1.

```sql
CREATE TABLE AUTHOR
(
    ANAME VARCHAR(40),
    DOB DATE,
    SEX CHAR,
    PRIMARY KEY (ANAME)
) ENGINE = INNODB;

CREATE TABLE BOOK
(
    ISBN CHAR(10),
    TITLE VARCHAR(100),
     PUBDATE  YEAR(4),
     PRIMARY KEY (ISBN)
) ENGINE = INNODB;

CREATE TABLE WROTE
(
    AUTHOR_NAME VARCHAR(40),
    BOOK_ISBN CHAR(10),
    PRIMARY KEY (AUTHOR_NAME, BOOK_ISBN),
    FOREIGN KEY (AUTHOR_NAME) REFERENCES AUTHOR(ANAME),
    FOREIGN KEY (BOOK_ISBN) REFERENCES BOOK(ISBN)
) ENGINE = INNODB;

CREATE TABLE PUBLISHER
(
    PNAME VARCHAR(40),
    ADDRESS VARCHAR(100),
    PRIMARY KEY (PNAME)
) ENGINE = INNODB;

-- multi valued attribute represented as additional table
CREATE TABLE PUB_PHONE
(
    PUB_NAME VARCHAR(40),
    PHONE CHAR(15),
    PRIMARY KEY (PUB_NAME, PHONE),
    FOREIGN KEY (PUB_NAME) REFERENCES PUBLISHER (PNAME)
) ENGINE = INNODB;

CREATE TABLE PUBLISHED
(
    PUB_NAME VARCHAR(40),
    BOOK_ISBN CHAR(10),
    PRIMARY KEY (PUB_NAME, BOOK_ISBN),
```

    FOREIGN KEY (PUB_NAME) REFERENCES PUBLISHER (PNAME)
) ENGINE = INNODB;

2.

INSERT INTO PUBLISHER VALUES ('addison wesley', 'park avenue nyc usa');
INSERT INTO PUBLISHER VALUES ('morgan kaufmann', 'south avenue san francisco usa ');

INSERT INTO PUB_PHONE VALUES ('addison wesley', '+1421232120');
INSERT INTO PUB_PHONE VALUES ('addison wesley', '+1421232121');
INSERT INTO PUB_PHONE VALUES ('morgan kaufmann', '+14231231412');

INSERT INTO BOOK VALUES ('0321228383', 'database systems', 2005);
INSERT INTO BOOK VALUES ('0321228384', 'computer science', 2007);
INSERT INTO BOOK VALUES ('0321228385', 'out of their minds', 2005);

INSERT INTO AUTHOR VALUES ('kifer', '1960-6-27', 'm');
INSERT INTO AUTHOR VALUES ('lewis', '1964-8-10', 'm');
INSERT INTO AUTHOR VALUES ('bernstein', '1956-7-28', 'm');
INSERT INTO AUTHOR VALUES ('shear', '1968-12-2', 'f');
INSERT INTO AUTHOR VALUES ('shasha', '1962-2-23', 'm');

INSERT INTO WROTE VALUES ('kifer', '0321228383');
INSERT INTO WROTE VALUES ('bernstein', '0321228383');
INSERT INTO WROTE VALUES ('lewis', '0321228383');
INSERT INTO WROTE VALUES ('shear', '0321228384');
INSERT INTO WROTE VALUES ('shear', '0321228385');
INSERT INTO WROTE VALUES ('shasha', '0321228385');

INSERT INTO PUBLISHED VALUES ('addison wesley', '0321228383');
INSERT INTO PUBLISHED VALUES ('addison wesley', '0321228384');
INSERT INTO PUBLISHED VALUES ('morgan kaufmann', '0321228385');

3.

a -
SELECT TITLE
FROM BOOK, PUBLISHED
WHERE PUB_NAME = 'addison wesley' AND ISBN = BOOK_ISBN;

```
+------------------+
| TITLE            |
+------------------+
| database systems |
| computer science |
+------------------+
```

b –
SELECT DISTINCT PP.PUB_NAME, PP.PHONE
FROM PUB_PHONE PP, PUBLISHED P, WROTE W
WHERE AUTHOR_NAME = 'KIFER'
     AND W.BOOK_ISBN = P.BOOK_ISBN
     AND P.PUB_NAME = PP.PUB_NAME;

Pub_name is optional. It indicates which publisher is associated to which phone number.
Distinct is optional: it garantees that the answer is a set (no duplicates).

```
+---------------+-------------+
| PUB_NAME      | PHONE       |
+---------------+-------------+
| addison wesley | +1421232120 |
| addison wesley | +1421232121 |
+---------------+-------------+
```

c-
SELECT AUTHOR_NAME
FROM WROTE
WHERE BOOK_ISBN = '0321228383';

```
+-------------+
| AUTHOR_NAME |
+-------------+
| bernstein   |
| kifer       |
| lewis       |
+-------------+
```

d-
SELECT W.AUTHOR_NAME
FROM WROTE W, PUBLISHED P
WHERE P.PUB_NAME = 'addison wesley'
     AND P.BOOK_ISBN = W.BOOK_ISBN;

```
+-------------+
| AUTHOR_NAME |
+-------------+
| bernstein   |
| kifer       |
| lewis       |
| shear       |
+-------------+
```

e-

```
SELECT PUB_NAME, COUNT(BOOK_ISBN)
FROM PUBLISHED, BOOK
WHERE BOOK_ISBN = ISBN AND PUBDATE = '2005'
GROUP BY PUB_NAME;
```

No need for distinct in count because (PUB_NAME, BOOK_ISBN) is a primary key.

```
+----------------+------------------+
| PUB_NAME       | COUNT(BOOK_ISBN) |
+----------------+------------------+
| addison wesley |              1 |
| morgan kaufmann |             1 |
+----------------+------------------+
```

f -

```
CREATE TABLE ISBN_COUNT
  SELECT BOOK_ISBN, COUNT(AUTHOR_NAME) AS COUNT
  FROM WROTE
  GROUP BY BOOK_ISBN;
```

```
SELECT T1.BOOK_ISBN, T1.COUNT
FROM ISBN_COUNT T1
WHERE NOT EXISTS
  (
  SELECT *
  FROM ISBN_COUNT T2
  WHERE T2.COUNT > T1.COUNT
  );
```

Using limit in the select clause is not satisfactory as there might be several books with the same number of co-authors.

```
+------------+-------+
| BOOK_ISBN  | COUNT |
+------------+-------+
| 0321228383 |    3 |
+------------+-------+
```

g-

```
SELECT AUTHOR_NAME
FROM WROTE
GROUP BY AUTHOR_NAME
HAVING COUNT(DISTINCT BOOK_ISBN) = 2;
```

```
+-------------+
| AUTHOR_NAME |
+-------------+
| shear       |
+-------------+
```

h-

```
SELECT ANAME
FROM AUTHOR
ORDER BY DOB
LIMIT 1;
```

```
+-----------+
| ANAME     |
+-----------+
| bernstein |
+-----------+
```

Another way to express this query is with a self join and a set difference (expressed with not exists) as in h.

i-

```
SELECT ANAME
FROM AUTHOR
WHERE NOT EXISTS (SELECT P.BOOK_ISBN
                  FROM PUBLISHED P, WROTE W
                  WHERE  P.PUB_NAME = 'addison wesley'
                      AND P.BOOK_ISBN = W.BOOK_ISBN
                      AND W.AUTHOR_NAME = ANAME
                  );
```

```
+--------+
| ANAME  |
+--------+
| shasha |
+--------+
```

j-

We compute age as the difference between the current date and the date of birth.

```
CREATE TABLE BOOK_AVGAGE
SELECT BOOK_ISBN, AVG(DATEDIFF(CURDATE(),DOB)/365) AS AVG_AGE
FROM AUTHOR, WROTE
WHERE ANAME = AUTHOR_NAME
GROUP BY BOOK_ISBN ;
```

```
SELECT AUTHOR_NAME
FROM WROTE W, BOOK_AVGAGE BA, AUTHOR A
WHERE  W.BOOK_ISBN = BA.BOOK_ISBN
    AND A.ANAME = W.AUTHOR_NAME
    AND DATEDIFF(CURDATE(),DOB)/365 <= AVG_AGE;
```

```
+-------------+
| AUTHOR_NAME |
+-------------+
| lewis       |
| shear       |
| shear       |
+-------------+
```

4.
There is a foreign key constraint on publisher name from published to publisher. As a consequence it is not possible to delete a tuple from publisher if the publisher name is being referenced in published. This is the case for addison wesley:

mysql> DELETE FROM PUBLISHER WHERE PNAME = 'addison wesley';
ERROR 1217 (23000): Cannot delete or update a parent row: a foreign key constraint fails

5.

There is a foreign key constraint between publisher (and pub_phone) and published.
As a consequence a brut force update of publisher or update is not possible.
We need to (1) insert pearson as a publisher, (2) update the published and pub_phone tables to reference pearson instead of addison wesley, and (3) delete addison wesley from publisher.

INSERT INTO PUBLISHER VALUES ('pearsson', 'somewhere');

UPDATE PUB_PHONE SET PUB_NAME = 'pearsson' WHERE PUB_NAME = 'addison wesley';
UPDATE PUBLISHED SET PUB_NAME = 'pearsson' WHERE PUB_NAME = 'addison wesley';

DELETE FROM PUBLISHER WHERE PNAME = 'addison wesley';

6.

We need to modify the schema:

ALTER TABLE BOOK ADD WEBSITE VARCHAR(100);

UPDATE BOOK SET WEBSITE = 'www.aw-bc.com/kifer' WHERE ISBN = '0321228383';

ISBN	TITLE	PUBDATE	WEBSITE
0321228383	database systems	2005	www.aw-bc.com/kifer
0321228384	computer science	2007	NULL
0321228385	out of their minds	2005	NULL