<epam>

Relationships – Part 2

Relational Databases Basics

TRAINING
CENTER

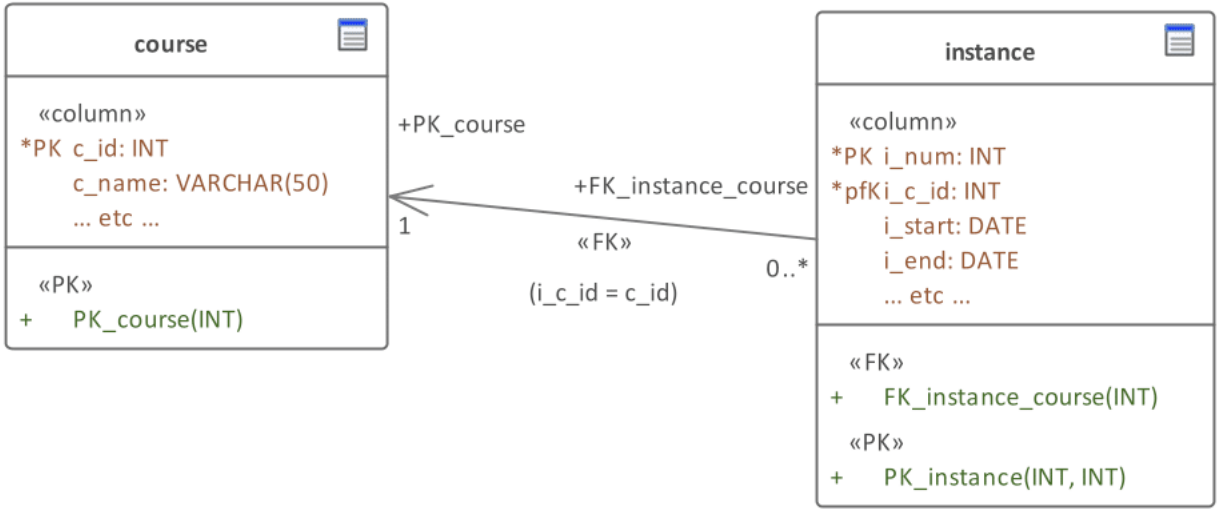
<epam>

Read and remember!

Identifying relationship – a record in a child relation has to reference a record in a parent relation.

Non-identifying relationship – a record in a child relation may exist without referencing a record in a parent relation.

Identifying relationship



Identifying relationship

course

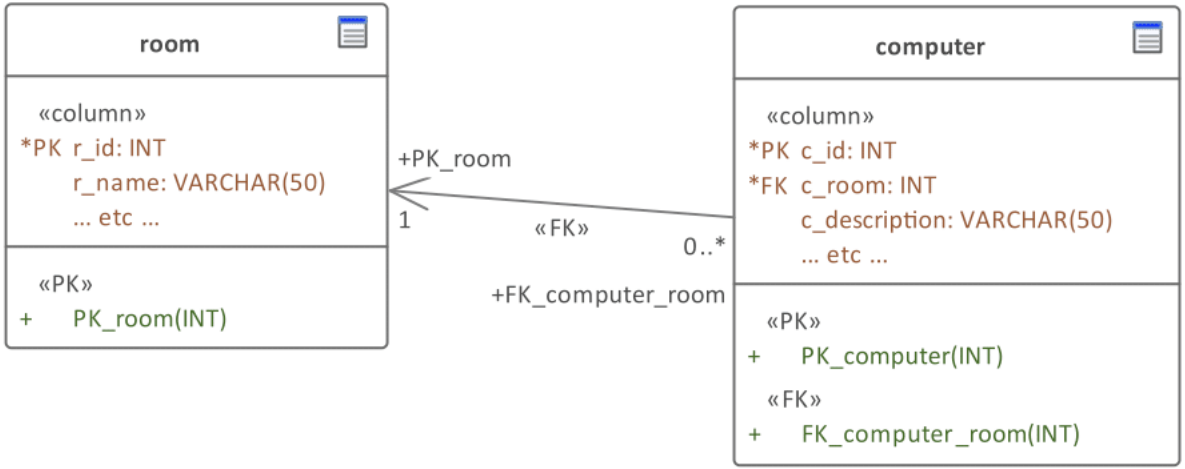
c_id	c_name	…
1	Java	
2	.NET	
3	C#	
4	Python	

instance

i_num	i_c_id	i_start	i_end	…
1	1	2017-01-10	2017-05-10	
2	1	2017-07-20	2017-10-18	
3	1	2018-03-05	2018-06-09	
1	2	2017-01-02	2017-03-04	
2	2	2017-08-17	2017-12-31	
1	3	2017-01-01	2030-12-31	

A course instance may not exist without referencing some course.

Non-identifying relationship



Non-identifying relationship

room

r_id	n_name	…
1	Room-213	
2	Room-216	
3	Room-210	
4	Dean's office	

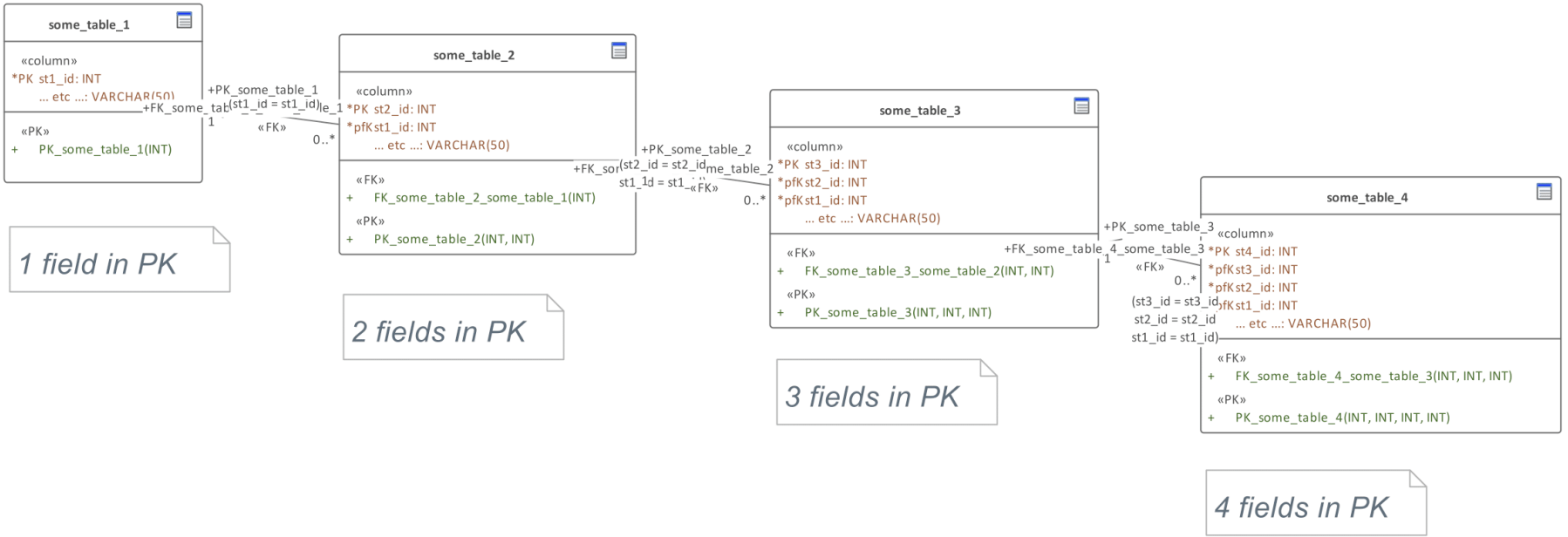
computer

c_id	c_room	c_description	…
1	1	Computer-1	
2	1	Computer-2	
3	1	Computer-3	
4	2	Computer-4	
5	2	Computer-5	
6	*NULL*	Computer-6	

A computer may either belong to a room, or not.

Identifying and non-identifying relationship approaches

In identifying relationship the FK (scientifically) has to be a part of the PK of the child relation.



Identifying and non-identifying relationship approaches

In identifying relationship the FK (scientifically) has to be a part of the PK of the child relation.

type

t_id	t_name
1	Furniture
2	Dishes

category

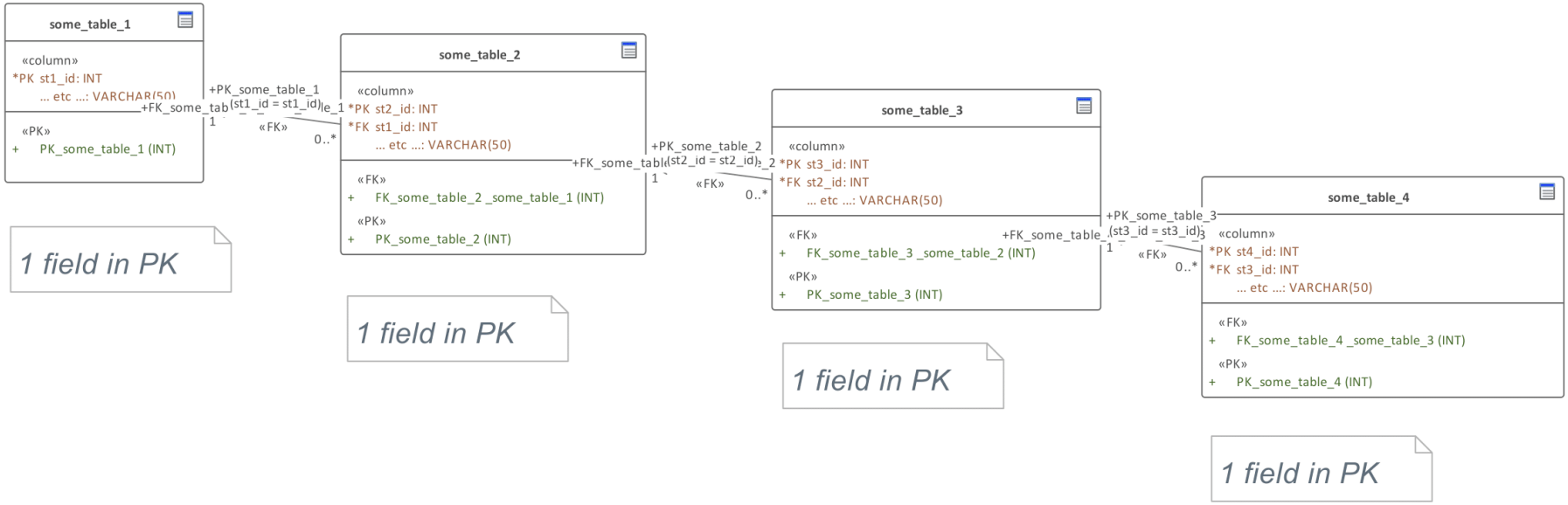
c_id	t_id	c_name
1	1	Home
2	1	Office
1	2	Glass

item

i_id	c_id	t_id	i_name
1	1	1	Cabinet
2	1	1	Bed
3	1	1	Sofa
1	2	1	Table
2	2	1	Table
1	1	2	Cup

Identifying and non-identifying relationship approaches

In non-identifying relationship the FK (scientifically) has NOT to be a part of the PK of the child relation.



Identifying and non-identifying relationship approaches

In non-identifying relationship the FK (scientifically) has NOT to be a part of the PK of the child relation.

type

t_id	t_name
1	Furniture
2	Dishes

category

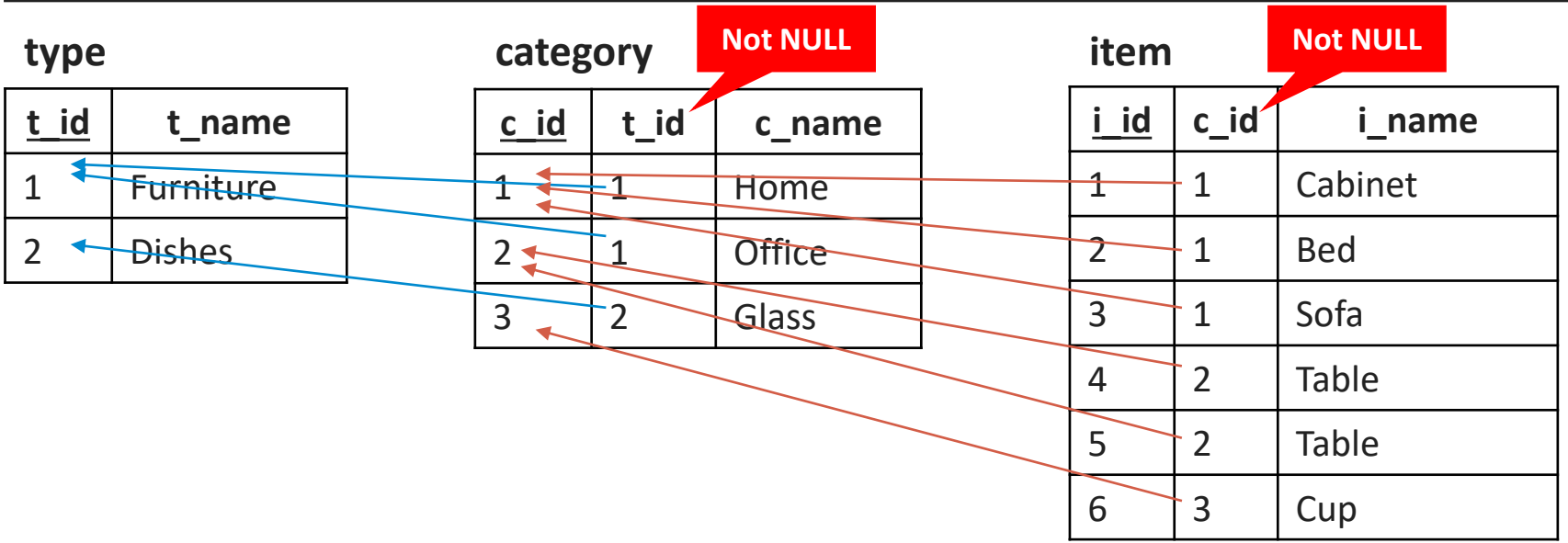
c_id	t_id	c_name
1	1	Home
2	1	Office
3	2	Glass

item

i_id	c_id	i_name
1	1	Cabinet
2	1	Bed
3	1	Sofa
4	2	Table
5	2	Table
6	*NULL*	Cup

Can we have small PKs in identifying relationships?

Technically, we may set “not NULL” property on FK and get a “semi-identifying” relationship.



How to create a relationship?

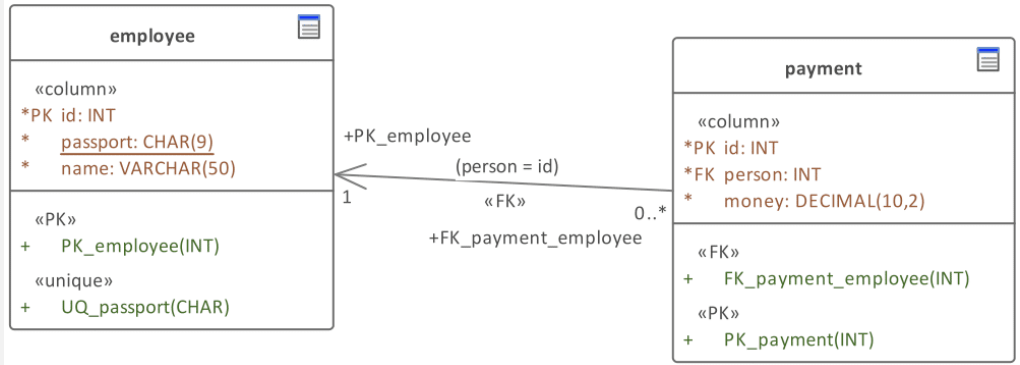
With an SQL query

```
CREATE TABLE `payment`
(
 `id` INT UNSIGNED NOT NULL AUTO_INCREMENT,
 `person` INT UNSIGNED NOT NULL,
 `money` DECIMAL(10,2) NOT NULL,
 CONSTRAINT `PK_payment` PRIMARY KEY (`id`)
);

CREATE TABLE `employee`
(
 `id` INT UNSIGNED NOT NULL AUTO_INCREMENT,
 `passport` CHAR(9) NOT NULL,
 `name` VARCHAR(50) NOT NULL,
 CONSTRAINT `PK_employee` PRIMARY KEY (`id`)
);

ALTER TABLE `employee`
 ADD CONSTRAINT `UQ_passport` UNIQUE (`passport`);

ALTER TABLE `payment` ADD CONSTRAINT `FK_payment_employee`
  FOREIGN KEY (`person`) REFERENCES `employee` (`id`)
   ON DELETE CASCADE ON UPDATE RESTRICT;
```



How to create a relationship?

Using some special tool

```
-- Here goes live demo :)
```

Live demo in Sparx Enterprise Architect

<epam>

Relationships – Part 2

Relational Databases Basics

TRAINING
CENTER

<epam>