Exercise 4.57.

Define a rule that says that person 1 can replace person 2 if either person 1 does the same job as person 2 or someone who does person 1's job can also do person 2's job, and if person 1 and person 2 are not the same person. Using your rule, give queries that find the following:

a. all people who can replace Cy D. Fect;

b. all people who can replace someone who is being paid more than they are, together with the two salaries.

Answer.

The `replace` rule below expresses the sufficient condition for replacing individuals:

```
(rule (replace ?person-1 ?person-2)
      (or (and (job ?person-1 ?position)
               (job ?person-2 ?position)
               (not (same ?person-1 ?person-2)))
          (and (job ?person-1 ?position-1)
               (job ?person-2 ?position-2)
               (can-do-job ?position-1 ?position-2)
               (not (same ?person-1 ?person-2)))))
```

Using this rule, we can find out all people who can replace Cy D. Fect with a simple query:

```
;;; Query input:
(replace ?person (Fect Cy D))

;;; Query results:
(replace (hacker alyssa p) (fect cy d))
(replace (bitdiddle ben) (fect cy d))
```

To find out all people who are qualified to someone else's job while bare lower salaries, we could consult the data base with the following query:

```
;;; Query input:
(and (replace ?person-1 ?person-2)
     (salary ?person-1 ?amount-1)
     (salary ?person-2 ?amount-2)
     (lisp-value < ?amount-1 ?amount-2))

;;; Query results:
(and (replace (aull dewitt) (warbucks oliver))
     (salary (aull dewitt) 25000)
     (salary (warbucks oliver) 150000)
     (lisp-value < 25000 150000))

(and (replace (fect cy d) (hacker alyssa p))
     (salary (fect cy d) 35000)
     (salary (hacker alyssa p) 40000)
     (lisp-value < 35000 40000))
```


Email address: informlarry@gmail.com