***Popis kódů***

***Kód na vytvoření primární tabulky:***

CREATE TABLE t\_barbora\_kalincova\_project\_sql\_primary\_final AS (

WITH EXTENDED\_CZECHIA\_PRICE AS (

SELECT \*,year(date\_from) AS year\_from, CASE WHEN MONTH(date\_from) BETWEEN 1 AND 3 THEN "1" WHEN MONTH(date\_from) BETWEEN 4 AND 6 THEN "2"

WHEN MONTH(date\_from) BETWEEN 7 AND 9 THEN "3" WHEN MONTH(date\_from) BETWEEN 10 AND 12 THEN "4" END AS "Quarter"

FROM czechia\_price

WHERE region\_code IS NULL

)

SELECT cp.value, cp.value\_type\_code, cpvt.name AS "VALUE\_TYPE\_NAME",cp.unit\_code, cpu.name AS "Unit\_code\_name", cp.calculation\_code, cpc.name AS "Calculation\_name",

cp.industry\_branch\_code , cpib.name AS "industry\_branch\_name", cp.payroll\_year, cp.payroll\_quarter, cpr.value AS "price", cpr.category\_code, cpcat.name, cpcat.price\_value AS "unit\_amount", cpcat.price\_unit

AS "unit", cpr.date\_from, cpr.date\_to

FROM czechia\_payroll cp

JOIN czechia\_payroll\_calculation cpc ON cp.calculation\_code = cpc.code

JOIN czechia\_payroll\_industry\_branch cpib ON cp.industry\_branch\_code = cpib.code

JOIN czechia\_payroll\_unit cpu ON cp.unit\_code = cpu.code

JOIN czechia\_payroll\_value\_type cpvt ON cp.value\_type\_code = cpvt.code

LEFT JOIN EXTENDED\_CZECHIA\_PRICE cpr ON cp.payroll\_year = cpr.year\_from AND cp.payroll\_quarter = cpr.Quarter

JOIN czechia\_price\_category cpcat ON cpr.category\_code = cpcat.code

)

Vytvořila jsem tabulku t\_barbora\_kalincova\_project\_sql\_primary\_final, v níž jsem spojila všechna data z primárních tabulek potřebné pro odpovědi na otázky. Problém byl v propojení tabulky czechia\_price a payroll, kde payroll má průměrné platy zadávané po čtvrtletích jednotlivých roků a ceny potravin jsou měřeny v krátkých intervalech v různých dobách roku. Propojení přímo na rok by tedy vyhodilo zbytečné a potenciálně nesmyslné údaje, kdy pro každé měření by se jeho výsledek aplikoval pro každé čtvrtletí. Tedy jsem vytvořila temporary tabulku EXTENDED\_CZECHIA\_PRICE, kde jsem určila rok na základě sloupce “date\_from” a čtvrtletí na základě měsíce ze sloupce “date\_from”. První čtvrtletí je definováno rozmezím měsíců 1-3, druhé rozmezím měsíců 4-6 atd. Tabulky jsem následně propojila dvojitou vazbou na rok v tab. Payroll a rok v EXTENDED\_CZECHIA\_PRICE a zároveň čtvtrletí v tab. Payroll a čtvrtletí, které jsem si vytvořila v tabulce EXTENDED\_CZECHIA\_PRICE. Zbývající tabulky jsem propojila jednoduše přes vazbu foreign to primary key.

**Kód na vytvoření sekundární tabulky:**

create table t\_barbora\_kalincova\_project\_SQL\_secondary\_final as (

select c.country, e.year, e.gdp, e.population,e.gini, e.taxes, e.fertility, e.mortaliy\_under5, c.abbreviation , c.avg\_height , c.calling\_code , c.capital\_city , c.continent, c.currency\_name ,c.religion ,

c.currency\_code, c.domain\_tld , c.elevation , c.north , c.south , c.west , c.east , c.government\_type , c.independence\_date ,

c.iso\_numeric , c.landlocked , c.life\_expectancy , c.national\_symbol , c.national\_dish , c.population\_density ,

c.population as "population\_in\_2018", c.region\_in\_world , c.surface\_area , c.yearly\_average\_temperature , c.median\_age\_2018,

c.iso2 , c.iso3

from countries c

left join economies e on c.country = e.country

UNION

select e.country, e.year, e.gdp, e.population,e.gini, e.taxes, e.fertility, e.mortaliy\_under5, c.abbreviation , c.avg\_height , c.calling\_code , c.capital\_city , c.continent, c.currency\_name ,c.religion ,

c.currency\_code, c.domain\_tld , c.elevation , c.north , c.south , c.west , c.east , c.government\_type , c.independence\_date ,

c.iso\_numeric , c.landlocked , c.life\_expectancy , c.national\_symbol , c.national\_dish , c.population\_density ,

c.population as "population\_in\_2018", c.region\_in\_world , c.surface\_area , c.yearly\_average\_temperature , c.median\_age\_2018,

c.iso2 , c.iso3

from economies e

left join countries c on e.country = c.country

)

Secondary tabulku jsem vytvořila propojením tabulek countries and economies. Po prvotním ozkoušení propojení přes left join jsem zjistila, že každá tabulka obsahuje názvy zemí (anebo úzmních celků), které nejsou obsažené v připojované tabulce. Abych tedy dosáhla toho, že nebudou chybět žádné záznamy ani z jedné tabulek, udělala jsem tedy dva selecty, kdy v jednom byla výchozí tabulka economies a propojila jsem ji s countries přes left join a společný prvek country. Následně jsem vytvořila druhý dotaz, kde jsem propojila výchozí tabulku countries s tabulkou economies opět přes left join a společný prvek country. Výyledek těchto dvou dotázů jsem spojila unionem. A nahrála do nové tabulky.

**Kód k získání odpovědi na první otázku:**

WITH avg\_salaries\_per\_year\_industry AS (

SELECT

payroll\_year,

industry\_branch\_name,

AVG(value) AS avg\_salary

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

VALUE\_TYPE\_NAME = 'Průměrná hrubá mzda na zaměstnance'

GROUP BY

payroll\_year, industry\_branch\_name

),

yoy\_changes AS (

SELECT

payroll\_year,

industry\_branch\_name,

avg\_salary,

LAG(avg\_salary) OVER (PARTITION BY industry\_branch\_name ORDER BY payroll\_year) AS prev\_year\_salary,

(avg\_salary - LAG(avg\_salary) OVER (PARTITION BY industry\_branch\_name ORDER BY payroll\_year)) / LAG(avg\_salary) OVER (PARTITION BY industry\_branch\_name ORDER BY payroll\_year) \* 100 AS yoy\_percent\_change

FROM

avg\_salaries\_per\_year\_industry

),

industry\_status AS (

SELECT

industry\_branch\_name,

MIN(yoy\_percent\_change) AS min\_yoy\_change,

MAX(yoy\_percent\_change) AS max\_yoy\_change

FROM

yoy\_changes

GROUP BY

industry\_branch\_name

)

SELECT

industry\_branch\_name,

CASE

WHEN min\_yoy\_change >= 0 THEN 'Increasing'

ELSE 'Decreasing at times'

END AS salary\_trend

FROM

industry\_status;

Nejprve jsem si vytvořila temp. tabulku avg\_salaries\_per\_year\_industry, kde jsem si zadefinovala průměr platu pro každé odvětví v každém roce. Následně jsem zadefinovala další temp. tabulku, kde jsem meziroční změnu (year on year = yoy) platu v každém obodbí. Z těchto meziročních změn jsem pak zjistila minimální a maximální meziroční změnu a na základě té pak v selectu určila zda li vždy platy rostou nebo i klesají.

***Kód k získání odpovědi na druhou otázku:***

WITH period\_bounds AS (

SELECT

GREATEST(MIN(payroll\_year), MIN(EXTRACT(YEAR FROM *date\_from*))) AS first\_year,

LEAST(MAX(payroll\_year), MAX(EXTRACT(YEAR FROM date\_from))) AS last\_year

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

),

avg\_salary AS (

SELECT

payroll\_year,

AVG(value) AS average\_salary

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

VALUE\_TYPE\_NAME = 'Průměrná hrubá mzda na zaměstnance'

GROUP BY

payroll\_year

HAVING

payroll\_year IN (SELECT first\_year FROM period\_bounds UNION SELECT last\_year FROM period\_bounds)

),

prices AS (

SELECT

EXTRACT(YEAR FROM date\_from) AS price\_year,

name AS item,

AVG(price) AS price,

unit AS unit

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

name IN ('Mléko polotučné pasterované', 'Chléb konzumní kmínový') AND

EXTRACT(YEAR FROM date\_from) IN (SELECT first\_year FROM period\_bounds UNION SELECT last\_year FROM period\_bounds)

GROUP BY

EXTRACT(YEAR FROM date\_from),

name,

unit

)

SELECT

s.payroll\_year,

p.item,

round(s.average\_salary / p.price,1) AS quantity,

p.unit

FROM

avg\_salary s

JOIN

prices p

ON

s.payroll\_year = p.price\_year

ORDER BY

payroll\_year,

item;

Nejdřív jsem v temp. tabulce period\_bounds zjistila nejmenší a největší společný rok v datech pro platy a ceny zboží. Následně jsem počítala průměrnou hrubou mzdu pro tyto roky. Dalším krokem bylo (temp. tabulka prices) spočítat průmrné ceny zboží pro mléko a chléb. S daty, která jsem takto získala v temporary tabulkách už můžeme definovat query, která bude obsahovat výsledek ve fromě zaoukrouhleného podílu průměrných platů a průmerných cen, seřazených dle roku a potravin.

***Kód k získání odpovědi na třetí otázku:***

WITH price\_data AS (

select

year(date\_from) AS price\_year,

name AS category\_name,

AVG(price) AS avg\_price

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

name IS NOT null

GROUP BY

price\_year, category\_name

),

yoy\_changes AS (

SELECT

price\_year,

category\_name,

avg\_price,

LAG(avg\_price) OVER (PARTITION BY category\_name ORDER BY price\_year) AS prev\_year\_price,

(avg\_price - LAG(avg\_price) OVER (PARTITION BY category\_name ORDER BY price\_year)) / LAG(avg\_price) OVER (PARTITION BY category\_name ORDER BY price\_year) \* 100 AS yoy\_percent\_change

FROM

price\_data

),

avg\_yoy\_changes AS (

SELECT

category\_name,

AVG(yoy\_percent\_change) AS avg\_yoy\_percent\_change

FROM

yoy\_changes

WHERE

prev\_year\_price IS NOT NULL to compare

GROUP BY

category\_name

)

SELECT

category\_name,

MIN(avg\_yoy\_percent\_change) AS slowest\_increase

FROM

avg\_yoy\_changes;

Prvním krokem bylo vytvoření temp. tabulky price\_data, kde jsem seřadila údaje o průměrných cenách jednotlivyých kategorií zboží podle roku a odfiltrovala data, kde nebyla určena kategorie zboží. V další temp. tabulce yoy\_changes jsem spočítala meziroční změny na základě dat z temp. tabulky price\_data. Z těchto změn jsem si zase spočítala jejich průměr v temp. tabulce avg\_yoy\_changes, kde jsem odfiltrovala řádky, kde chyběla data pro předchozí rok a seřadila dle kategorií zboží. Následně jsem selectem určila nejmenší průměrnnou meziroční změnu a označila ji jako nejpomalejší nárust ceny zboží. Zboží, kterého se to týká, je odpovědí na otázku.

***Kód k získání odpovědi na čtvrtou otázku:***

WITH price\_data AS (

SELECT

EXTRACT(YEAR FROM date\_from) AS price\_year,

AVG(price) AS avg\_price

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

price IS NOT NULL

GROUP BY

price\_year

),

yoy\_price\_changes AS (

SELECT

price\_year,

overall\_avg\_price,

prev\_year\_avg\_price,

CASE

WHEN prev\_year\_avg\_price IS NULL THEN NULL

ELSE (overall\_avg\_price - prev\_year\_avg\_price) / prev\_year\_avg\_price \* 100

END AS yoy\_price\_percent\_change

FROM (

SELECT

price\_year,

AVG(avg\_price) AS overall\_avg\_price,

LAG(AVG(avg\_price)) OVER (ORDER BY price\_year) AS prev\_year\_avg\_price

FROM

price\_data

GROUP BY

price\_year

) as sub

WHERE

prev\_year\_avg\_price IS NOT NULL

),

salary\_data AS (

SELECT

payroll\_year,

AVG(value) AS avg\_salary

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

VALUE\_TYPE\_NAME = 'Průměrná hrubá mzda na zaměstnance'

GROUP BY

payroll\_year

),

yoy\_salary\_changes AS (

SELECT

payroll\_year,

avg\_salary,

prev\_year\_avg\_salary,

CASE

WHEN prev\_year\_avg\_salary IS NULL THEN NULL

ELSE (avg\_salary - prev\_year\_avg\_salary) / prev\_year\_avg\_salary \* 100

END AS yoy\_salary\_percent\_change

FROM (

SELECT

payroll\_year,

avg\_salary,

LAG(avg\_salary) OVER (ORDER BY payroll\_year) AS prev\_year\_avg\_salary

FROM

salary\_data

) as sub

WHERE

prev\_year\_avg\_salary IS NOT NULL

)

SELECT

p.price\_year,

p.yoy\_price\_percent\_change,

s.yoy\_salary\_percent\_change,

p.yoy\_price\_percent\_change - s.yoy\_salary\_percent\_change AS difference,

CASE

WHEN (p.yoy\_price\_percent\_change - s.yoy\_salary\_percent\_change) > 10 THEN 'Significant Increase'

ELSE 'Normal'

END AS status

FROM

yoy\_price\_changes p

JOIN

yoy\_salary\_changes s

ON

p.price\_year = s.payroll\_year

ORDER BY

p.price\_year;

První temp. tabulka počítá průměr průměrné ceny všech potravin za rok. V další temp. tabulce jsem si spočítala meziroční změnu cen potravin, čímž jsem získala první datovou sadu k porovnání. V temp. tabulce salary\_data jsem si spočítala průměrnou hrubou mzdu za rok. Následně v temp tabulce yoy\_salary\_changes jsem si spočítala meziroční změnu průměrné mzdy. Výstupem pak byla query, ve které jsem si propojila tyto dvě temp. tabulky (yoy\_price\_changes a yoy\_salary\_changes) a vyselectovala rozdíl procentuální změny meziročního platu a meziroční ceny potravin. Pakliže byl někde rozdíl vyšší než 10 (procent), označila jsem to jako 'Significant Increase a zároveň je tento řádek odpovědí na výzkumnou otázku.

***Kód k získání odpovědi na pátou otázku:***

WITH gdp\_data AS (

SELECT

year gdp\_year,

gdp,

LAG(gdp) OVER (ORDER BY year) AS prev\_year\_gdp,

CASE

WHEN LAG(gdp) OVER (ORDER BY year) IS NULL THEN NULL

ELSE (gdp - LAG(gdp) OVER (ORDER BY year)) / LAG(gdp) OVER (ORDER BY year) \* 100

END AS yoy\_gdp\_percent\_change

FROM

t\_barbora\_kalincova\_project\_sql\_secondary\_final

WHERE

lower(country) like 'czech%'

GROUP BY

year

),

price\_data AS (

SELECT

YEAR(date\_from) AS price\_year,

AVG(price) AS avg\_price,

LAG(AVG(price)) OVER (ORDER BY YEAR(date\_from)) AS prev\_year\_avg\_price

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

price IS NOT null

GROUP BY

price\_year

),

yoy\_price\_changes AS (

SELECT

price\_year,

avg\_price,

prev\_year\_avg\_price,

CASE

WHEN prev\_year\_avg\_price IS NULL THEN NULL

ELSE (avg\_price - prev\_year\_avg\_price) / prev\_year\_avg\_price \* 100

END AS yoy\_price\_percent\_change

FROM

price\_data

WHERE

prev\_year\_avg\_price IS NOT NULL

),

salary\_data AS (

SELECT

payroll\_year,

AVG(value) AS avg\_salary,

LAG(AVG(VALUE)) OVER (ORDER BY payroll\_year) AS prev\_year\_avg\_salary

FROM

t\_barbora\_kalincova\_project\_sql\_primary\_final

WHERE

VALUE\_TYPE\_NAME = 'Průměrná hrubá mzda na zaměstnance'

GROUP BY

payroll\_year

),

yoy\_salary\_changes AS (

SELECT

payroll\_year,

avg\_salary,

prev\_year\_avg\_salary,

CASE

WHEN prev\_year\_avg\_salary IS NULL THEN NULL

ELSE (avg\_salary - prev\_year\_avg\_salary) / prev\_year\_avg\_salary \* 100

END AS yoy\_salary\_percent\_change

FROM salary\_data

WHERE

prev\_year\_avg\_salary IS NOT NULL

)

SELECT

g.gdp\_year,

g.yoy\_gdp\_percent\_change,

p.yoy\_price\_percent\_change,

s.yoy\_salary\_percent\_change,

g.yoy\_gdp\_percent\_change - p.yoy\_price\_percent\_change AS gdp\_vs\_price\_difference,

g.yoy\_gdp\_percent\_change - s.yoy\_salary\_percent\_change AS gdp\_vs\_salary\_difference

FROM

gdp\_data g

LEFT JOIN

yoy\_price\_changes p ON g.gdp\_year = p.price\_year

LEFT JOIN

yoy\_salary\_changes s ON g.gdp\_year = s.payroll\_year

where

g.yoy\_gdp\_percent\_change is not null and p.yoy\_price\_percent\_change is not null and s.yoy\_salary\_percent\_change is not null

ORDER BY

g.gdp\_year;

Vytvořila jsem temporary tabulku gdp\_data, kde jsem si zadefinovala GDP pro Českou republiku, GDP v předchozím roce pro ČR a jejich podíl v procentech. Následně jsem vytvořila druhou temporary tabulku, ve které jsem udělala to stejné pro průměrné ceny potravin. V temp. tabulce yoy\_price\_changesjsem si zadefinovala procentuální meziroční rozdíl v průměrných cenáchpotravin. Následně v tabulce salary\_data jsem si zadefinovala průměrný plat a průměrný plat z předchozího roku, abych jej použila temp. tabulce yoy\_salary\_changes, kde jsem vytvořila meziroční procentuální rozdíl v průměrných platech. Tyto temporary tabulky jsem následně propojila podle roku, a vytvořila řádky pro rozdíl změny GDP a cen potravin a zároveň rozdíl změny v GDP a platů. Na základě získaných údajů jsem nenašla žádnou korelaci mezi změnou GDP platů a cenou potravin.