

## Aufgabe 2

. (a)

i.

```
1 SELECT Shipment.sno
2 FROM   Shipment, Part
3 WHERE  Shipment.pno = Part.pno AND
4        Part.price > 100
```

ii.

```
1 SELECT Shipment.sno
2 FROM   Shipment, Project
3 WHERE  Shipment.pno = Part.pno AND
4        Shipment.jno = Project.jno AND
5        Project.jloc = 'Philadelphia'
```

(b)

i. Ein vereinfachen des Joins würde zum Kreuzprodukt führen, ebenso würde ein Abändern der Selektion zu einem anderen Ergebnis führen.  
Damit ist keine Vereinfachung möglich.

ii. Ein Projekt ist einem eindeutigen Part zugeschrieben, wogegen ein Part mehreren Projekt zugewiesen sein kann.

Eine gleich resultierende Vereinfachung ist demnach:

```
1 SELECT Shipment.sno
2 FROM   Shipment, Project
3 WHERE  Shipment.jno = Project.jno AND
4        Project.jloc = 'Philadelphia'
```

wobei nachfolgende Abfrage zusätzlich Tupel liefern würde, die dem richtigen Part, aber einem falschen Projekt zugewiesen sind:

```
1 SELECT Shipment.sno
2 FROM   Shipment, Project
3 WHERE  Shipment.pno = Part.pno AND
4        Project.jloc = 'Philadelphia'
```

. (a)

i. nested

Subquery		Result
Supply.pnum = Parts.pnum	COUNT(shipdate)	pnum
3	2	10
10	1	8
8	0	

ii. unnested

Subquery		Result
Supply.pnum	COUNT(shipdate) AS ct	pnum
3	2	10
10	1	

(b)

i.

```

1  SELECT          pnum
2  FROM            Parts
3  LEFT OUTER JOIN (SELECT pnum, COUNT(shipdate) AS ct
4                    FROM    Supply
5                    WHERE   shipdate < 1-1-80
6                    GROUP BY pnum) T ON (Parts.pnum = T.pnum)
7  WHERE           (Parts.qoh = 0 AND
8                    T.count IS NULL) OR
9                    Parts.qoh = T.ct;
```

Tupel zu denen das GROUP subquery kein Ergebnis liefert werden mit dem LEFT OUTER JOIN mit NULL in den Join aufgenommen. Die WHERE clause nimmt Selektiert diese Ergebnisse zusätzlich zu der bisherigen Selektion.