

## Projekt SQL

Cílem tohoto projektu je analyzovat ekonomická data, která máme k dispozici o České republice a zemích světa. A prostřednictvím dvou nově vytvořených tabulek odpovědět na pět výzkumných otázek:

1. Rostou v průběhu let mzdy ve všech odvětvích, nebo v některých klesají?
2. Kolik je možné si koupit litrů mléka a kilogramů chleba za první a poslední srovnatelné období v dostupných datech cen a mezd?
3. Která kategorie potravin zdražuje nejpomaleji (je u ní nejnižší procentuální meziroční nárůst)?
4. Existuje rok, ve kterém byl meziroční nárůst cen potravin výrazně vyšší než růst mezd (větší než 10%)?
5. Má výška HDP vliv na změny ve mzdách a cenách potravin? Neboli, pokud HDP vzroste výrazněji v jednom roce, projeví se to na cenách potravin či mzdách ve stejném nebo následujícím roce výraznějším růstem?

Z databáze Engeto budeme tvořit dvě tabulky, které pak využijeme k analýze:

t\_Daniel\_Rizak\_project\_SQL\_primary\_final (obsahující data o mzdách a cenách potravin v Česku sjednocených na totožné porovnatelné období – společné roky) a

t\_Daniel\_Rizak\_project\_SQL\_secondary\_final (s dodatečnými daty o dalších evropských státech).

### Prvotní průzkum výchozích tabulek

K výzkumu dat využijí DBeaver, který jsme standardně využívali během kurzu.

Prozkoumejme nejdřív tabulky, které máme k dispozici. Hlavním zdrojem dat o mzdách v odvětvích bude czechia\_payroll a pro ceny druhů zboží to bude tabulka czechia\_price. Pro ekonomická data (HDP, daně, Gini apod.) využijeme nejspíš tabulku economies, popř. countries.

Jaký je rozsah období měření v tabulce czechia\_price? Zeptáme se dotazem:

```
SELECT min(date_from), max(date_from), min(date_to), max(date_to)
FROM czechia_price;
```

Vidíme, že měření probíhalo od ledna 2006 do prosince 2018. Pro výzkum musíme zvolit jen jeden sloupec s časovou veličinou, kterou využijeme v nové tabulce. Volím "date\_to", která sahá až do 16. prosince. Obdobně pro rozsah roků měření v tabulce mezd se zeptáme:

```
SELECT min(payload_year), max(payload_year)
FROM czechia_payroll cp;
```

Jsou to roky 2000 až 2021. Z výše uvedených dvou období musíme udělat průnik, protože obě proměnné – mzdy a ceny – budeme mj. srovnávat mezi sebou. Výsledné období 8.1.2006 až 16.12.2018, celkem 13 let, bude výchozí pro první tabulku, kterou vytvoříme.

V tabulce czechia\_payroll se vyskytuje calculation code s hodnotou 100 nebo 200. Jak plyne z czechia\_payroll\_calculation, 100 je fyzická hodnota a 200 nějakým způsobem přepočtená. Nevíme, do jaké míry je přepočtená hodnota pro výzkum spolehlivá, zda jde o teoretickou nebo skutečně naměřenou. Budu tedy dál pracovat jen s fyzickým kódem 100. Ještě z tabulky czechia\_payroll\_type\_code zjistíme, že relevantní je pro nás jen kód 5958 tj. průměrná hrubá mzda na zaměstnance. Ta je vyjádřená v korunách, jak plyne z czechia\_payroll\_unit. Kolik měření mezd máme k dispozici? Zjistíme:

```
SELECT*
FROM czechia_payroll cp
WHERE value_type_code = 5958 AND calculation_code = 100 AND payroll_year
BETWEEN 2006 AND 2018;
```

Na spodní liště pod tabulkou v calculate total row count dostáváme 1040 hodnot. Na každý z 13 roků měření připadá 80 hodnot. Budeme-li pracovat se čtvrtletími, bude to jen 20 hodnot pro každé čtvrtletí a dostaneme dlouhou časovou řadu s 52 čtvrtroky. Ovšem HDP, na kterém budeme zkoumat závislost mezd, se měří ročně, jak později vyplyne z tabulky economies. Volím tedy interval roční nikoliv čtvrtletní a bude stejný pro mzdy, ceny i HDP.

Z tabulky czechia\_payroll\_industry\_branch ještě vidíme, že pracujeme s 19 odvětvími. Krát 13 let měření a čtyři původně měřená čtvrtletí dává 988 hodnot. Vyšlo nám ale 1040. Nejspíš jsou někde nulové hodnoty nebo nastal jiný zádrhel. Úpravou předchozího dotazu se ptáme:

```
SELECT count(industry_branch_code), industry_branch_code, payroll_year
FROM czechia_payroll cp
WHERE value_type_code = 5958 AND calculation_code = 100 AND payroll_year
BETWEEN 2006 AND 2018
GROUP BY industry_branch_code, payroll_year;
```

Ano, nulové hodnoty zohledníme v dalších dotazech. Tabulku prolistujeme a zjistíme, že pro každý rok a každý průmyslový obor máme 4 měření. Na dolní liště vidíme 260 hodnot. Odečteme 13 nulových v letech 2006 až 2018 a dostaneme 247. To přesně odpovídá 19 oborům krát 13 let. Nic nechybí ani nepřebývá. Ještě omrkneme tabulku czechia\_price:

```
SELECT DISTINCT YEAR(date_from) AS rok, category_code, count(1)
FROM czechia_price
GROUP BY rok, category_code
ORDER BY count(1) DESC, rok;
```

Měření postupně ubývá. Ze 750 v roce 2006 až na 180 v letech 2011-2018. Podezřele málo máme měření u kategorie 2,000,001. Jde o chybu nebo specifický produkt? Zeptáme se na to:

```
SELECT code, name
FROM czechia_price_category cpc
WHERE code = '2000001';
```

Je to Kapr živý. Do statistik cen vstupuje jen na Vánoce. O chybu tedy nejde. Na spodní liště pod Calculate total row count vidíme 342 řádků. Z tabulky czechia\_price\_category víme, že máme 27 druhů zboží. Tedy 27 krát 13 let měření musí dát 351 řádků. Někde nám devět řádků chybí. Zjistíme proč:

```
SELECT count(rok), name
FROM (
    SELECT round(avg(cp.value),2) AS avg_price,
           YEAR(cp.date_to) AS rok,
           cpc.name
    FROM czechia_price cp
    JOIN czechia_price_category cpc
        ON cp.category_code = cpc.code
    GROUP BY cpc.name, rok
) tab
GROUP BY name
ORDER BY count(rok)
```

Jakostní víno bílé bylo měřeno jen ve čtyřech letech a ne ve 13 jako ostatní zboží. To je těch devět řádků, které nám chyběly. Můžeme se ještě zeptat, v kterých rocích sběr dat o cenách probíhal:

```
SELECT
    min(YEAR(cp.date_to)) AS minim_rok,
    max(YEAR(cp.date_to)) AS maxim_rok,
    cpc.name
```

```
FROM czechia_price cp
JOIN czechia_price_category cpc
    ON cp.category_code = cpc.code
GROUP BY cpc.name;
```

A vidíme, že cena vína byla měřena jen v letech 2015-2018. Pojďme ještě na tabulku economies, ze které vezmeme data o hrubém domácím produktu, daních a dalších ukazatelích.

```
SELECT*
FROM economies e
WHERE country = 'Czech Republic';
```

### Primární tabulka

Pro stěžejní hodnotu potřebnou k zodpovězení otázky č. 5 - HDP - máme k dispozici data od r. 1990 do r. 2020. To nám vyhovuje, ceny a mzdy budeme vyhodnocovat v období 2006-2018.

Výchozí tabulky, které budeme potřebovat, jsme prozkoumali. Jdeme tvořit primární tabulku projektu.

Pro zodpovězení otázek 1-5 musíme v jedné tabulce spojit ceny i mzdy. Inspirací může být tabulka czechia\_payroll z databáze Engeto. V ní jsou v jednom sloupci "value" uloženy společně dvě rozdílné proměnné: mzdy a počet zaměstnanců. Rozlišené jsou pomocí unit\_code.

Nezbytnou veličinou je rok. Pro data o mzdách převezmeme hodnoty ze sloupce "payroll\_year" z tabulky czechia\_payroll. U cen z tabulky czechia\_price využijeme sloupec „date\_to“, jen upravíme formát.

V projektových otázkách budeme zkoumat růsty mezd, cen a hrubého domácího produktu. Musíme tedy vždy porovnávat zvolenou proměnnou v běžném roce (n) a v roce předchozím (n-1). Z nich později spočteme procentuální nebo indexové růsty resp. poklesy.

Kategorie zboží mají různé váhové resp. objemové jednotky. Získáme je z czechia\_price\_category a můžeme klidně spojit "price\_value" a "price\_unit". Sloupec pojmenujme "unit".

Určitě budeme pracovat s kódem kategorie, ať už mzdové (kód odvětví) nebo cenové (kód potravin). Nazvěme tedy sloupec "category\_code". A stejně tak se jménem odvětví resp. názvem potravin. Pojmenujme ho "category\_name".

Pro všechny potraviny máme k dispozici i region, kde probíhala měření. Škoda tak cenné informace zahodit. Příslušné dva sloupce budou "region\_code" a "region\_name". Zůstanou null pro část tabulky se mzdami, kde informace o regionech nemáme.

### Dočasná tabulka mezd

Základní kámen pro novou tabulku se mzdami vynikne spojením dvou vnořených selectů. Spojíme je přes industry\_branch\_code (kód odvětví) a year (rok). Jen v druhém vnořeném selectu (pay 2) bude rok posunutý o jeden, abychom na stejném řádku dostali hodnotu předchozího roku.

```
SELECT pay1.`year`, pay1.industry_branch_code, pay1.value_current_year,
pay2.value_prev_year
FROM (SELECT
    payroll_year AS `year`,
    industry_branch_code,
    avg(value) AS value_current_year
    FROM czechia_payroll cp
    WHERE value_type_code = 5958 AND industry_branch_code IS NOT NULL
    AND payroll_year BETWEEN 2005 AND 2018
    GROUP BY payroll_year, industry_branch_code) AS pay1
LEFT JOIN (SELECT
    payroll_year AS `year`,
    industry_branch_code,
    avg(value) AS value_prev_year
    FROM czechia_payroll cp
```

```

WHERE value_type_code = 5958 AND industry_branch_code IS NOT NULL
AND payroll_year BETWEEN 2005 AND 2018
GROUP BY payroll_year, industry_branch_code) AS pay2
ON pay1.industry_branch_code = pay2.industry_branch_code
AND pay1.year = pay2.year+1;

```

Běžný rok nazýváme “value\_current\_year” a předchozí bude “value\_prev\_year”. Nezapomene-me na podmínky, value\_type\_code = 5958 (kód pro mzdy) a industry\_branch\_code chceme nenulový. Rozpětí let je 2005 až 2018, čímž získáme hodnotu z předchozího roku pro 2006, od kterého zkoumáme mzdy a ceny. Mzdy rovnou zprůměrujeme. Pokud chceme zkoumat jejich závislost s cenami a HDP a měřit trendy, stejně to budeme muset udělat. Nevíme v jakých regionech probíhalo jejich měření, takže je rozlišíme jen v rámci roků a odvětví.

Ještě připojíme názvy odvětví z tabulky czechia\_payroll\_industry\_branch a základ tabulky mezd můžeme rovnou vytvořit:

```

CREATE TABLE t_payroll_temporary AS
SELECT pay1.`year`, pay1.value_current_year, pay2.value_prev_year,
pay1.industry_branch_code, cpib.name
FROM (
    SELECT payroll_year AS `year`, industry_branch_code, avg(value) AS
value_current_year
FROM czechia_payroll cp
WHERE value_type_code = 5958 AND calculation_code = 100 AND
industry_branch_code IS NOT NULL AND payroll_year BETWEEN 2005 AND 2018
GROUP BY payroll_year, industry_branch_code) AS pay1
LEFT JOIN (
    SELECT payroll_year AS `year`, industry_branch_code, avg(value) AS
value_prev_year
FROM czechia_payroll cp
WHERE value_type_code = 5958 AND calculation_code = 100 AND
industry_branch_code IS NOT NULL AND payroll_year BETWEEN 2005 AND 2018
GROUP BY payroll_year, industry_branch_code) AS pay2
ON pay1.industry_branch_code = pay2.industry_branch_code
AND pay1.year = pay2.year+1
LEFT JOIN czechia_payroll_industry_branch cpib
ON pay1.industry_branch_code = cpib.code;

```

Přidáme potřebné sloupce a přejmenujeme je. Stejný formát dostane předběžná tabulka cen, než obě spojíme do výsledné primární tabulky. Do sloupce „unit“ vložíme zkratku pro koruny.

```

ALTER TABLE t_payroll_temporary
ADD COLUMN unit VARCHAR(10);

UPDATE t_payroll_temporary
SET unit = 'czk';

ALTER TABLE t_payroll_temporary
RENAME COLUMN name TO category_name;

ALTER TABLE t_payroll_temporary
RENAME COLUMN industry_branch_code TO category_code;

ALTER TABLE t_payroll_temporary
ADD COLUMN region_code CHAR(5);

ALTER TABLE t_payroll_temporary
ADD COLUMN region_name VARCHAR(255);

```

### Dočasná tabulka cen

Obdobně jako v předchozím případě, spojíme dva vnořené selecty přes kód kategorií, kód regionů a roky. Díky zápisu "year+1" získáme opět hodnotu pro předchozí rok na každém řádku běžného roku. Názvy sloupců volíme stejně jako u mezd.

```
SELECT prices.year, prices.value_current_year, prices2.value_prev_year,
prices.category_code, prices.region_code
FROM (
    SELECT YEAR(date_to) AS `year`, category_code, region_code, avg(value)
    AS value_current_year
    FROM czechia_price cp
    WHERE YEAR(date_to) BETWEEN 2005 AND 2018 AND region_code IS NOT NULL
    GROUP BY YEAR(date_to), category_code, region_code) prices
LEFT JOIN (
    SELECT YEAR(date_to) AS `year`, category_code, region_code, avg(value)
    AS value_prev_year
    FROM czechia_price cp
    WHERE YEAR(date_to) BETWEEN 2005 AND 2018 AND region_code IS NOT NULL
    GROUP BY YEAR(date_to), category_code, region_code) prices2
ON prices.category_code = prices2.category_code
AND prices.region_code = prices2.region_code
AND prices.year = prices2.year+1;
```

Ještě budeme připojovat jména kategorií zboží a názvy region, tak si zjednodušíme život a vytvoříme rovnou tabulku z výše uvedeného dotazu. Je to užitečný mezikrok:

```
CREATE TABLE t_price_temporary AS ...
```

A zkopírujeme celý dotaz výše.

Připojíme kategorie zboží a názvy regionů přes jejich kódy. V druhém mezikroku už máme stejné sloupce jako v předběžné tabulce mezd. Rovnou ji vytvoříme:

```
CREATE TABLE t_price_temporary_2 AS
SELECT tpt.*, cpc.name AS category_name, concat(cpc.price_value, ' ',
cpc.price_unit) AS unit, cr.name AS region_name
FROM t_price_temporary tpt
JOIN czechia_price_category cpc
ON tpt.category_code = cpc.code
JOIN czechia_region cr
ON tpt.region_code = cr.code;
```

Pomocí funkce UNION spojíme obě předběžné tabulky a získáme výslednou primární tabulku, kterou chceme:

```
CREATE TABLE t_Daniel_Rizak_project_SQL_primary_final
SELECT `year`, value_current_year, value_prev_year, unit, category_code,
category_name, region_code, region_name
FROM t_payroll_temporary tpt
UNION
SELECT `year`, value_current_year, value_prev_year, unit, category_code,
category_name, region_code, region_name
FROM t_price_temporary_2 tpt2;
```

Přidejme ještě id, které bude zároveň primárním klíčem a může nám ulehčit práci s dotazy, snado označit řádky atp.

```
ALTER TABLE t_Daniel_Rizak_project_SQL_primary_final
ADD COLUMN id INT NOT NULL AUTO_INCREMENT PRIMARY KEY;
```

Primární tabulka je hotová. Vytvoříme teď sekundární s ekonomickými ukazateli pro evropské země.

### Sekundární tabulka

V tabulce economies, odkud bereme data, nejsou definované země podle kontinentu. Tento údaj máme v tabulce countries. Musíme tedy informace propojit přes vnořený select.

```
SELECT*
FROM economies e
WHERE e.country IN (
    SELECT c.country
    FROM countries c
    WHERE continent = 'Europe'
);
```

Stejně jako u cen a mezd, budeme potřebovat růstové indexy i u HDP. Vedle běžného roku tak chceme sloupec hodnot roku předchozího. Jejich vydělením získáme později index růstu produktu. V otázce č. 5 se nás ptají na vztah cen resp. mezd a produktu v předchozím nebo stejném roce. Tj. pro růst/pokles cen a mezd v roce 2006 (odkdy máme data v primární tabulce) budeme hledat stav HDP v roce 2005. Pro výpočet růstového indexu tak bereme HDP už od roku 2004. Podílem hodnot r. 2005 vůči 2004 získáme růstový index HDP pro rok 2005. S pomocí předchozího dotazu rovnou tvoříme tabulku s využitím common table expression.

```
CREATE TABLE t_daniel_rizak_project_sql_secondary_final
WITH half_table AS (
    SELECT*
    FROM economies e
    WHERE e.country IN (
        SELECT c.country
        FROM countries c
        WHERE continent = 'Europe')
    AND e.`year` BETWEEN 2004 AND 2018)
SELECT ht1.country, ht1.`year`, ht1.GDP AS GDP_current_year, ht2.GDP AS
    GDP_previous_year, ht1.population, ht1.taxes, ht1.fertility,
    ht1.mortality_under5 AS mortality_under5
FROM half_table ht1
JOIN half_table ht2
    ON ht1.country = ht2.country
    AND ht1.`year` = ht2.`year`+1
ORDER BY country, `year` ASC;
```

Alternativní způsob jejího vytvoření je i pomocí funkce LAG, díky níž se hodnoty HDP ze sloupce "GDP\_current\_year" posunou o řádek a získáme "GDP\_previous\_year", tj. HDP v předchozím roce.

```
CREATE TABLE t_daniel_rizak_project_sql_secondary_final
SELECT country, `year`, GDP AS GDP_current_year,
    LAG(GDP) OVER (ORDER BY `year`) AS GDP_previous_year,
    population, taxes, fertility, mortality_under5 AS mortality_under5
FROM economies e
WHERE e.country IN (
    SELECT c.country
    FROM countries c
    WHERE continent = 'Europe'
)
AND e.`year` BETWEEN 2004 AND 2018
ORDER BY country, `year` ASC;
```

Obě tabulky máme vytvořené, pojďme na výzkumné otázky.

## Výzkumné otázky

### 1. Rostou v průběhu let mzdy ve všech odvětvích, nebo v některých klesají?

Jednoduchým dotazem zjistíme indexy růstu produktu. Kde je index větší než 1, mzdy meziročně rostly. Kde je menší než 1, mzdy poklesly. Pokud by byl index roven přesně 1, pak mzdy meziročně stagnovaly. Nezapomeneme na podmínku unit = 'czk'. Zajímá nás jen část primární tabulky se mzdami, které jsou v korunách. Pro část tabulky s cenami zboží je ve sloupci unit množtevní jednotka zboží (kg, l apod.)

```
SELECT `year`, value_current_year/value_prev_year, category_name,
       CASE WHEN value_current_year/value_prev_year > 1 THEN 1
       ELSE 0
       END AS growth
FROM t_daniel_rizak_project_sql_primary_final tdrpspf
WHERE unit = 'czk' AND value_prev_year IS NOT NULL;
```

Tabulku s výpočtem můžeme zjednodušit a ponechat pouze rok, odvětví a binární index (1 = růst, 0 = pokles).

```
SELECT `year`, category_name, growth
FROM (
  SELECT `year`, value_current_year/value_prev_year, category_name,
         CASE WHEN value_current_year/value_prev_year > 1 THEN 1
         ELSE 0
         END AS growth
  FROM t_daniel_rizak_project_sql_primary_final tdrpspf
  WHERE unit = 'czk' AND value_prev_year IS NOT NULL) AS gr;
```

Vidíme, že roků s meziročním poklesem mezd je výrazná menšina. Můžeme je tedy vybrat dotazem níže a rovnou je srovnáme podle odvětví:

```
SELECT `year`, category_name, growth
FROM (
  SELECT `year`, value_current_year/value_prev_year, category_name,
         CASE WHEN value_current_year/value_prev_year > 1 THEN 1
         ELSE 0
         END AS growth
  FROM t_daniel_rizak_project_sql_primary_final tdrpspf
  WHERE unit = 'czk' AND value_prev_year IS NOT NULL) AS gr
WHERE growth = 0
ORDER BY category_name, `year`;
```

Ve všech ostatních letech a odvětvích kromě těchto mzdy rostly. Nejhorší na tom byla Těžba a dobývání. Platy tam poklesly celkem čtyřikrát v letech 2009, 2013, 2014 a 2016. Nejslabší rok za dobu měření získáme malou úpravou klauzule ORDER BY:

```
SELECT `year`, category_name, growth
FROM (
  SELECT `year`, value_current_year/value_prev_year, category_name,
         CASE WHEN value_current_year/value_prev_year > 1 THEN 1
         ELSE 0
         END AS growth
  FROM t_daniel_rizak_project_sql_primary_final tdrpspf
  WHERE unit = 'czk' AND value_prev_year IS NOT NULL) AS gr
WHERE growth = 0
ORDER BY `year`, category_name;
```



Nejhorším rokem z pohledu mezd byl 2013. V deseti z devatenácti odvětví tehdy mzdové příjmy klesly. Druhý v pořadí byl rok 2009 se čtyřmi odvětvími ve mzdovém poklesu.

**Odpověď:** Ano, v některých odvětvích mzdy klesly. Bylo to v těchto případech:

rok	odvětví	rok	odvětví
2009	Těžba a dobývání	2013	Kulturní, zábavní a rekreační činnosti
2009	Ubytování, stravování a pohostinství	2013	Peněžnictví a pojišťovnictví
2009	Velkoobchod a maloob.; opravy a údrž. Motor. Vozidel	2013	Profesní, vědecké a technické činnosti
2009	Zemědělství, lesnictví, rybářství	2013	Stavebnictví
2010	Profesní, vědecké a technické činnosti	2013	Těžba a dobývání
2010	Veřejná správa a obrana; povinné sociální zabezpečení	2013	Velkoobchod a maloob.; opravy a údržba motor. Vozidel
2010	Vzdělávání	2013	Výr. a rozvod elektřiny, plynu, tepla a klimatiz. Vzduchu
2011	Kulturní, zábavní a rekreační činnosti	2013	Zásobování vodou; činnosti související s odpady a sanacemi
2011	Ubytování, stravování a pohostinství	2014	Těžba a dobývání
2011	Veřejná správa a obrana; povinné sociální zabezpečení	2015	Výr. a rozvod elektřiny, plynu, tepla a klimatiz. Vzduchu
2013	Činnosti v oblasti nemovitostí	2016	Těžba a dobývání
2013	Informační a komunikační činnosti		

## 2. Kolik je možné si koupit litrů mléka a kg chleba za první a poslední srovnatelné období v dostupných datech cen a mezd?

Prvním srovnatelným obdobím je pro nás rok 2006, posledním rok 2018. Ceny potravin se v krajích liší. Průměrný příjem v každém odvětví vydělíme cenou kilogramu chleba resp. litru mléka v jednotlivých krajích a dostaneme množství těchto potravin, které si pořídíme z jedné mzdy. Kód pro chléb je 111301 a pro mléko 114201. Kódy získáme z primární tabulky. Připojíme její dvě různé části na sebe přes sloupec "rok" (year). První část obsahuje mzdy (unit = ,czk') a druhá ceny potravin (unit se nerovná ,czk').

```
SELECT fin1.`year`, fin1.value_current_year AS avg_wage_sector,
       fin1.category_name AS sector,
       fin2.value_current_year AS bread_price_CZK_region, fin2.region_name,
       round(fin1.value_current_year/fin2.value_current_year) AS
       bread_kg_for_wage
FROM (
  SELECT `year`, value_current_year, category_name
  FROM t_daniel_rizak_project_sql_primary_final
  WHERE unit = ,czk' AND `year` IN (2006, 2018)) fin1
JOIN (
  SELECT `year`, value_current_year, category_name, region_name,
  category_code
  FROM t_daniel_rizak_project_sql_primary_final
  WHERE unit != ,czk' AND `year` IN (2006, 2018) AND category_code =
  ,111301') fin2
ON fin1.`year` = fin2.`year`;
```

Množství ve sloupci bread\_kg\_for\_wage (kilogram chleba za jednu mzdu) je zaokrouhleno na celé chleby pro lepší čitelnost výsledku. A navíc, můžeme si koupit polovinu konzumního chleba, ale už ne např. 0,3 nebo 0,9 kg. Balení po litrech mléka nemůžeme v obchodě dělit vůbec. Dotaz na množství mléka za jednu mzdu se liší jen jeho kódem, viz níže:



```

SELECT fin1.`year`, fin1.value_current_year AS avg_wage_sector,
       fin1.category_name AS sector,
       fin2.value_current_year AS bread_price_CZK_region, fin2.region_name,
       round(fin1.value_current_year/fin2.value_current_year) AS
       bread_kg_for_wage
FROM (
  SELECT `year`, value_current_year, category_name
  FROM t_daniel_rizak_project_sql_primary_final
  WHERE unit = 'czk' AND `year` IN (2006, 2018)) fin1
JOIN (
  SELECT `year`, value_current_year, category_name, region_name,
         category_code
  FROM t_daniel_rizak_project_sql_primary_final
  WHERE unit != 'czk' AND `year` IN (2006, 2018) AND category_code =
  '114201') fin2
ON fin1.`year` = fin2.`year`;

```

Přidáním klauzule **ORDER BY bread\_kg\_for\_wage** resp. **ORDER BY milk\_litres\_for\_wage**, snadno zjistíme, které kraje a hospodářská odvětví jsou na tom nejlépe a které nejhůře.

**Odpověď:** Nejméně chleba si koupí zaměstnanci v sektoru ubytování, stravování a pohostinství a to platí jak v r. 2006 tak 2018. Asi nepřekvapí, že nejlépe na tom bylo odvětví peněžnictví a pojišťovnictví (rok 2006) a také informační a komunikační činnosti (zejména rok 2018).

Kdo žije v Moravskoslezském kraji a pracuje v ubytování a gastronomii, koupí si za svůj plat jen 625 kilogramových chlebů. Kdo je z Karlovarska a živí se peněžnictvím získá 2581 kg chleba, tj. více než čtyřnásobek.

U mléka je situace velmi podobná. Nejméně potravin získají pracovníci v odvětví ubytování, stravování a pohostinství. Nejvíce v první řadě lidé z IT oborů a peněžnictví/pojišťovnictví. Jihočeši pracující v ubytování a gastronomii dostanou jen 723 litrů mléka za svou mzdu, proti „ajtákům“ z Karlovarska, kteří si pořídí 3052 litrů. To je také více než čtyřnásobek jako v případě chleba.

### 3. Která kategorie potravin zdražuje nejpomaleji (je u ní nejnižší procentuální meziroční nárůst)?

Opět pro zkoumání růstu využijeme index cen, tj. cena běžného roku dělena cenou roku předchozího. Z indexu odečteme 1 (de facto 100%) a po vynásobení 100 dostaneme hodnotu meziročního růstu v procentech. Všechny růsty v procentech sečteme (funkce sum). Někdy cena rostla, jindy mohla klesat, součtem zjistíme, které potraviny v průběhu let zdražily nejvíce resp. nejméně.

```

SELECT sum(gr.price_growth_percent) AS sum_growth, gr.category_name,
       gr.region_name
FROM (
  SELECT `year`, round((value_current_year/value_prev_year-1)*100,2) AS
  price_growth_percent, category_name, region_name
  FROM t_daniel_rizak_project_sql_primary_final
  WHERE unit != 'czk' AND value_prev_year IS NOT NULL) AS gr
GROUP BY gr.category_name, gr.region_name
ORDER BY sum_growth;

```

**Odpověď:**

Cukr krystal ve všech krajích Česka dokonce zlevnil a to v rozpětí 18% (Královéhradecký kraj) až téměř 29% (Karlovarsko). Rajska jablka červená zlevnila o 3 až 16% v jedenácti krajích ze čtrnácti. V kraji Vysočina poklesla cena bílého vína o necelé procento. Všechny ostatní potraviny zdražily. Nejvíce papriky o 74 procent v Jihomoravském kraji až o 108 procent v Královéhradeckém. Dále máslo o 68% v Jihočeském kraji až o 87% na Plzeňsku.

Když se budeme striktně držet otázky a zkoumat jen zdračující potraviny, ne zlevňující, musíme dotaz upravit. Přidáme klauzuli HAVING:

```
SELECT sum(gr.price_growth_percent) AS sum_growth, gr.category_name,
       gr.region_name
FROM (
    SELECT `year`, round((value_current_year/value_prev_year-1)*100,2) AS
           price_growth_percent, category_name, region_name
    FROM t_daniel_rizak_project_sql_primary_final
    WHERE unit != 'czk' AND value_prev_year IS NOT NULL) AS gr
GROUP BY gr.category_name, gr.region_name
HAVING sum_growth >0
ORDER BY sum_growth;
```

#### Druhá část odpovědi:

Nejvíce zdražující potraviny už jsme zmínili. Nejméně ceny rostly u rajčat na jižní Moravě (0,18%), banánů na Plzeňsku (0,34%) a u bílého vína ve Středočeském kraji (0,97%).

#### 4. Existuje rok, ve kterém byl meziroční růst cen potravin výrazně vyšší než růst mezd (větší než 10 %)?

Meziroční indexy cen zboží odečteme od meziročních indexů mezd a tam kde bude tento rozdíl vyšší než 0,1 byl meziroční růst cen potravin rychlejší o více než 10% proti růstu mezd. Získáme indexy cen kategorií zboží v průběhu let 2006-2018:

```
SELECT `year`, avg(value_current_year/value_prev_year) AS grow_prices,
       category_name
FROM t_daniel_rizak_project_sql_primary_final
WHERE unit != 'czk'
GROUP BY category_name, `year`
```

A indexy mezd ve stejném období:

```
SELECT `year`, avg(value_current_year/value_prev_year) AS grow_wages
FROM t_daniel_rizak_project_sql_primary_final
WHERE unit = 'czk'
GROUP BY `year`
```

Oba selecty spojíme přes roky a zanoříme je do dalšího selectu. V něm odečteme indexy cen zboží od indexů mezd a pomocí Case Expression odlišíme, kde je rozdíl růstu vyšší než 10 procent a kde není.

```
SELECT pri.`year`, pri.grow_prices, pri.category_name, pay.grow_wages,
       CASE WHEN pri.grow_prices-pay.grow_wages >0.1 THEN 'YES'
       ELSE 'no'
       END AS prices_growth_much_higher
FROM (
    SELECT `year`, avg(value_current_year/value_prev_year) AS grow_prices,
           category_name
    FROM t_daniel_rizak_project_sql_primary_final
    WHERE unit != 'czk'
    GROUP BY category_name, `year`) pri
JOIN (
    SELECT `year`, avg(value_current_year/value_prev_year) AS grow_wages
    FROM t_daniel_rizak_project_sql_primary_final
    WHERE unit = 'czk'
    GROUP BY `year`) pay
ON pri.`year` = pay.`year`
HAVING pri.grow_prices != 'czk'
```

Výsledek vidíme v dynamickém sloupci `prices_growth_much_higher`. Z indexu cen a mezd vypadl rok 2006, protože pro tento rok máme ceny zboží, ale už ne pro předchozí rok 2005. Nelze tedy spočítat index růstu cen v r. 2006.

Ještě vybereme pouze ty případy, které odpovídají zadání, tj. růst cen vyšší o více než 10% ve srovnání s růstem mezd.

```
SELECT pri.`year`, pri.grow_prices, pri.category_name, pay.grow_wages,
       CASE WHEN pri.grow_prices-pay.grow_wages >0.1 THEN 'YES'
       ELSE 'no'
       END AS prices_growth_much_higher
FROM (
    SELECT `year`, avg(value_current_year/value_prev_year) AS grow_prices,
    category_name
    FROM t_daniel_rizak_project_sql_primary_final
    WHERE unit != 'czk'
    GROUP BY category_name, `year`) pri
JOIN (
    SELECT `year`, avg(value_current_year/value_prev_year) AS grow_wages
    FROM t_daniel_rizak_project_sql_primary_final
    WHERE unit = 'czk'
    GROUP BY `year`) pay
ON pri.`year` = pay.`year`
HAVING pri.grow_prices IS NOT NULL AND prices_growth_much_higher LIKE 'YES'
ORDER BY `year`
```

#### Odpověď:

Kromě dvou roků 2009 a 2014 ve všech ostatních letech nastal alespoň jeden případ, kdy meziroční růst cen potravin byl výrazně vyšší než růst mezd. Z toho nejvíce jich bylo v r. 2013, celkem sedm druhů potravin. Jmenovitě kuřata, sýr eidam, mouka hladká, brambory, vaječné těstoviny, jablka a máslo. Zatímco ceny potravin rostly, mzdy v tomto roce mírně poklesly.

#### 5. Má výška HDP vliv na změny ve mzdách a cenách potravin? Neboli, pokud HDP vzroste výrazněji v jednom roce, projeví se to na cenách potravin či mzdách ve stejném nebo následujícím roce výraznějším růstem?

Jinými slovy zjistíme, zda růst/pokles HDP v roce N-1 je následován růstem/poklesem mezd nebo cen v roce N. A také zda růst/pokles HDP v roce N je následován růstem/poklesem mezd nebo cen v témže roce. Pro přehlednost nazvěme rok N-1 rokem předcházejícím a rok N rokem běžným.

Nejdřív získáme růstové indexy. Začneme vztahem produktu a mezd. Dotaz na databázi zní:

```
SELECT base.country, tab1.`year`,
       tab1.value_current_year/tab1.value_prev_year AS wage_grow_same_year,
       base.GDP_current_year/base.GDP_previous_year AS GDP_grow_prev_year,
       LEAD(base.GDP_current_year,19)OVER(ORDER BY
       base.`year`)/LEAD(base.GDP_previous_year,19)OVER(ORDER BY base.`year`)
       AS GDP_grow_same_year, tab1.category_name
FROM t_daniel_rizak_project_sql_secondary_final base
JOIN t_daniel_rizak_project_sql_primary_final tab1
ON base.`year`+1 = tab1.`year`
AND base.country = 'Czech Republic' AND tab1.unit = 'czk';
```

Bázi (base) je tabulka s HDP, tj. sekundární, na ni napojujeme tabulku s informacemi o mzdách resp. cenách (tab1), tj. primární. Spojujeme je přes sloupec "year" (rok). Abychom se s HDP dostali na rok předcházející (N-1) roku běžnému (N) se mzdou, stanovíme pro JOIN podmínku: `base.`year`+1 = tab1.`year``.

Chceme zkoumat i možnou závislost růstu mezd (jejich indexu) v tomto roce (N) na růstu HDP (jeho indexu) v tomtéž roce (N). Z původního "kroku zpátky" uděláme "krok dopředu" pomocí funkce LEAD. Jen musíme

poskočit o 19 řádků, protože odvětví je 19. Spojením tabulek máme pro každý rok indexu HDP 19 indexů růstu mezd pro každé odvětví.

Alternativní dotaz bez použití funkce Lead je tento:

```
SELECT tab1.country, base.`year`,
base.value_current_year/base.value_prev_year AS wage_grow_same_year,
tab1.GDP_current_year/tab1.GDP_previous_year AS GDP_grow_prev_year,
tab2.GDP_current_year/tab2.GDP_previous_year AS GDP_grow_same_year,
base.category_name
FROM t_daniel_rizak_project_sql_primary_final base
JOIN t_daniel_rizak_project_sql_secondary_final tab1
ON base.`year` = tab1.`year`+1
AND tab1.country = 'Czech Republic' AND base.unit = 'czk'
JOIN t_daniel_rizak_project_sql_secondary_final tab2
ON base.`year` = tab2.`year`
AND tab2.country = 'Czech Republic' AND base.unit = 'czk';
```

Jdeme na růstové indexy pro HDP a ceny potravin. Do roku 2014 je 26 kategorií potravin. Od roku 2015 až do konce měření v r. 2018 přibyl bílé víno, takže kategorií je 27. Funkce Lead se "skokem" o 26 pozic by v tomto případě v r. 2015 přestala fungovat. Zvolíme tedy dotaz bez Lead, podobný dotazu výše.

```
SELECT tab1.country, base.`year`,
base.value_current_year/base.value_prev_year AS price_grow_same_year,
tab1.GDP_current_year/tab1.GDP_previous_year AS GDP_grow_prev_year,
tab2.GDP_current_year/tab2.GDP_previous_year AS GDP_grow_same_year,
base.category_name, base.region_name
FROM t_daniel_rizak_project_sql_primary_final base
JOIN t_daniel_rizak_project_sql_secondary_final tab1
ON base.`year` = tab1.`year`+1
AND tab1.country = 'Czech Republic' AND base.unit != 'czk'
JOIN t_daniel_rizak_project_sql_secondary_final tab2
ON base.`year` = tab2.`year`
AND tab2.country = 'Czech Republic' AND base.unit != 'czk' AND
base.`year` > 2006;
```

Bází je primární tabulka s cenami všech potravin a na ni napojujeme sekundární tabulku s HDP. Nejdříve s podmínkou base.`year` = tab1.`year`+1, abychom u HDP dostali rok N-1, předcházející vývoji cen v roce N. Druhý Join páruje HDP a ceny ve stejném roce. Získané tabulky s indexy HDP (Y) a mezd (wage) resp. cen (price) využijeme dále.

Vyvíjí se růsty resp. poklesy HDP a mezd stejnosměrně nebo protichůdně? Stejný dotaz si pak položíme pro vztah produktu a cen.

```
SELECT trends1.*,
CASE
WHEN trends1.wage_grow_same_year > 1 AND
trends1.GDP_grow_prev_year > 1 THEN 1
WHEN trends1.wage_grow_same_year < 1 AND
trends1.GDP_grow_prev_year < 1 THEN 1
ELSE 0
END wages_and_Y_previous,
CASE
WHEN trends1.wage_grow_same_year > 1 AND
trends1.GDP_grow_same_year > 1 THEN 1
WHEN trends1.wage_grow_same_year < 1 AND
trends1.GDP_grow_same_year < 1 THEN 1
ELSE 0
END wages_and_Y_same
FROM (
```

```

SELECT base.country, tabl.`year`,
       tabl.value_current_year/tabl.value_prev_year AS wage_grow_same_year,
       base.GDP_current_year/base.GDP_previous_year AS GDP_grow_prev_year,
       LEAD(base.GDP_current_year,19)OVER(ORDER BY
       base.`year`)/LEAD(base.GDP_previous_year,19)OVER(ORDER BY base.`year`)
       AS GDP_grow_same_year, tabl.category_name
FROM t_daniel_rizak_project_sql_secondary_final base
JOIN t_daniel_rizak_project_sql_primary_final tabl
      ON base.`year`+1 = tabl.`year`
      AND base.country = 'Czech Republic' AND tabl.unit = 'czk'
) trends1

```

Ve druhém sloupci zprava “wages\_and\_Y\_previous” je 1 tam, kde HDP v předchozím roce a mzdy v běžném roce buď společně rostou nebo společně klesají. Tam kde je 0, se pohybují opačným směrem. Sloupec “wages\_and\_Y\_same” zobrazuje HDP a mzdy ve stejném roce. Pro shodný trend pohybu hodnot je opět 1, pro rozdílný trend je 0.

Ptáme se dále. Z celkového počtu případů, v kolika se vyvíjí stejnosměrně mzda a HDP?

```

SELECT sum(wages_and_Y_previous), sum(wages_and_Y_same), count(1) AS
       total_cases
FROM (
      SELECT trends1.*,
             CASE
               WHEN trends1.wage_grow_same_year >1 AND
               trends1.GDP_grow_prev_year >1 THEN 1
               WHEN trends1.wage_grow_same_year <1 AND
               trends1.GDP_grow_prev_year <1 THEN 1
               ELSE 0
             END wages_and_Y_previous,
             CASE
               WHEN trends1.wage_grow_same_year >1 AND
               trends1.GDP_grow_same_year >1 THEN 1
               WHEN trends1.wage_grow_same_year <1 AND
               trends1.GDP_grow_same_year <1 THEN 1
               ELSE 0
             END wages_and_Y_same
      FROM (
            SELECT base.country, tabl.`year`,
                   tabl.value_current_year/tabl.value_prev_year AS
                   wage_grow_same_year,
                   base.GDP_current_year/base.GDP_previous_year AS
                   GDP_grow_prev_year,
                   LEAD(base.GDP_current_year,19)OVER(ORDER BY
                   base.`year`)/LEAD(base.GDP_previous_year,19)OVER(ORDER BY
                   base.`year`) AS GDP_grow_same_year,
                   tabl.category_name
            FROM t_daniel_rizak_project_sql_secondary_final base
            JOIN t_daniel_rizak_project_sql_primary_final tabl
                  ON base.`year`+1 = tabl.`year`
                  AND base.country = 'Czech Republic' AND tabl.unit = 'czk'
            ) trends1
      ) frequency1

```

Celkový počet je ve sloupci “total\_cases”, 247 případů. Ve 195 případech z 247 (tj. shoda 79%) při růstu/poklesu produktu v předchozím roce rostly/poklesly v roce běžném také mzdy. V ostatních případech se proměnné vyvíjely opačně. O něco menší je stejnosměrný vývoj u vztahu produkt a mzdy ve stejném roce. Viz sloupec “sum(wages\_and\_Y\_same)” 176 případů z 247, tj. 71% shody. O silné korelaci mluvíme až od hodnot 0,8 a výše (tedy nad 80%). Prozatím můžeme ale říct, že pokud existuje závislost mezi HDP a mzdami, pak je větší mezi HDP v předchozím roce a mzdami v roce běžném než mezi HDP a mzdami ve stejném roce.

V kolika odvětvích z celkového počtu 19 se shodují trendy růstu/poklesu HDP a mezd v jednotlivých letech?  
Upravíme vnější Select a přidáme klauzuli Group By.

```
SELECT `year`, sum(wages_and_Y_previous), count(wages_and_Y_previous),
       sum(wages_and_Y_same), count(wages_and_Y_same)
FROM (
    SELECT trends1.*,
           CASE
               WHEN trends1.wage_grow_same_year >1 AND
                    trends1.GDP_grow_prev_year >1 THEN 1
               WHEN trends1.wage_grow_same_year <1 AND
                    trends1.GDP_grow_prev_year <1 THEN 1
               ELSE 0
           END wages_and_Y_previous,
           CASE
               WHEN trends1.wage_grow_same_year >1 AND
                    trends1.GDP_grow_same_year >1 THEN 1
               WHEN trends1.wage_grow_same_year <1 AND
                    trends1.GDP_grow_same_year <1 THEN 1
               ELSE 0
           END wages_and_Y_same
    FROM (
        SELECT base.country, tab1.`year`,
               tab1.value_current_year/tab1.value_prev_year AS
               wage_grow_same_year,
               base.GDP_current_year/base.GDP_previous_year AS
               GDP_grow_prev_year,
               LEAD(base.GDP_current_year,19)OVER(ORDER BY
               base.`year`)/LEAD(base.GDP_previous_year,19)OVER(ORDER BY
               base.`year`) AS GDP_grow_same_year,
               tab1.category_name
        FROM t_daniel_rizak_project_sql_secondary_final base
        JOIN t_daniel_rizak_project_sql_primary_final tab1
        ON base.`year`+1 = tab1.`year`
        AND base.country = 'Czech Republic' AND tab1.unit = 'czk'
    ) trends1
) frequencyl
GROUP BY year
```

#### Odpověď – HDP a mzdy:

Jak vidíme v prvních dvou sloupcích, v obdobích 2006-2009, 2011-2012 a 2015-2018 můžeme říci, že HDP v předcházejícím roce a mzdy v běžném roce oboje stejným směrem rostly nebo klesaly. V letech 2010 a 2014 tomu bylo ve většině případů naopak a v r. 2013 to nelze říct jednoznačně.

Nemůžeme tedy tvrdit, že při rostoucím HDP v předchozím roce rostly mzdy v roce následujícím. Tato závislost se z dostupných dat neprokázala.

Při hodnocení závislosti HDP a mezd ve stejném roce je viditelná stejnosměrnost v letech 2006-2008, 2010-2011 a 2014-2017. Ve třech rocích 2009, 2012 a 2018 se mzdy vyvíjely přesně opačně než produkt.

Nemůžeme tedy ani tvrdit, že při rostoucím HDP v běžném roce rostly také mzdy. Tato závislost se z dostupných dat neprokázala.

Totéž co pro mzdy zjistíme ještě pro vývoj cen. Dotaz z konce strany 12 upravíme pro ceny potravin:

```
SELECT trends2.*,
       CASE
           WHEN trends2.price_grow_same_year >1 AND
                trends2.GDP_grow_prev_year >1 THEN 1
           WHEN trends2.price_grow_same_year <1 AND
                trends2.GDP_grow_prev_year <1 THEN 1
       END
```

```

        ELSE 0
    END prices_and_Y_previous,
CASE
    WHEN trends2.price_grow_same_year >1 AND
    trends2.GDP_grow_same_year >1 THEN 1
    WHEN trends2.price_grow_same_year <1 AND
    trends2.GDP_grow_same_year <1 THEN 1
    ELSE 0
END prices_and_Y_same
FROM (
    SELECT tab1.country, base.`year`,
    base.value_current_year/base.value_prev_year AS
    price_grow_same_year,
    tab1.GDP_current_year/tab1.GDP_previous_year AS
    GDP_grow_prev_year,
    tab2.GDP_current_year/tab2.GDP_previous_year AS
    GDP_grow_same_year, base.category_name, base.region_name
    FROM t_daniel_rizak_project_sql_primary_final base
    JOIN t_daniel_rizak_project_sql_secondary_final tab1
        ON base.`year` = tab1.`year`+1
        AND tab1.country = 'Czech Republic' AND base.unit != 'czk'
    JOIN t_daniel_rizak_project_sql_secondary_final tab2
        ON base.`year` = tab2.`year`
        AND tab2.country = 'Czech Republic' AND base.unit != 'czk'
        AND base.`year` >2006
) AS trends2

```

Ve druhém sloupci zprava “prices\_and\_Y\_previous” je 1 tam, kde HDP v předchozím roce a ceny v roce běžném buď společně rostly nebo společně klesaly. Tam kde je 0, se pohybovaly protisměrně. Sloupec úplně v pravo “prices\_and\_Y\_same” zobrazuje HDP a ceny ve stejném roce. Pro shodný trend pohybu hodnot platí opět 1, pro rozdílný trend platí 0.

Z celkového počtu případů, v kolika se vyvíjí stejnosměrně ceny a HDP? Dotaz, který byl pro mzdy, upravíme pro ceny:

```

SELECT sum(prices_and_Y_previous), sum(prices_and_Y_same), count(1) AS
    total_cases
FROM (
    SELECT trends2.*,
        CASE
            WHEN trends2.price_grow_same_year >1 AND
            trends2.GDP_grow_prev_year >1 THEN 1
            WHEN trends2.price_grow_same_year <1 AND
            trends2.GDP_grow_prev_year <1 THEN 1
            ELSE 0
        END prices_and_Y_previous,
        CASE
            WHEN trends2.price_grow_same_year >1 AND
            trends2.GDP_grow_same_year >1 THEN 1
            WHEN trends2.price_grow_same_year <1 AND
            trends2.GDP_grow_same_year <1 THEN 1
            ELSE 0
        END prices_and_Y_same
    FROM (
        SELECT tab1.country, base.`year`,
        base.value_current_year/base.value_prev_year AS
        price_grow_same_year,
        tab1.GDP_current_year/tab1.GDP_previous_year AS
        GDP_grow_prev_year,
        tab2.GDP_current_year/tab2.GDP_previous_year AS
        GDP_grow_same_year, base.category_name, base.region_name
    ) AS trends2
) AS trends2

```



```

FROM t_daniel_rizak_project_sql_primary_final base
JOIN t_daniel_rizak_project_sql_secondary_final tab1
    ON base.`year` = tab1.`year`+1
    AND tab1.country = 'Czech Republic' AND base.unit != 'czk'
JOIN t_daniel_rizak_project_sql_secondary_final tab2
    ON base.`year` = tab2.`year`
    AND tab2.country = 'Czech Republic' AND base.unit != 'czk'
    AND base.`year` >2006
) AS trends2
) frequency2

```

Celkový počet měření je v pravém sloupci “total\_cases”, 4424 případů. Ve 2524 případech (tj. 57%) při růstu/poklesu produktu v předchozím roce (N-1) rostly/poklesly také ceny v roce běžném (N). V ostatních případech se proměnné vyvíjely opačně. Mírně menší je stejnosměrný vývoj u vztahu produkt a mzdy ve stejném roce. Viz sloupec “sum(prices\_and\_Y\_same)” 2390 případů ze 4424, tj. shoda 54%. To je poměrně slabá shoda. O silné korelaci mluvíme až od hodnot 0,8 a výše (tzn. nad 80%). Budeme zkoumat dále, abychom dostali odpověď.

V kolika případech tvořených počtem druhů potravin krát počet krajů se shodují trendy růstu/poklesu HDP a cen v jednotlivých letech? Do roku 2014 je to 14 krajů krát 26 druhů tzn. 364 případů, od roku 2015 přibýlo bílé víno, tedy 378 případů. Upravíme vnější Select a přidáme Group By.

```

SELECT `year`, sum(prices_and_Y_previous), count(prices_and_Y_previous),
       sum(prices_and_Y_same), count(prices_and_Y_same)
FROM (
    SELECT trends2.*,
           CASE
               WHEN trends2.price_grow_same_year >1 AND
                    trends2.GDP_grow_prev_year >1 THEN 1
               WHEN trends2.price_grow_same_year <1 AND
                    trends2.GDP_grow_prev_year <1 THEN 1
               ELSE 0
           END prices_and_Y_previous,
           CASE
               WHEN trends2.price_grow_same_year >1 AND
                    trends2.GDP_grow_same_year >1 THEN 1
               WHEN trends2.price_grow_same_year <1 AND
                    trends2.GDP_grow_same_year <1 THEN 1
               ELSE 0
           END prices_and_Y_same
    FROM (
        SELECT tab1.country, base.`year`,
               base.value_current_year/base.value_prev_year AS
               price_grow_same_year,
               tab1.GDP_current_year/tab1.GDP_previous_year AS
               GDP_grow_prev_year,
               tab2.GDP_current_year/tab2.GDP_previous_year AS
               GDP_grow_same_year, base.category_name, base.region_name
        FROM t_daniel_rizak_project_sql_primary_final base
        JOIN t_daniel_rizak_project_sql_secondary_final tab1
            ON base.`year` = tab1.`year`+1
            AND tab1.country = 'Czech Republic' AND base.unit != 'czk'
        JOIN t_daniel_rizak_project_sql_secondary_final tab2
            ON base.`year` = tab2.`year`
            AND tab2.country = 'Czech Republic' AND base.unit != 'czk'
            AND base.`year` >2006
        ) AS trends2
    ) frequency2
GROUP BY year

```

### Odpověď – HDP a ceny:

V rocích 2007, 2008, 2011, 2012 a 2017 vidíme, že většina cen (68%-87%) se pohybovala stejně s HDP v roce předcházejícím. V letech 2009 a 2013 byl trend přesně opačný. U roků 2010, 2014-2016 a 2018 je to nejednoznačné (shoda 49%-59%).

Když hodnotíme závislost HDP a cen v témže roce, pak je nejviditelnější stejnosměrný vývoj v letech 2007-2009, 2011 a 2017 (od 74 do 79% případů). V období 2012-2013 se ceny vyvíjely většinou jinak než produkt a v ostatních letech různými směry, shoda je jen v 41-59% případů.

Nemůžeme tedy tvrdit, že při rostoucím HDP v předchozím roce rostly ceny v roce následujícím. Neplatí ani, že při rostoucím HDP v určitém roce rostly také ceny ve stejném roce. Stejný závěr musíme konstatovat i pro pokles HDP a současný pokles cen. Tyto závislosti se z dostupných dat neprokázaly.

### Shrnutí průběhu výzkumu

Během výzkumu jsem nenarazil na nějaké zásadní překážky. Data zdrojových tabulek jsou vyčištěna a dobře zformátována. Malou záležitostí je u cen potravin položka Jakostní víno bílé, jehož cena byla měřena jen po dobu 4 let, tedy ne po celou dobu. S tím je potřeba počítat při využití window functions.

U dat týkajících se mezd bohužel nejsou informace o místech měření hodnot – okresy nebo kraje. To je škoda, protože bychom jinak mohli porovnávat mzdy a ceny přímo v regionech. Takhle jsme měli jen celonárodní hodnoty mezd.

Určitou překážkou je nefungující funkce CORR v Maria DB resp. v databázovém programu DBeaver. Pomocí ní bychom získali snadno hodnoty korelace a nemuseli bychom je obcházet složitější konstrukcí dotazů, viz str. 13-16. Příklad jak by takový dotaz s funkcí Corr mohl vypadat:

```
SELECT CORR(wage_grow_same_year, GDP_grow_prev_year) AS correlation
FROM (
    SELECT tab1.value_current_year/tab1.value_prev_year AS
wage_grow_same_year,
        base.GDP_current_year/base.GDP_previous_year AS GDP_grow_prev_year
    FROM t_daniel_rizak_project_sql_secondary_final base
    JOIN t_daniel_rizak_project_sql_primary_final tab1
        ON base.`year`+1 = tab1.`year`
        AND base.country = 'Czech Republic' AND tab1.unit IS NULL
) AS cor_check;
```

Znalosti načerpané během kurzu bohatě postačovaly pro vytvoření tabulek a vyhodnocení otázek tohoto projektu.