Database System Principles

Test One

Class 20122113306 No 20122112266 Name 王鹤祥

1. (2 x 3 points) Given a table Employees and some SQL queries on it, why are these queries wrong?

Employees(employee-id, employee-name, *company-id*, employee-city, age, salary)

It is assumed that each employee has an unique id and name.

(1)
```
create table Employees
  ( employee-id   char(20),
    employee-name char(20),
    company-id    char(20),
    employee-city char(20),
    age           integer,
    salary        integer,
    primary key (employee-id), primary key (employee-name), ×
    check (age >0)
  )
```
改为 primary key (employee-id, employee-name).

(2)
```
select employeeid, sum(salary)
from Employees
group by company-id ×
having avg(salary)>1000
```
改为 group by employee-id

2. (6 points) 给出下列关系代数操作对应的 **SQL** 语句

(1) $\sigma_p(r)$ (2) $\Pi_{A1, A2, ... Am}(r)$

(3) $r \infty s$,
, 假设 r(A, B, C), s(C, E, F)

Answers:

(1) select * from r where P

(2) select $A_1, A_2, \cdots A_m$ from r

(3) select * from r natual join s
select * from r
where r.C = s.C

3. (6 points) 给出下列 **SQL** 语句对应的关系代数表达式

(1) **select** *branch-name*, **max** *(salary)*
from *pt-works*
group by *branch-name*
假设 pt-works(employee-name, branch-name, salary)

(2) **insert into** *r*
select $A_1, A_2, \ldots, A_m$
from $r_1, r_2, .., r_n$
where P

(3) **update** *loan*
set *amount* = **amount** *1.2
where *amount* > 1000

Answers:

(1) branch_name $\mathcal{G}$ max(salary) (pt_works)

(2) $r \leftarrow r \cup \Pi_{A_1, A_2 \cdots A_m}(\sigma_P(r_1, r_2, \cdots, r_n))$

(3) $T_1 \leftarrow \Pi$ loan_number, amount ×1.2 (σ amount > 1000 (loan))
$T_2 \leftarrow \sigma$ amount ≤ 1000 (loan)
loan $\leftarrow T_1 \cup T_2$

4. (10 points)Given two tables *branch* (*branch-name, branch-city, assets*) and *Account*(*account-number, branch-name, balance*) as follows.

– **dbo. branch** 摘要 对象资

branch-name	branch-city	assets
Brighton	Brooklyn	7100000
Downtown	Brooklyn	9000000
Mianus	Horseneck	400000
North Town	Rye	3700000
Perryridge	Horseneck	1700000
Pownal	Bennington	300000
Redwood	Palo Alto	2100000
Round Hill	Horseneck	8000000
NULL	*NULL*	*NULL*

– **dbo. account** 摘要 对象资源

account-number	branch-name	balance
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750
A-222	Redwood	700
A-305	Round Hill	350
NULL	*NULL*	*NULL*

(1) If the table *account* is defined as :

create table *account*

(*account_number* **char**(10),

branch_name **char**(15),

balance **integer**,

primary key (*account_number*),

foreign key (*branch_name*) **references** *branch*)

whether or not the following SQL statements are permitted to be executed, and why?

(i) Update ***account***
set ***branch-name***='Haidian'
where ***account-number***='A-101'

(ii) delete
from *branch*
where *branch-name*='Pownal'

Answers:

(i) 不能执行. 不满足外键约束.
account 必须满足 branch 的主键有此branch-name.

(ii) 能执行.
account 中无此branch-name, 被参照表可正常删除.

(2) If the table *account* is defined as :

create table *account*
(*account_number* **char**(10),
branch_name **char**(15),
balance **integer**,
primary key (*account_number*),
foreign key (*branch_name*) **references** *branch*
on delete cascade 级联外键.
on update cascade
)

whether or not the following SQL statements are permitted to be executed, and why?

(iii) Update ***branch***
set ***branch-name***='Haidian'
where ***branch-name***='Brighton'

(iv) Update ***account***
set ***branch-name***='Haidian'
where ***account-number***='A-101'

Answers:

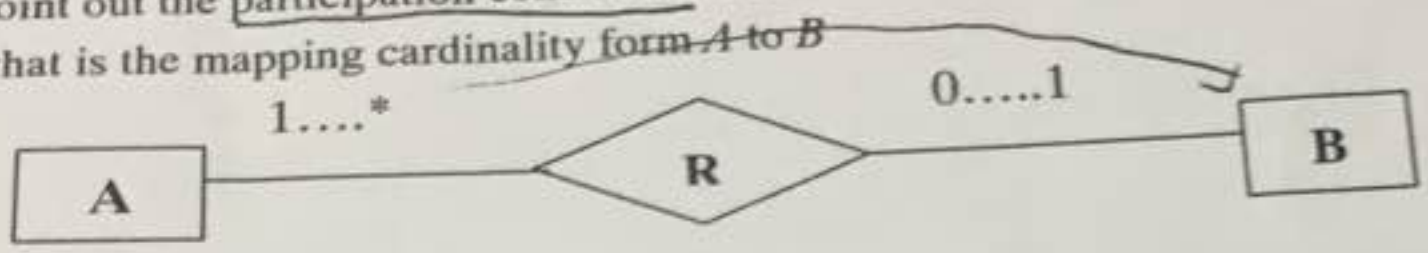
(iii). 能执行.
级联外键中被参照表更新时, 参照表account也会随之更新

(iv). 不能执行.
级联外键中参照表更新为被参照表中没有的值时, 不

5. **(4 points)** For the entity sets **A** and **B** and the relationship set **R** among them in the following figure,

(1) point out the participation constraints of **A** and **B** in **R**

(2) what is the mapping cardinality form *A to B*



Answers:

(1) 每个A可以通过关系R对应1到多个B;
每个B通过关系R只能对应1个或不对应某A.

(2) ~~many to one~~ one to many.

6. (6 points) Reduce entity set *customer* into two relational tables:

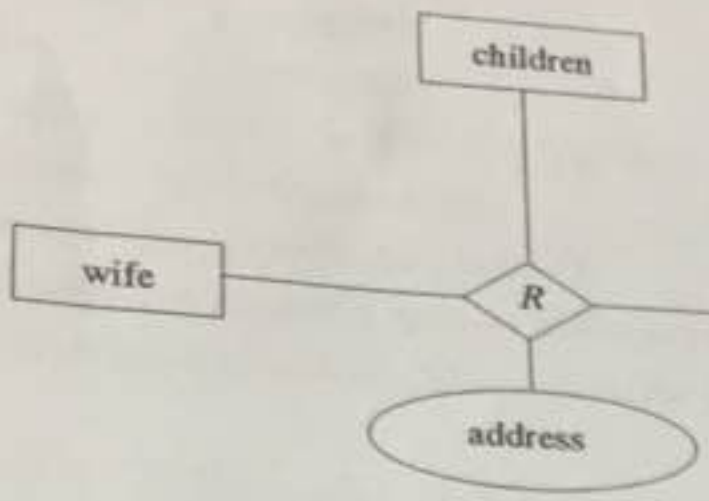
c-id	*c-name*	*d-names*	*street*	*city*
321-12-2	John	{Hayes, Adams}	North	Rye
322-10-4	Smith	{Mary, Elice}	Spring	Princeton

Answers:

c-id	d-name
321-12-2	Hayes
321-12-2	Adams
322-10-4	Mary
322-10-4	Elice

c-id	c-name	street	city
321-12-2	John	North	Rye
322-10-4	Smith	Spring	Princeton

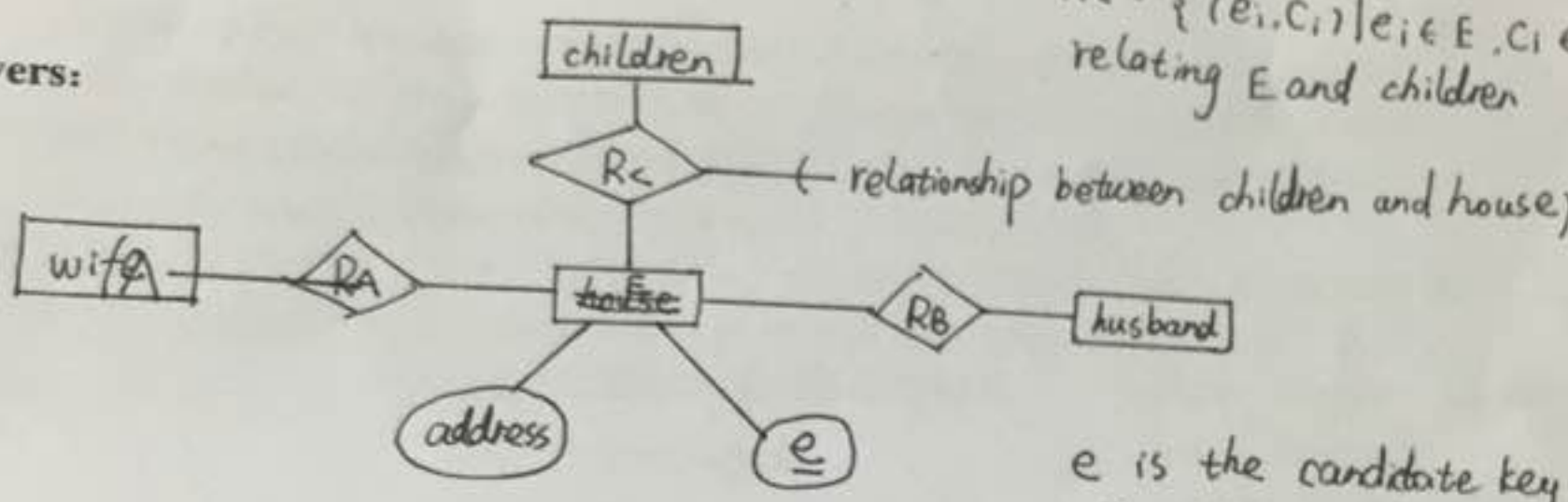
7. (6 points) **Convert the following E-R diagram into a diagram that contains only binary relationships, and give the definitions of the entities and relationships in this diagram**



$E = \{e_i\}$. $E = |R|$.
$E = \{e_i\}$. $R = \{(A_i, B_i, C_i)\}$
$R_A = \{(e_i, A_i) | e_i \in E, A_i \in Wife\}$.
relating E and wife,

$R_B = \{(e_i, B_i) | e_i \in E, B_i \in husband\}$
relating E and husband)
$R_C = \{(e_i, C_i) | e_i \in E, C_i \in children\}$
relating E and children

Answers:



e is the candidate key to E.
all attributes for R. address is assigned to E

8. (20 points) Consider the following relations in an enterprise database, where the primary keys are underlined.

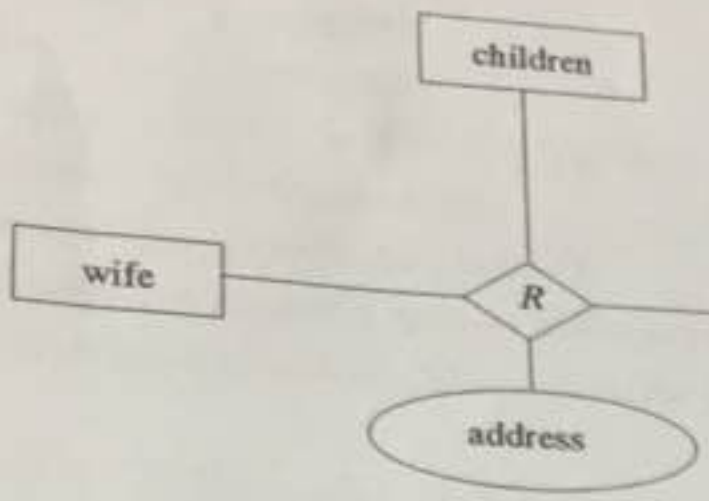
企业

Employee(employeeID, employeename, age, address, sex, salary, deptID)
Department(deptID, deptname, managerID, managername)
DepartLocations(deptID, deptlocation)
Project (projectID, projectname, projectlocation, deptID)
Workson(employeeID, projectID, hours)
Dependent(employeeID, dependentname, sex, Birthdate, Relationship)

1) (5 points) Use a SQL statement to define the relational table *Employee*, in which {*employeeID*} is the primary key, and {*employee-name*} is the candidate key and is not permitted to be null; there also exists the referential integrity between the table *Employee* and *Department*. It is also required that the value of an employee's salary is between 2000 and 10000.

Answer:

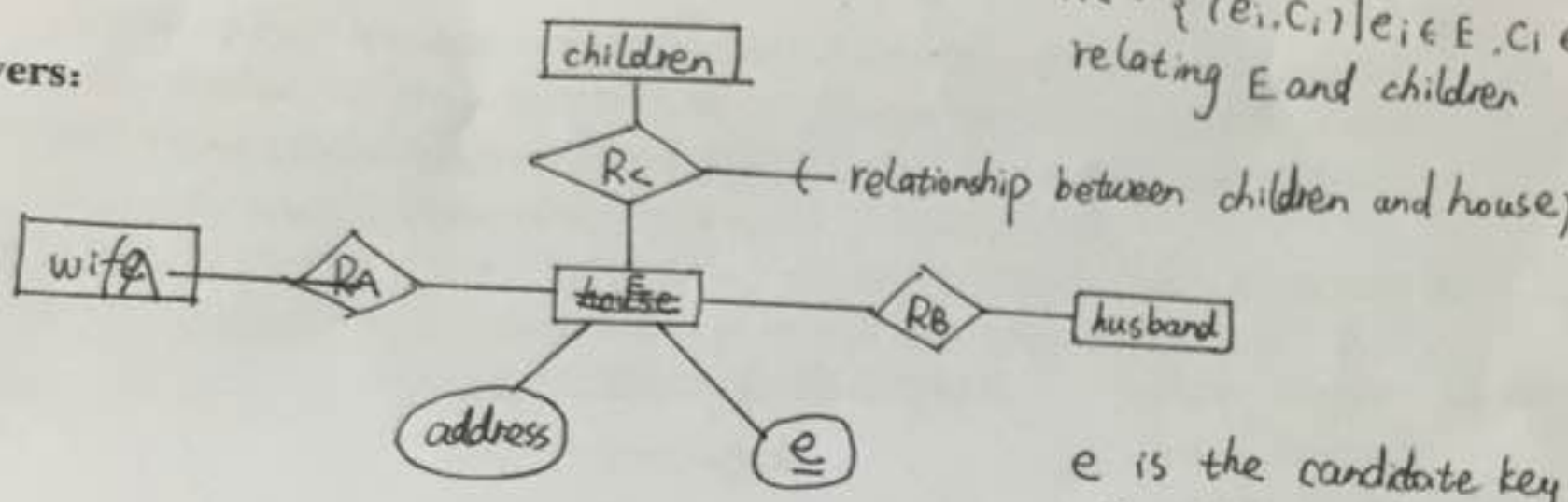
create table *employee*
(employeeID integer,



$E = \{e_i\}$. $E = |R|$.
$E = \{e_i\}$. $R = \{(A_i, B_i, C_i)\}$
$R_A = \{(e_i, A_i) | e_i \in E, A_i \in Wife\}$.
relating E and wife,

$R_B = \{(e_i, B_i) | e_i \in E, B_i \in husband\}$
relating E and husband)
$R_C = \{(e_i, C_i) | e_i \in E, C_i \in children\}$
relating E and children

Answers:



e is the candidate key to E.
all attributes for R. address is assigned to E

8. (20 points) Consider the following relations in an enterprise database, where the primary keys are underlined.

企业

Employee(employeeID, employeename, age, address, sex, salary, deptID)
Department(deptID, deptname, managerID, managername)
DepartLocations(deptID, deptlocation)
Project (projectID, projectname, projectlocation, deptID)
Workson(employeeID, projectID, hours)
Dependent(employeeID, dependentname, sex, Birthdate, Relationship)

1) (5 points) Use a SQL statement to define the relational table *Employee*, in which {*employeeID*} is the primary key, and {*employee-name*} is the candidate key and is not permitted to be null; there also exists the referential integrity between the table *Employee* and *Department*. It is also required that the value of an employee's salary is between 2000 and 10000.

Answer:

create table *employee*
(employeeID integer,

whether or not the following SQL statements are permitted to be executed, and why?

(i) Update ***account***
set ***branch-name***='Haidian'
where ***account-number***='A-101'

(ii) delete
from *branch*
where *branch-name*='Pownal'

Answers:

(i) 不能执行. 不满足外键约束.
account 必须满足 branch 的主键有此branch-name.

(ii) 能执行.
account 中无此branch-name, 被参照表可正常删除.

(2) If the table *account* is defined as :

create table *account*
(*account_number* **char**(10),
branch_name **char**(15),
balance **integer**,
primary key (*account_number*),
foreign key (*branch_name*) **references** *branch*
on delete cascade 级联外键.
on update cascade
)

whether or not the following SQL statements are permitted to be executed, and why?

(iii) Update ***branch***
set ***branch-name***='Haidian'
where ***branch-name***='Brighton'

(iv) Update ***account***
set ***branch-name***='Haidian'
where ***account-number***='A-101'

Answers:

(iii). 能执行.
级联外键中被参照表更新时, 参照表account也会随之更新

(iv). 不能执行.
级联外键中参照表更新为被参照表中没有的值时, 不