

CREATE TABLE fsia (

name VARCHAR(30) NOT NULL PRIMARY KEY,

capital FLOAT NOT NULL);

CREATE TABLE fsib (

name VARCHAR(30) NOT NULL PRIMARY KEY,

price FLOAT NOT NULL,

outstanding integer NOT NULL);

INSERT INTO fsia(name, capital) values('aaaaaa', 10000);

INSERT INTO fsia(name, capital) values('bbbbbb', 20000);

INSERT INTO fsia(name, capital) values('cccccc', 30000);

INSERT INTO fsib(name, price, outstanding) values('dddddd', 10000, 4);

INSERT INTO fsib(name, price, outstanding) values('eeeeee', 10000, 5);

INSERT INTO fsib(name, price, outstanding) values('ffffff', 10000, 6);

--

SELECT b.name as name, b.outstanding \* b.price as capital

FROM [internetstore].[dbo].[fsib] as b

UNION

SELECT a.name, a.capital

FROM [internetstore].[dbo].[fsia] as a;



CREATE TABLE customers (

id INTEGER PRIMARY KEY NOT NULL,

name VARCHAR(50),

balance DECIMAL(10,2));

CREATE TABLE orders (

id INTEGER PRIMARY KEY NOT NULL,

customerId INTEGER NOT NULL REFERENCES customers(id),

product VARCHAR(100));

INSERT INTO customers(id, name, balance) VALUES(1, 'ACME', 500);

INSERT INTO orders(id, customerId, product) VALUES(1, 1, 'Diamond blade');

INSERT INTO customers(id, name, balance) VALUES(2, 'Doolittle Co.', -100);

INSERT INTO orders(id, customerId, product) VALUES(2, 2, 'Vacuum cleaner');

--

delete orders

where customerId IN (SELECT id FROM customers where balance < 0)



CREATE TABLE employees (

id INTEGER NOT NULL PRIMARY KEY,

name VARCHAR(30) NOT NULL);

CREATE TABLE sales (

employeeId INTEGER NOT NULL REFERENCES employees(id),

value INTEGER NOT NULL CHECK(value > 0));

INSERT INTO employees(id, name) VALUES(1, 'John');

INSERT INTO employees(id, name) VALUES(2, 'Mark');

INSERT INTO employees(id, name) VALUES(3, 'Jane');

INSERT INTO sales(employeeId, value) VALUES(1, 100);

INSERT INTO sales(employeeId, value) VALUES(3, 300);

--

SELECT name

FROM employees

where id not in (select employeeId from sales)



CREATE TABLE customers (

id INTEGER NOT NULL PRIMARY KEY,

name VARCHAR(30) NOT NULL);

CREATE TABLE transactions (

id INTEGER NOT NULL PRIMARY KEY,

customerId INTEGER REFERENCES customers(id),

amount DECIMAL(15,2) NOT NULL);

INSERT INTO customers(id, name) VALUES(1, 'Steve');

INSERT INTO customers(id, name) VALUES(2, 'Jeff');

INSERT INTO transactions(id, customerId, amount) VALUES(1, 1, 100);

--

select c.name as name, (SELECT count(customerId)

FROM transactions as t

WHERE c.id = t.customerId) AS transactions

from customers as c