

SmartCook project



**Co-funded by
the European Union**



**VOŠ | SPŠ | SOŠ
Varnsdorf**



The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Database

step-by-step creation of a simple databases

Specification of the problem

In our database, we want to manage information about **recipes** and its **ingredients**.

An example would be tea, which contains, for example, hot water, a dried mixture of tea leaves and optionally sugar.



Ingredients

We want to record the **name** of the ingredients. To simplify our task, we will not register anything else with them. To clearly distinguish the ingredients from each other, we choose a unique identifier in the form of a unique number.

- **id**
 - unique value for each record
 - a positive integer
 - required value
- **name**
 - text
 - required value

Recipe

We will also simplify the recipe itself. We will only record the name, the date of entering the recipe into the database and again a unique identifier.

- **id**
 - unique value for each record
 - a positive integer
 - required value
- **name**
 - text
 - required value
- **dtm**
 - Date and time: YYYY-MM-DD
HH:MM:SS
 - required value

Table with relationships

recipe

id	name	dtm
1	tea	2023-09-10 14:00:00
2	pancake	2023-09-10 15:20:00
3	chicken schnitzel	2023-10-01 16:00:00
4	cappuccino	2023-10-15 08:00:00

We can visualize the data in tables.

ingredient

id	name
1	water
2	tea leaves
3	salt
4	sugar
5	flour
6	pork
7	egg
8	milk
9	chicken breast

Relationships between records

We record recipes and ingredients, but we do not link them together. It is not yet possible to determine what ingredients the tea is made from.

So, we must think about the **relationship** itself and what we expect from it.

Relationship properties:

- Recipe must have at least one ingredient. There may be more.
- An ingredient can be in more than one recipe. For example, milk can be in a pancake recipe as well as a milkshake.
- From the relationship of the ingredient to the recipe, we should also know the quantity.

E-R diagram (MySQL Workbench)

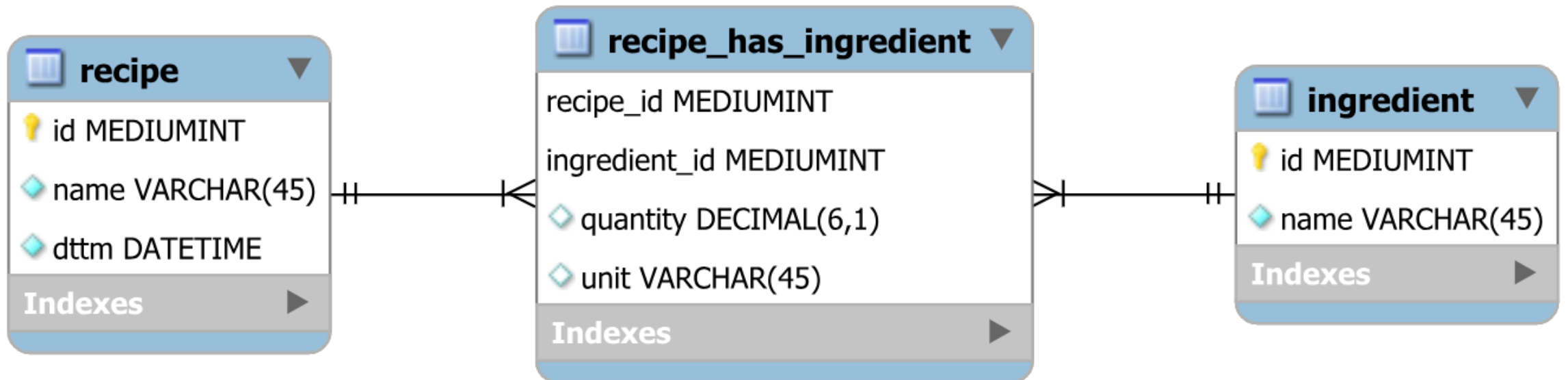


Table again

recipe

id	name	dttm
1	tea	2023-09-10 14:00:00
2	pancake	2023-09-10 15:20:00
3	chicken schnitzel	2023-10-01 16:00:00
4	cappuccino	2023-10-15 08:00:00

recipe_has_ingredient

recipe_id	ingredient_id	quantity	unit
1	1	250	ml
1	2	3	g
1	4	4	g

ingredient

id	name
1	water
2	tea leaves
3	salt
4	sugar
5	flour
6	pork
7	egg
8	milk
9	chicken breast

File formats

Databases also use special data file formats for data transfer:

- JSON
- XML
- CSV

```
{ } tea.json > [ ] ingredients > { } 2 > unit
1   {
2     "id": 1,
3     "name": "Tea",
4     "ingredients": [
5       {
6         "id": 1,
7         "name": "water",
8         "quantity": 250,
9         "unit": "ml"
10      },
11     {
12       "id": 2,
13       "name": "tea leaves",
14       "quantity": 3,
15       "unit": "g"
16     },
17     {
18       "id": 4,
19       "name": "sugar",
20       "quantity": 4,
21       "unit": "g"
22     }
23   ]
24 }
```

SQL

```
-- -----  
-- Table `smartcook`.`recipe_has_ingredient`  
-- -----  
CREATE TABLE IF NOT EXISTS `smartcook`.`recipe_has_ingredient` (  
  `recipe_id` MEDIUMINT UNSIGNED NOT NULL,  
  `ingredient_id` MEDIUMINT UNSIGNED NOT NULL,  
  `quantity` DECIMAL(6,1) NULL DEFAULT 1,  
  `unit` VARCHAR(45) NULL,  
  PRIMARY KEY (`recipe_id`, `ingredient_id`),  
  INDEX `fk_recipe_has_ingredient_ingredient1_idx` (`ingredient_id` ASC),  
  INDEX `fk_recipe_has_ingredient_idx` (`recipe_id` ASC),  
  CONSTRAINT `fk_recipe_has_ingredient`  
    FOREIGN KEY (`recipe_id`)  
    REFERENCES `smartcook`.`recipe` (`id`)  
    ON DELETE RESTRICT  
    ON UPDATE RESTRICT,  
  CONSTRAINT `fk_recipe_has_ingredient_ingredient1`  
    FOREIGN KEY (`ingredient_id`)  
    REFERENCES `smartcook`.`ingredient` (`id`)  
    ON DELETE RESTRICT  
    ON UPDATE RESTRICT)  
ENGINE = InnoDB;
```

SQL

```
INSERT INTO ingredient VALUES (NULL, 'sugar')
```

```
INSERT INTO ingredient(name) VALUES ('salt'), ('milk'), ('water')
```

```
UPDATE ingredient SET name='butter' WHERE id=4
```

```
DELETE FROM ingredient WHERE id=3
```