

RĪGAS TEHNISKĀ UNIVERSITĀTE

Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte

5.praktiskais darbs

**Izvēlētas NewSQL datu bāzes sistēmas instalēšana un darbības izpēte**

Izstrādāja: Natans Šalamberidze

171RMC203

Pārbaudīja: lektors Jānis Eiduks

2023./2024. māc. Gads

# Darba mērķis

1. Iepazīties ar jaunākās datu bāzes sistēmu klases NewSQL tehnoloģijām.

2. Detalizēti praktiski izpētīt kādu NewSQL datu bāzes sistēmu.

# Darba uzdevums.

1. Izvēlēties NewSQL datu bāzes sistēmu un pamatot savu izvēli (kāpēc izvēlēta).

2. Iepazīties un apgūt sistēmas darbības uzbūvi (izpētīt dokumentāciju).

3. Izveidot sistēmas arhitektūras shēmu. Obligāti uzzīmēt to un iekļaut darba atskaitē (tā ir izpratnes pamats).

4. Veikt sistēmas instalēšanu un izveidot tā aprakstu.

5. Definēt vairākas (2) datu glabāšanas struktūras un veikt datu ievadi (katrā 5 datu rindas).

6. Izveidot 3 vidējas sarežģītības vaicājumus, veikt to ievadi un iegūt rezultātus.

7. Veikt analīzi, kas sistēmas darbībā ir labs un kādi ir trūkumi (salīdzinoši ar relāciju datu bāzi).

8. Noformēt atskaiti (teksti, rezultāti, attēli).

# NewSQL datubāzes izvēles pamatojums

Tika izvelēta Google Spanner NewSQL datubāze.

 **Mērogojamība un pieejamība**: Google Spanner ir īpaši pazīstama ar savu augsto mērogojamību un izturību. Tas spēj apkalpot ļoti lielas datu bāzes, kas izplatītas pāri dažādiem ģeogrāfiskajiem reģioniem, nodrošinot augstu pieejamību un izturību pret kļūmēm.

 **Spēcīga infrastruktūra**: Kā Google Cloud pakalpojums, Spanner izmanto Google vispasaules tīkla priekšrocības, piedāvājot vienu no labākajiem veiktspējas un uzticamības līmeņiem.

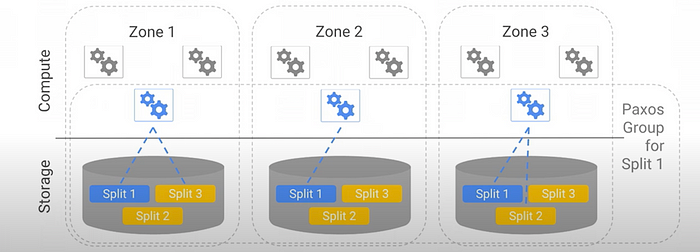
 **Globāla transakciju atbalsts**: Spanner nodrošina globālas ACID transakcijas, kas ir reti sastopamas citās datubāzēs, ļaujot jums veikt transakcijas, kas ir konsekventas pār vairākiem datu centriem. Tas ir ļoti svarīgi, ja jūsu lietojumprogramma prasa stingru datu konsekvenci.

 **Automātiska datu dublēšana un atjaunošana**: Google Spanner piedāvā automātisku datu dublēšanu un atjaunošanu, kas samazina riskus un administratīvo slogu.

 **SQL atbalsts**: Google Spanner atbalsta standarta SQL ar paplašinājumiem, kas atvieglo lietojumprogrammu izstrādi un integrāciju ar citiem rīkiem un pakalpojumiem.

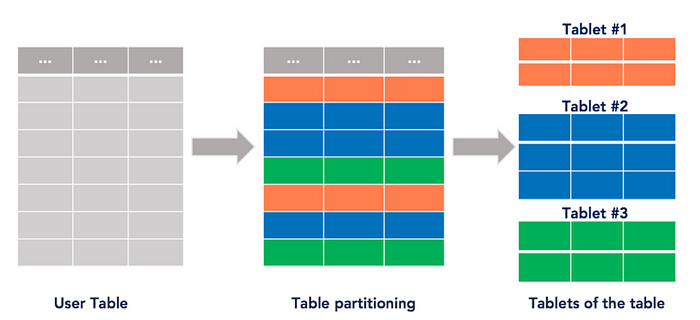
Google Spanner arhitektūra

Spanner ir datu bāze, kas datus sadala daudzās Paxos stāvokļa mašīnu kopās, kas atrodas datu centros visā pasaulē. Spanner augstākā līmeņa arhitektūra balstās uz diviem slāņiem: transakciju slāni un replikācijas slāni. Klients (tīmekļa pārlūks) sūta savu pieprasījumu sistēmas klientam (tīmekļa serverim), kurš ietver klienta bibliotēku. Tīmekļa serveris mijiedarbojas ar Spanner datu bāzi un tās serveru mezgliem. Zemāk redzamajā attēlā ir parādīts, kā uzglabāšanas vieta (Spanner datu bāze) un serveri (spanserveri) tiek ģeogrāfiski replikāti.

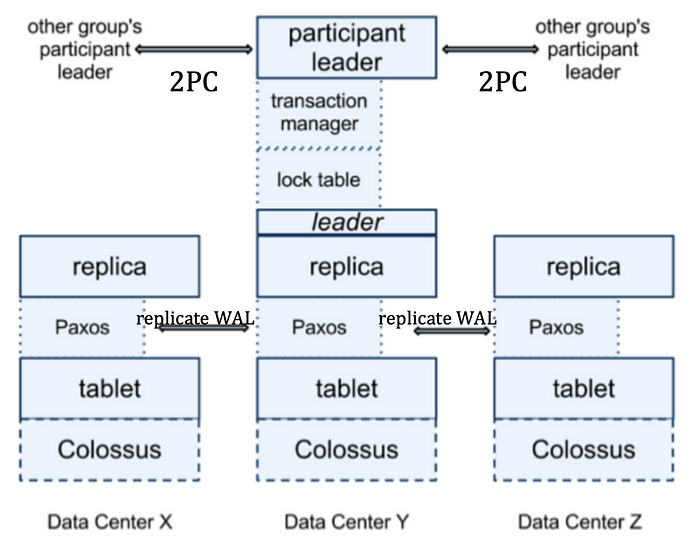


## Spanserveru arhitektūra:

Spanner datu bāze glabā vienu vai vairākas relacionālās tabulas. Tabulas ir sadalītas segmentos (tablets), un to stāvoklis tiek fiksēts diskā vai rakstīts priekšā logā (WAL) distribuētajā failu sistēmā, ko sauc par Colossus. Parasti katru segmentu pārvalda trīs segmentu serveri, un stāvoklis tiek replikāts starp zonām. Segmentu serveru kopu, kas replikā vēl vienu segmentu, sauc par Paxos grupu. Grupā viens no serveriem kādu laiku darbojas kā līderis, pārējie ir sekotāji.



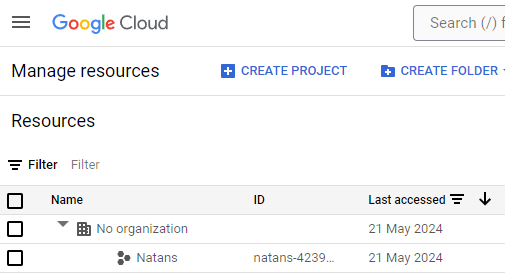
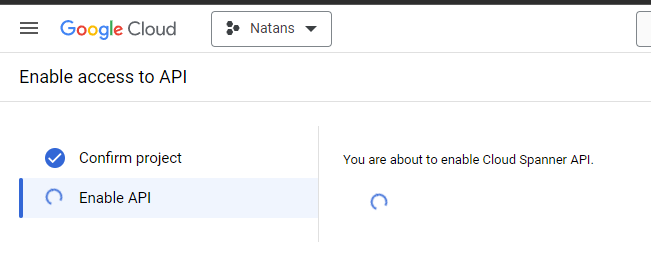
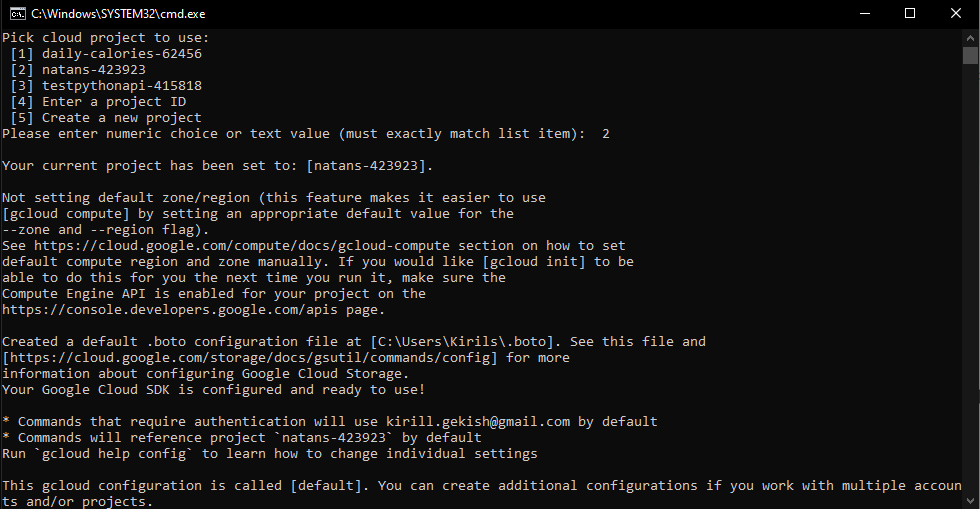
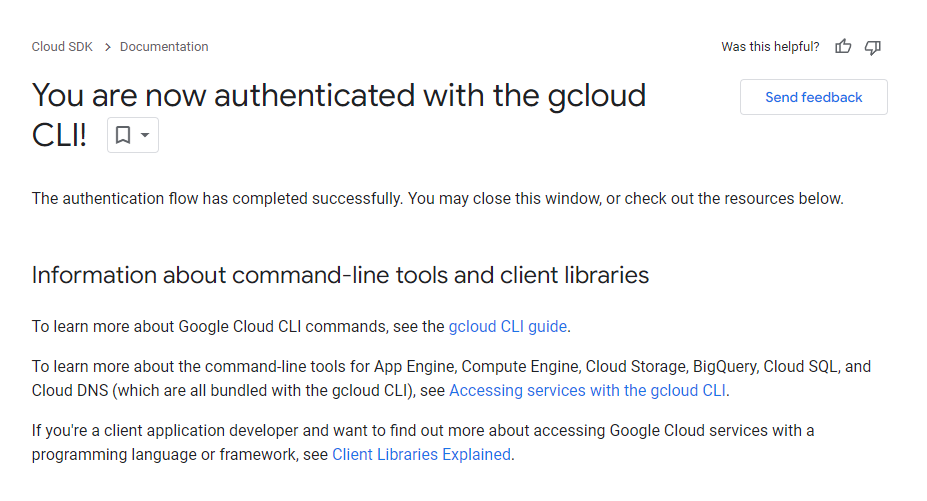
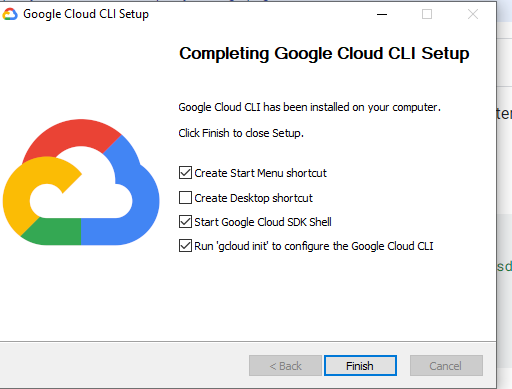
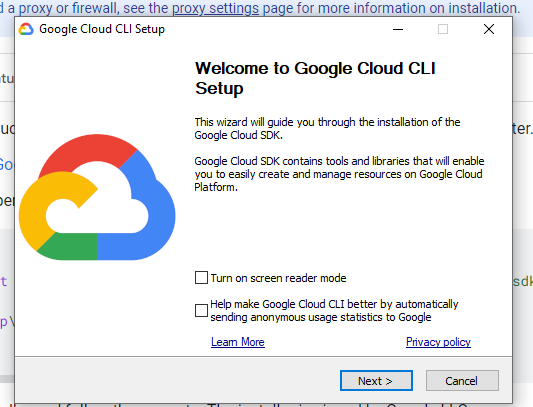
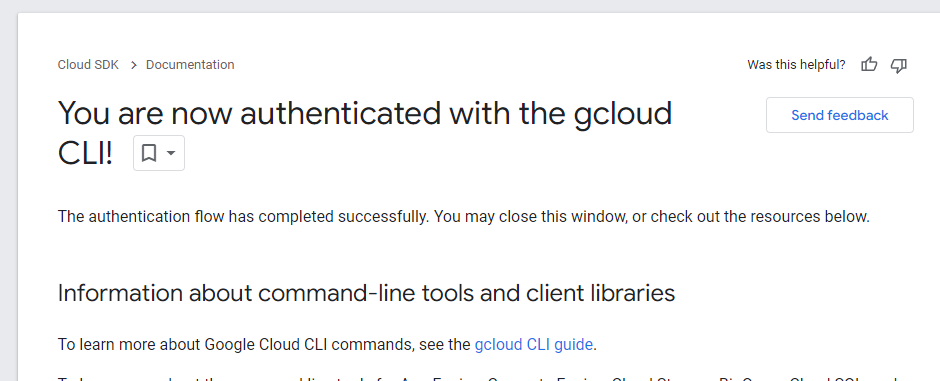
Līderis rūpējas par klientu operācijām, izmantojot bloķēšanu, lai nodrošinātu izolāciju no vienlaicīgām transakcijām un žurnāla ierakstiem. Starp segmentiem, repliku grupu līderi izmanto divpakāpju apņemšanos (2PC), lai saglabātu atomaritāti. Viena grupa ir koordinators, pārējie - dalībnieki. Ja koordinators kļūst nereaģējošs, sistēma var apstāties. Tomēr 2PC tiek izmantots kopā ar Paxos Spanner. Katras grupas līderis ir atbalstīts ar daudziem sekotājiem, tādēļ jebkurš sekotājs var kļūt par līderi.

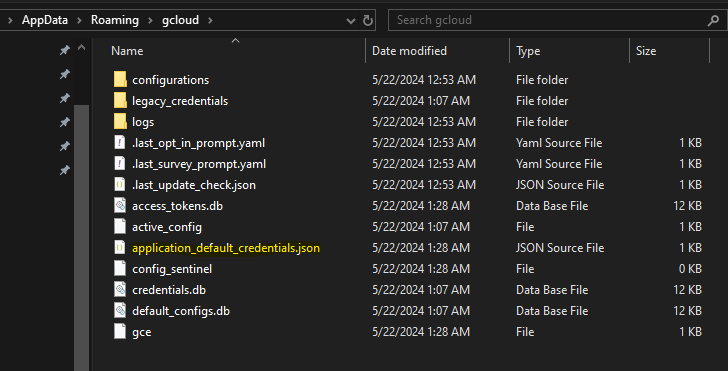


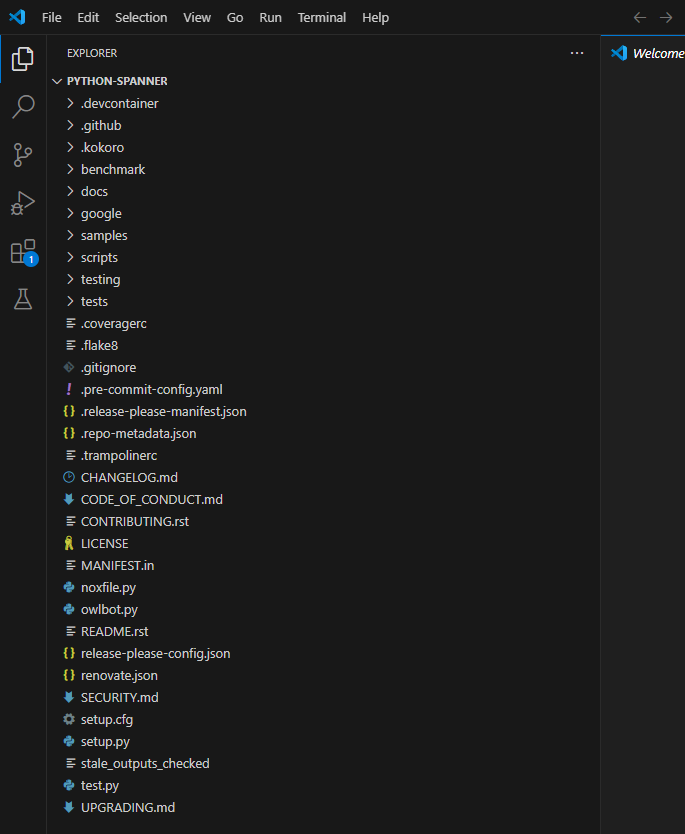
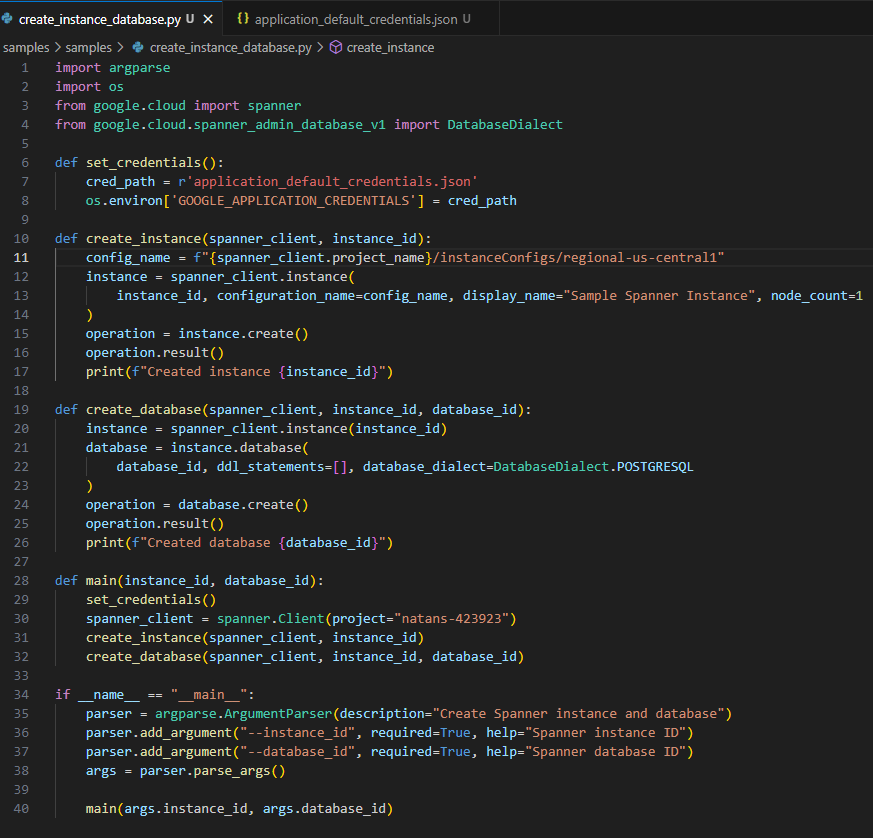
Spanner piedāvā arī citas dizaina izvēles, piemēram, direktoriju. Direktorija ir datu izvietojuma vienība (ģeogrāfiskās replikācijas īpašības), kas pārvalda replikāciju un lokalitāti. Kad dati tiek pārvietoti starp Paxos grupām, tie tiek pārvietoti direktoriju pa direktorijai. 50MB direktoriju var pārvietot dažu sekunžu laikā.

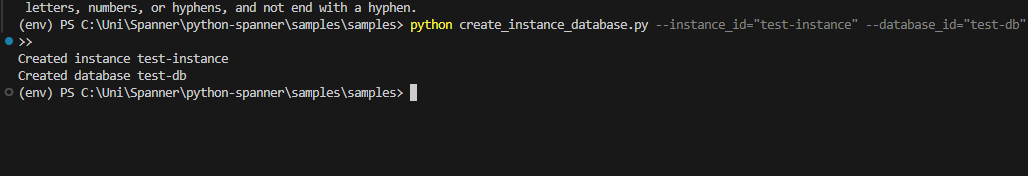
Spanner datu modelī ir arī izveidota noderīga vecāku-bērnu attiecību struktūra, ko sauc par interleave, lai uzlabotu sistēmas veiktspēju un datu glabāšanu. Interleave ļauj Spanner glabāt vecāku un bērnu rindas kopā. Piemēram, klientu un rēķinu tabulas. Ja lietojumprogramma bieži pieprasa klientu tabulas ar visām tās rēķinu tabulām, ir jādefinē rēķinu tabula kā iekļauta bērna tabula klientiem. Tas nozīmē, ka Spanner glabās klientu rindas kopā ar vienu vai vairākām rēķinu rindām.

Google Spanner instalācija

1. Jāizveido projektu Google Cloud platformā. 
2. Jāieslēdz Google Spanner API 
3. Jāinstalē Google Cloud CLI. Instalācijas failā var izvelēties automātisko komandas “gcloud init” palaišanu. Ir tikai jāizvēlas projektu.
4. Jāpeirākstas Google Cloud CLI, palaidot komandu “gcloud auth application-default login”. Rezultāta tiek izveidots application\_default\_credentials.json fails.



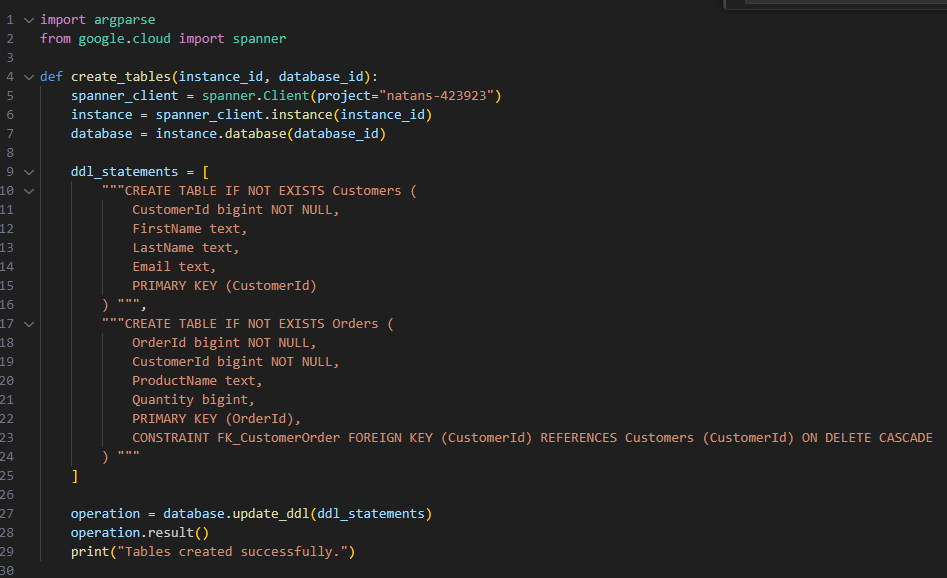
1. Tika izvelēta Python valoda darbībam ar Google Spanner datubāzi. Jānokopē Git repozitoriju ar Google Spanner 
2. Tika izveidotā atsevišķa vidē, kur tika ieinstalēti visas bibliotēkas, kas ir vajadzīgas Google Spanner darbībai. Turpat tika pārnests iepriekš ģenerēts fails. Tika izveidots Python skripts, kas izveido Google Spanner instanci un datubāzi tajā.

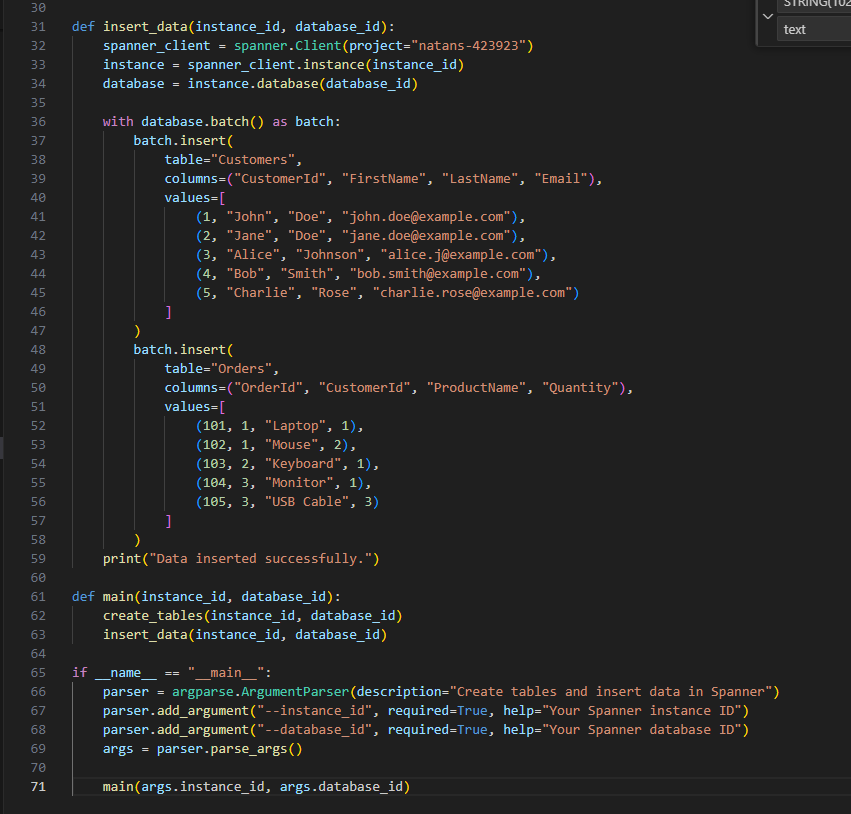


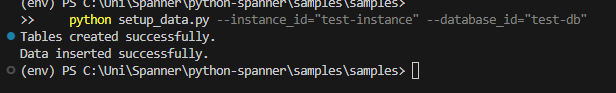
# Tabulu definēšana un aizpildīšana

Tabulu definēšenai tika izveidots Python scripts, kur tika noteikti DDL vaicājumi, kas tika nodoti Google Spanner API un izpildīti.

Arī tika veikta batch insert metode tabulu aizpildīšanai.

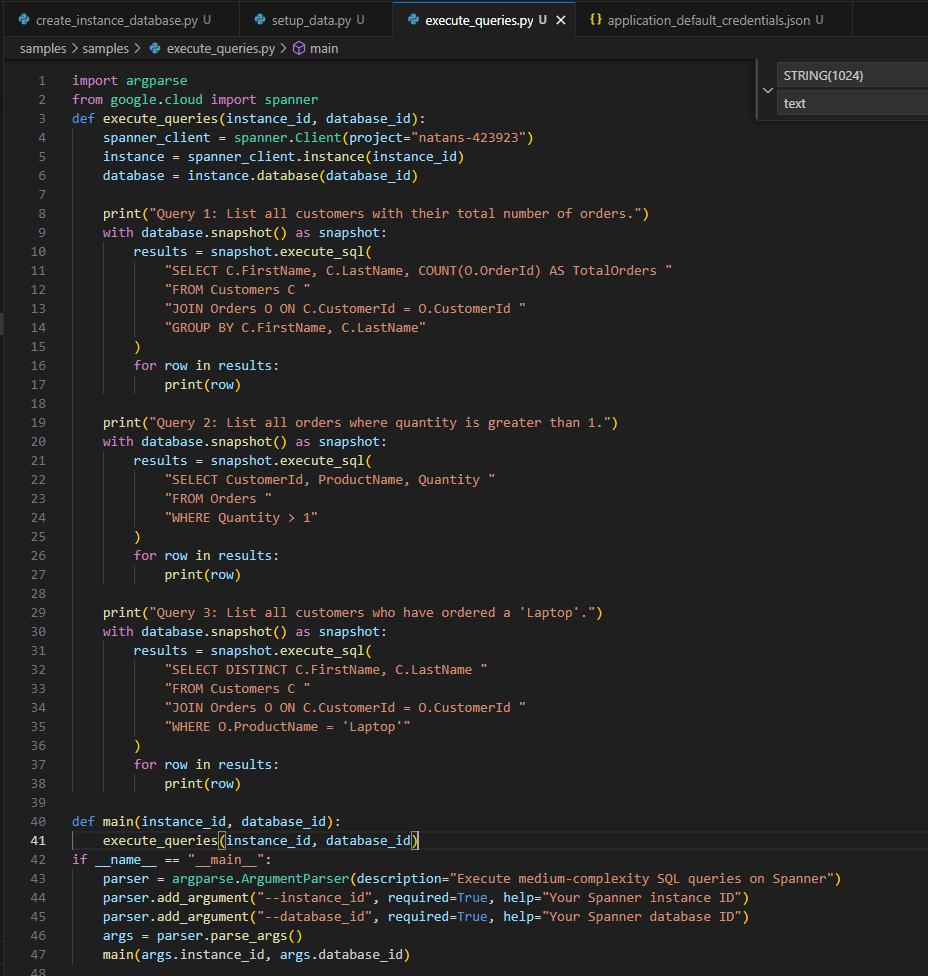


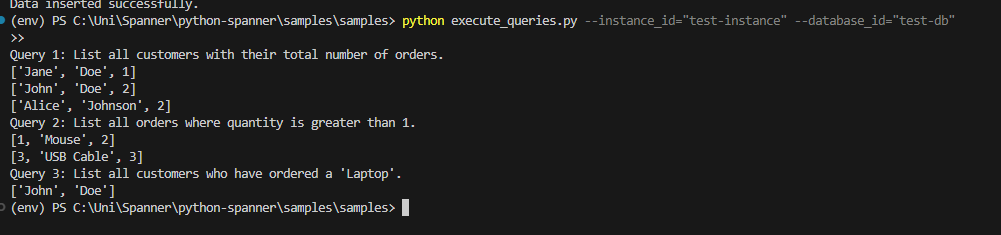




# Vaicājumu izpilde

Vaicājumu izpildei tika veidots Python skripts, kur tika definēti vaicājumi, kas tika nodoti Google Spanner un izpildīti. Vaicājumu rezultāts tika paradīts konsolē.





# Google Spanner datubāzes analīze

## Google Cloud Spanner stiprās puses:

1. **Mērogojamība un izturība:** Google Cloud Spanner piedāvā horizontālu mērogojamību, kas ļauj datu bāzei augt un apstrādāt lielu datu un lietotāju apjomu bez ievērojama veiktspējas samazinājuma. Tas nodrošina arī augstu pieejamību un izturību, kas ir būtiski kritisku sistēmu darbībai.
2. **Globāla transakciju konsistence:** Atšķirībā no daudzām NoSQL datu bāzēm, Spanner nodrošina stingru ACID transakciju atbalstu, arī globālā mērogā, kas garantē datu konsekvenci visos reģionos.
3. **Automātiska pārvaldība:** Spanner samazina nepieciešamību pēc manuālas datu bāzes optimizācijas un pārvaldības, piedāvājot automātiskas atjaunināšanas un mērogošanas iespējas.

## Google Cloud Spanner trūkumi:

1. **Izmaksas:** Spanner var būt salīdzinoši dārgs risinājums, it īpaši mazākiem projektiem vai organizācijām, kurām nav nepieciešama tā piedāvātā mērogojamība un izturība.
2. **Kompleksitāte:** Lai gan daudzas operācijas ir automatizētas, Spanner sistēmas uzturēšana un optimizācija var prasīt dziļākas zināšanas un pieredzi ar mākoņdatošanas platformām, kas var būt šķērslis mazāk pieredzējušiem lietotājiem.
3. **SQL atbalsts:** Lai gan Spanner atbalsta SQL, tas ietver dažas specifiskas izmaiņas un ierobežojumus, kas var neļaut pilnībā izmantot visus SQL standarta piedāvātos iespējas.

## Salīdzinājums ar tradicionālām relāciju datu bāzēm:

Tradicionālās relāciju datu bāzes, piemēram, PostgreSQL vai MySQL, ir labi zināmas ar plašu SQL atbalstu, plašām funkcionalitātēm un stabilu veiktspēju. Tās bieži izvēlas gadījumos, kad ir nepieciešama precīza datu konsistence un transakciju integritāte lokālā vai privatizētā vidē, bet tās var sastapties ar mērogojamības un izturības ierobežojumiem liela apjoma un ģeogrāfiski izkliedētu datu apstrādē.

Izvēloties starp Google Cloud Spanner un tradicionālām relāciju datu bāzēm, ir svarīgi ņemt vērā konkrētas biznesa vajadzības, datu apstrādes prasības, un budžeta ierobežojumus. Spanner ir piemērots lielām, globāli izkliedētām aplikācijām, kam nepieciešama augsta veiktspēja un konsistence reāllaikā, savukārt tradicionālās SQL datu bāzes ir labāk piemērotas darbam ar kompleksām transakcijām un ziņojumu apstrādi ar zemākām izmaksām un vienkāršāku pārvaldību.