	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	4			
	STiNE-Gruppe 11	Kobras, Pöhlmann, Tsiamis			
	Ausgabe	Mi. 25.11.2015	Abgabe	Fr. 11.12.2015	

1 Relationenalgebra

1.1 Teilaufgabe a

$$\pi_{\text{Sorte}}(\sigma_{\text{Vorname}="Horst"}(\pi_{\text{PNR}, \text{Vorname}}(\text{Person}))) \bowtie_{\text{Person.PNR}=\text{Obst.Entdecker}} \pi_{\text{Sorte}, \text{Entdecker}}(\text{Obst}))$$

1.2 Teilaufgabe b

$$\pi_{\text{Vorname}, \text{Nachname}}(\pi_{\text{PNR}, \text{Vorname}, \text{Nachname}}(\text{Person})) \bowtie_{\text{Person.PNR}=\text{Allergie.Person}} \sigma_{\text{Symptom}="Halskratzen"}(\pi_{\text{Person}, \text{Symptom}}(\text{Allergie})))$$

1.3 Teilaufgabe c

$$\pi_{\text{Obst.Sort}, \text{Person.Nachname}}(\pi_{\text{Sorte}, \text{Entdecker}}(\text{Obst})) \bowtie_{\text{Obst.Entdecker}=\text{Person.PNR}} \pi_{\text{PNR}, \text{Nachname}}(\text{Person}) \bowtie_{\text{Person.PNR}=\text{Allergie.Person}} \pi_{\text{Person}}(\sigma_{\text{Symptom}="Würgereiz"}(\pi_{\text{Person}, \text{Symptom}}(\text{Allergie}))))$$

2 Schemadefinition

CREATE TABLE 'Person'

('Vorname' VARCHAR(30) **PRIMARY KEY NOT NULL**,

'Nachname' VARCHAR(30) **PRIMARY KEY NOT NULL**,

'Geb.Datum' DATE **CHECK**('Geb.Datum' < CURRENT_DATE OR 'Geb.Datum' IS NULL),

'Wohnort' VARCHAR(30) **NOT NULL**);

CREATE TABLE 'Vorstand'

('VID' INT **PRIMARY KEY NOT NULL**,

'Sitz' VARCHAR(30) **UNIQUE NOT NULL**);

CREATE TABLE 'Verein'

('Name' VARCHAR(30) **PRIMARY KEY NOT NULL**,

'Gründungsjahr' DATE);

CREATE TABLE 'ist in'

('Person Vorname' VARCHAR(30) **PRIMARY KEY NOT NULL**,

'Person Nachname' VARCHAR(30) **PRIMARY KEY NOT NULL**,

'Vorstand VID' VARCHAR(30) **PRIMARY KEY NOT NULL**,


'Amt' VARCHAR(30) **NOT NULL**,

'von' DATE **NOT NULL**,

'bis' DATE,

CONSTRAINT 'ist in Person Vorname'

FOREIGN KEY('Person Vorname') **REFERENCES** 'Person' ('Vorname'),

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	4			
	STiNE-Gruppe 11	Kobras, Pöhlmann, Tsiamis			
	Ausgabe	Mi. 25.11.2015	Abgabe	Fr. 11.12.2015	

CONSTRAINT 'ist in Person Nachname'
FOREIGN KEY('Person Nachname') **REFERENCES** 'Person' ('Nachname'),
CONSTRAINT 'ist in Vorstand VID'
FOREIGN KEY('Vorstand VID') **REFERENCES** 'Vorstand' ('VID');

ALTER TABLE 'Vorstand'
ADD 'Verein Name' VARCHAR(30) **NOT NULL**,
ADD CONSTRAINT 'von Verein' **FOREIGN KEY** ('Verein Name') **REFERENCES** 'Verein' ('Name');

ALTER TABLE 'Verein'
ADD 'Person Vorname' VARCHAR(30) **NOT NULL**,
ADD CONSTRAINT 'trainiert Person Vorname' **FOREIGN KEY** ('Person Vorname') **REFERENCES** 'Person' ('Vorname'),
ADD 'Person Nachname' VARCHAR(30) **NOT NULL**,
ADD CONSTRAINT 'trainiert Person Nachname' **FOREIGN KEY** ('Person Nachname') **REFERENCES** 'Person' ('Nachname');

ALTER TABLE 'Person'
ADD 'Verein Name' VARCHAR(30) **NOT NULL**,
ADD CONSTRAINT 'Fan von Verein' **FOREIGN KEY** ('Verein Name') **REFERENCES** 'Verein' ('Name');


CREATE TABLE 'spielt für'
 'Person Vorname' VARCHAR(30) **PRIMARY KEY NOT NULL**,
 'Person Nachname' VARCHAR(30) **PRIMARY KEY NOT NULL**,
 'Verein Name' VARCHAR(30) **PRIMARY KEY NOT NULL**,
 ('Position' VARCHAR(30) **NOT NULL**,
CONSTRAINT 'spielt für Person Vorname' **FOREIGN KEY** ('Person Vorname') **REFERENCES** 'Person' ('Vorname'),
CONSTRAINT 'spielt für Person Nachname' **FOREIGN KEY** ('Person Nachname') **REFERENCES** 'Person' ('Nachname'),
CONSTRAINT 'spielt für Verein Name' **FOREIGN KEY** ('Verein Name') **REFERENCES** 'Verein' ('Name'));

3 Datenmanipulation mit SQL

3.1 Teilaufgabe a

3.1.1 i

SELECT DISTINCT RS.Name, **COUNT**(RF) **AS** Anzahl
FROM Rennstall RS, RennfahrerIn RF
WHERE RF.Rennstall = RS.RSID;

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	4			
	STiNE-Gruppe 11	Kobras, Pöhlmann, Tsiamis			
	Ausgabe	Mi. 25.11.2015	Abgabe	Fr. 11.12.2015	

3.1.2 ii

```
SELECT RF.Vorname, RF.Nachname
FROM RennfahrerIn RF
WHERE NOT IN
(SELECT RF2.Vorname, RF2.Nachname
FROM RennfahrerIn RF2, Platzierung PL
WHERE RF2.RID = PL.RID);
```

3.1.3 iii

```
SELECT RF.Nachname, MIN(PL.Platz) AS BestePlatzierung
FROM RennfahrerIn RF, Platzierung PL
WHERE RF.RID = PL.RID
ORDER BY PL.Platz;
```

3.1.4 iv


```
SELECT RF.Nachname, MIN(PL.Platz) AS BestePlatzierung
FROM RennfahrerIn RF, Platzierung PL
ORDER BY RF.Nachname;
```

3.1.5 v

```
SELECT *
FROM RennfahrerIn RF
WHERE RF.Geburt BETWEEN 1980-01-01 AND 1985-01-01
AND RF.Nachname = _a%;
```

3.1.6 vi

```
SELECT RS.TeamchefIn
FROM Rennstall RS, RennfahrerIn RF
WHERE RS.RSID = RF.Rennstall
AND RF.RID IN
(SELECT RF2.RID
FROM RennfahrerIn RF2, Platzierung PL
WHERE RF2.RID = PL.RID
AND PL.OID IN
(SELECT PL2.OID
```

	Lehrveranstaltung	Grundlagen von Datenbanken WS 2015/16		
	Aufgabenzettel	4		
	STiNE-Gruppe 11	Kobras, Pöhlmann, Tsiamis		
	Ausgabe	Mi. 25.11.2015	Abgabe	Fr. 11.12.2015

FROM Platzierung PL2
WHERE COUNT(PL2.OID)> 5));

3.2 Teilaufgabe b

3.2.1 i

UPDATE RennfahrerIn RF
SET RF.Wohnort = "Aston Clinton (United Kingdom)"
WHERE RF.RID = "8";

3.2.2 ii

UPDATE Rennstall RS
SET RS.Budget = RS.Budget + 50
WHERE EXISTS
(SELECT *
FROM RennfahrerIn RF, Platzierung PL
WHERE RS.RSID = RF.Rennstall
AND RF.RID = PL.RID
GROUP BY RS.RSID
HAVING AVG(PL.Platz) < 5);


3.2.3 iii

INSERT INTO Rennort (Name, Strecke)
VALUES ('Deutschland GP', 'Hockenheimring');

Dies wird so nicht funktionieren, da der Primärschlüssel, welcher nicht "NULL" sein darf, fehlt und somit "NULL" ist.

3.2.4 iv

INSERT INTO Platzierung
VALUES (SELECT PL.RID
FROM Platzierung PL, Rennort RO
WHERE RO.Name = "Brasilien GP"
AND RO.OID = PL.OID
AND PL.Platz = 1,
SELECT RO.OID
FROM Rennort RO

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	4			
	STiNE-Gruppe 11	Kobras, Pöhlmann, Tsiamis			
	Ausgabe	Mi. 25.11.2015	Abgabe	Fr. 11.12.2015	

WHERE RO.Name = "Deutschland GP",
1);

Dies wird so nicht funktionieren, da der "Deutschland GP" nicht richtig initialisiert wurde.

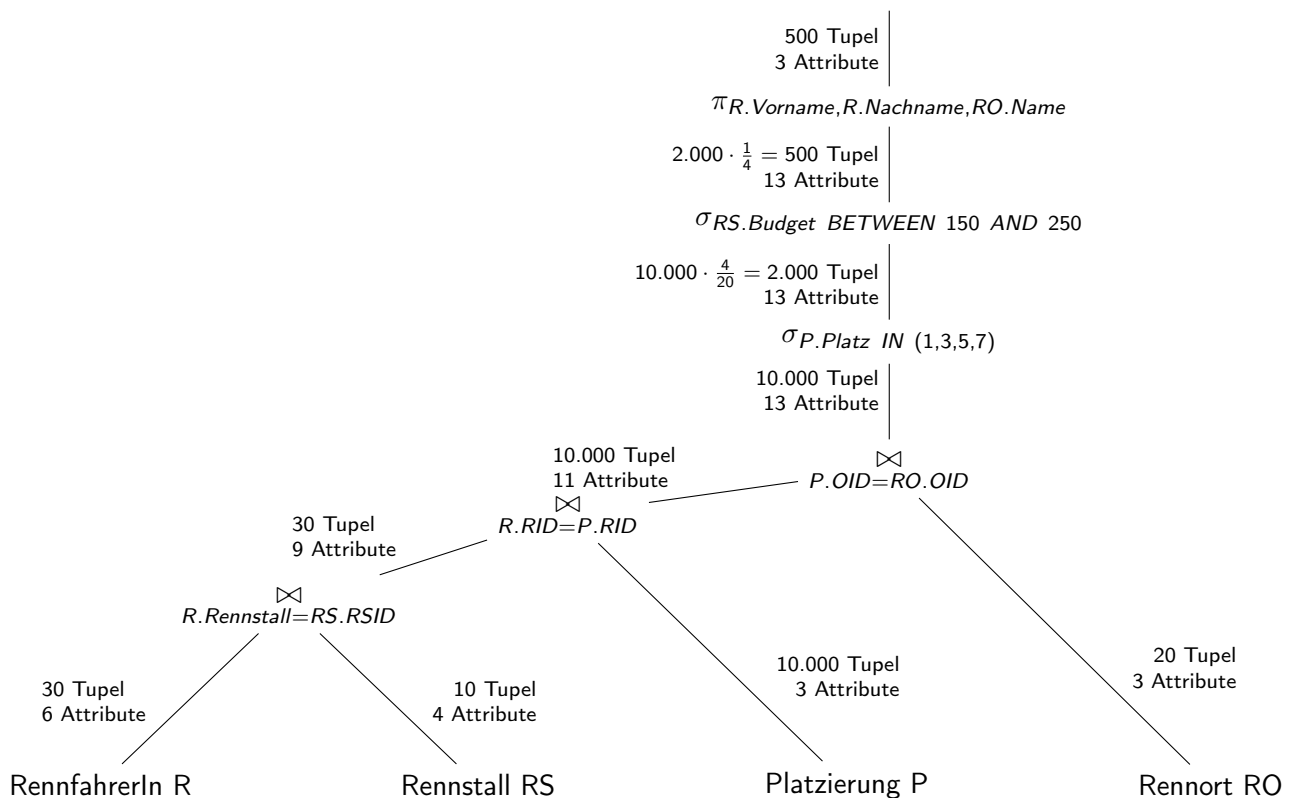
3.2.5 v


```

INSERT INTO Platzierung
VALUES (SELECT RF.RID
FROM RennfahrerIn RF
WHERE RF.Vorname = "Felipe"
AND RF.Nachname = "Massa",
SELECT RO.OID
FROM Rennort RO
WHERE Ro.Name = "Grobritannien GP",
5);

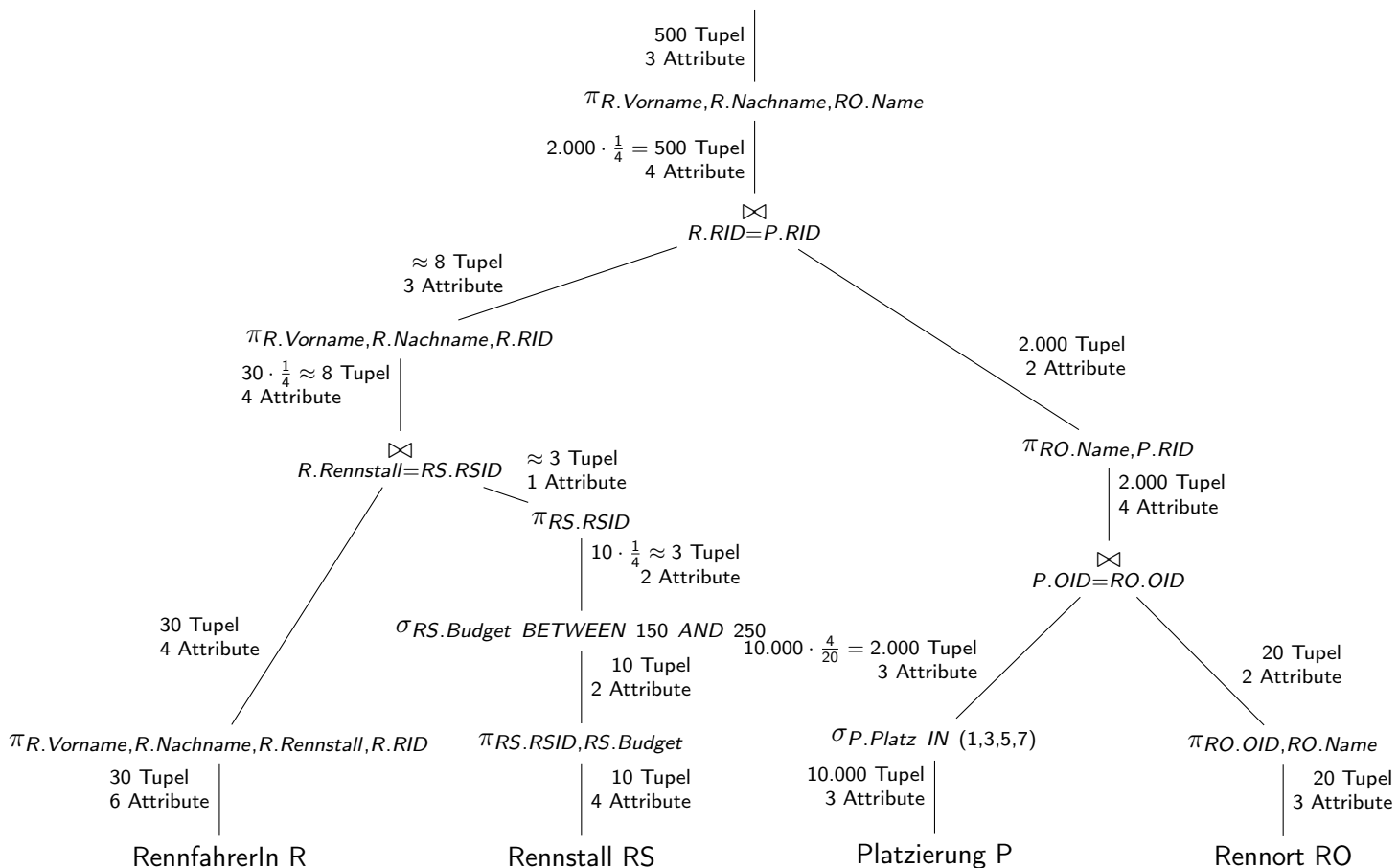
```

4 Anfrageoptimierung



	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	4			
	STiNE-Gruppe 11	Kobras, Pöhlmann, Tsiamis			
	Ausgabe	Mi. 25.11.2015	Abgabe	Fr. 11.12.2015	

Der optimierte Baum:



Bewertung Suchbaum 1(der obere):

Er ist übersichtlich. Allerdings unbalanciert. Und er rechnet mit sehr groSSen Tupeln. Und er rechnet durchgängig mit groSSen Zahlen.

Bewertung von Suchbaum 2 (der untere):

Er ist unübersichtlich. Allerdings balancierter. Und er rechnet mit kleinen Tupeln im Vergleich zum ersten. Und er rechnet mit kleineren Zahlen als der erste.