 5.50, 3);
        INSERT INTO product VALUES(1030, 'Coke 1.5l', 1.80, 3);



CREATE TABLE containing
       (shopping_date DATE     REFERENCES shopping_list(shopping_date)
ON DELETE CASCADE,
        productid     CHAR(4)  REFERENCES product(productid) ON DELETE
CASCADE,
        quantity      INT      NOT NULL,
```

```
        PRIMARY KEY(shopping_date, productid));

        INSERT INTO containing VALUES
        (TO_DATE('16-05-2014, 5:34 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1016, 1);

        INSERT INTO containing VALUES
        (TO_DATE('16-05-2014, 5:34 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1008, 1);

        INSERT INTO containing VALUES
        (TO_DATE('16-05-2014, 5:34 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1004, 2);


        INSERT INTO containing VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1000, 1);

        INSERT INTO containing VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1007, 1);

        INSERT INTO containing VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1017, 1);

        INSERT INTO containing VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1018, 1);

        INSERT INTO containing VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1029, 2);

        INSERT INTO containing VALUES
        (TO_DATE('20-05-2014, 10:54 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1030, 2);


        INSERT INTO containing VALUES
        (TO_DATE('14-06-2014, 7:10 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1024, 1);

        INSERT INTO containing VALUES
        (TO_DATE('14-06-2014, 7:10 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1025, 1);

        INSERT INTO containing VALUES
        (TO_DATE('14-06-2014, 7:10 A.M.', 'DD-MM-YYYY, HH:MI A.M.'),
1030, 1);


        INSERT INTO containing VALUES
        (TO_DATE('2-07-2014, 12:39 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1012, 2);
```

```
        INSERT INTO containing VALUES
        (TO_DATE('22-08-2014, 3:35 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1019, 1);

        INSERT INTO containing VALUES
        (TO_DATE('22-08-2014, 3:35 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1027, 1);

        INSERT INTO containing VALUES
        (TO_DATE('22-08-2014, 3:35 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1011, 1);

        INSERT INTO containing VALUES
        (TO_DATE('22-08-2014, 3:35 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1006, 1);

        INSERT INTO containing VALUES
        (TO_DATE('22-08-2014, 3:35 P.M.', 'DD-MM-YYYY, HH:MI P.M.'),
1003, 1);



COMMIT;
```

RESULT.LST

```
SQL> @q01.sql

SSN       FNAME      CCONTACT_N
--------- ---------- ----------
345876567 Rupert     0271815678
119982732 John       0275642231
453428890 Joanne     0224351677
556674390 Sarah      0221556783


SSN       FNAME      M LNAME                BDATE     AREA
S
--------- ---------- - -------------------- --------- -----------------
--- -
    SALARY      HOURS SUPERSSN         DNO
---------- ---------- --------- ----------
333445555 Earl       V Vonstrozzenburger    16-SEP-77 North Dunedin
M
     35000         40 123456789          1

223691415 Doug       M Glatt                03-APR-83 North Dunedin
M
     35000         40 987654321          2

239715567 Katie      S Margaret             01-JUL-95 North Dunedin
F
     13000         20 987654321          2


SQL> @q02.sql

SSN       SHOPPING_ FNAME      LNAME                NUM_PRODUCTS
--------- --------- ---------- -------------------- ------------
345876567 16-MAY-14 Rupert     Princeton            4
345876567 20-MAY-14 Rupert     Princeton            8
119982732 14-JUN-14 John       Bert                 3
453428890 02-JUL-14 Joanne     Rutherford           2
556674390 22-AUG-14 Sarah      Edwards              5

SQL> @q03.sql

SSN       SHOPPING_ FNAME      LNAME                PROD   QUANTITY
--------- --------- ---------- -------------------- ---- ----------
345876567 16-MAY-14 Rupert     Princeton            1016          1
345876567 16-MAY-14 Rupert     Princeton            1008          1
345876567 16-MAY-14 Rupert     Princeton            1004          2
345876567 20-MAY-14 Rupert     Princeton            1000          1
345876567 20-MAY-14 Rupert     Princeton            1007          1
345876567 20-MAY-14 Rupert     Princeton            1017          1
345876567 20-MAY-14 Rupert     Princeton            1018          1
345876567 20-MAY-14 Rupert     Princeton            1029          2
345876567 20-MAY-14 Rupert     Princeton            1030          2
119982732 14-JUN-14 John       Bert                 1024          1
119982732 14-JUN-14 John       Bert                 1025          1
```

```
SSN       SHOPPING_ FNAME      LNAME                PROD   QUANTITY
--------- --------- ---------- -------------------- ---- ----------
119982732 14-JUN-14 John       Bert                 1030          1
453428890 02-JUL-14 Joanne     Rutherford           1012          2
556674390 22-AUG-14 Sarah      Edwards              1019          1
556674390 22-AUG-14 Sarah      Edwards              1027          1
556674390 22-AUG-14 Sarah      Edwards              1011          1
556674390 22-AUG-14 Sarah      Edwards              1006          1
556674390 22-AUG-14 Sarah      Edwards              1003          1

18 rows selected.

SQL> @q04.sql

SSN       FNAME      LNAME
--------- ---------- --------------------
987688888 Rose       Petersond
333445555 Earl       Vonstrozzenburger
223691415 Doug       Glatt
239715567 Katie      Margaret
303012889 Jessica    Stevens
887722669 Sophie     Smith
999887777 Alicia     Zelaya
666884444 Ramesh     Narayan
158345766 Joyce      English
453453453 Ahmad      Jabbar
992134455 James      Irnbru

11 rows selected.

SQL> @q05.sql

FNAME      SSN       LNAME                     SALARY
---------- --------- -------------------- ----------
Tim        123456789 Jones                     45000
Tom        888665555 Johnson                   45000

SQL> @q06.sql

SSN       FNAME      LNAME                BDATE
--------- ---------- -------------------- ---------
987688888 Rose       Petersond            05-MAY-89
333445555 Earl       Vonstrozzenburger    16-SEP-77
987654321 Pete       Mcgee                20-SEP-65
223691415 Doug       Glatt                03-APR-83
999887777 Alicia     Zelaya               19-JUL-68
666884444 Ramesh     Narayan              15-SEP-62
158345766 Joyce      English              31-JUL-72
453453453 Ahmad      Jabbar               29-MAR-69
992134455 James      Irnbru               10-NOV-37

9 rows selected.
```

```
SQL> @q07.sql

       DNO   COUNT(*)
---------- ----------
         1          2
         2          3
         4          3
         5          3
         3          1


SQL> @q08.sql

       DNO   COUNT(*)
---------- ----------
         4          3
         5          3
         3          1

SQL> @q09.sql

SSN       FNAME      M LNAME                CCONTACT_N AREA
S
--------- ---------- - -------------------- ---------- ----------------
---- -
119982732 John       R Bert                 0275642231 North Dunedin
M
345876567 Rupert     P Princeton            0271815678 Roslyn
M
453428890 Joanne     G Rutherford           0224351677 South Dunedin
F
556674390 Sarah      L Edwards              0221556783 Mornington
F

SQL> @q10.sql

PROD PRODUCT_NAME              PRICE        DNO
---- -------------------- ---------- ----------
1000 Flyspray                   7.5          3
1002 Frozen Pizza               8.5          4
1003 6 Pack Beer                 14          5
1005 Stawberries               6.99          1
1007 Carrots 1kg               5.99          1
1008 Chicken Breasts          13.99          2
1009 Rump Steak                5.99          2
1010 Pork Sausages 1kg         9.99          2
1012 Hash Browns                8.5          4
1021 Butter                    5.99          4
1029 Container                  5.5          3

11 rows selected.

SQL> @q11.sql

FNAME      LNAME
---------- --------------------
Katie      Margaret
```

```
Doug       Glatt
Sophie     Smith
Rose       Petersond
Alicia     Zelaya
James      Irnbru
Jessica    Stevens
Ahmad      Jabbar
Earl       Vonstrozzenburger
Ramesh     Narayan

10 rows selected.

SQL> @q12.sql

COUNT(SHOPPING_DATE)
--------------------
                   5

SQL> @q13.sql

MIN(PRICE)
----------
      5.99

SQL> @q14.sql

1 row deleted.

SQL> @q15.sql

1 row updated.

SQL> spool off
```