Engineer

SQL

19 resultat

Koppla samman systemets delar

Arbetet under ”SQL” visar en del av ett fungerande system. Vi kan hjälpa er att koppla den delen till resten av verksamheten. Hittar databasfrågorna och de saknade indexen bakom ett system som har blivit tyngre för varje år, och åtgärdar dem som spelar roll.

Se våra tjänster Hör av er →

Hitta en jämförbar utmaning

Öppna ett exempel som liknar er situation. Genomgången förklarar besluten bakom resultatet, utöver teknikens namn. 30 dagars eldata på 180 ms — 180 ms för att aggregera en månads mätaravläsningar

30 dagars eldata på 180 ms Hör av er →

30 dagars eldata på 180 ms

Utvecklade varianter för timvis aggregering av elförbrukning i Python, SQL och Bash + Jq; Python-varianten behandlade en JSONL-datamängd för 30 dagar på 180 ms i prestandatestet.

Gjorde pipelinen för geodata 50 gånger snabbare

Dokumenterade en förbättring på 50 gånger i en geografisk datapipeline i grundarens arbete hos Utiligize genom SQL- och datamodelländringar i PostgreSQL, MS SQL och Google Cloud BigQuery.

Gjorde GIS-kartan 10 gånger snabbare i PostgreSQL

Förbättrade prestandan i en kartapplikation för geodata 10 gånger genom en strategisk databasövergång från MSSQL till PostgreSQL, inklusive själva omställningen av data, och optimerade bearbetningen och datasäkerheten.

Löste 1 000 problem i geodata och tidsseriedata

Löste 1 000 problem i geodata och tidsseriedata med GDAL, ArcGIS, PostGIS, Mapbox, QGIS, SQL (PL/pgSQL, Transact-SQL) och Bash, vilket säkrade en högkvalitativ bearbetning av stora datamängder.

Utformade och drev 6 ETL/ELT-pipelines

Utformade, implementerade och administrerade 6 ETL/ELT-pipelines med Google BigQuery, MSSQL, PostgreSQL, Shell-skript, PL/pgSQL och Transact-SQL och integrerade data för effektiv bearbetning i ett Python-API.

Publicerade 500 rapporter om el- och GIS-data

Publicerade 500 analysrapporter om el- och GIS-data och använde grundlig research och felsökning för att säkerställa korrekta insikter ur stora mängder geodata och tidsseriedata.

Drev 100 datalager i PostgreSQL, MS SQL, BigQuery

Utformade, byggde upp och förvaltade 100 datalagerdatabaser i PostgreSQL, MS SQL och Google BigQuery, främst med GIS- och tidsseriedata, och optimerade prestanda och skalbarhet.

Gjorde PostgreSQL 10 gånger snabbare

Gjorde PostgreSQL 10 gånger snabbare med strategisk indexering, partitionering och optimering av databasfrågor och effektiviserade databasen för användar- och tenantdata, geospatiala data och tidsseriedata om el.

Automatiserade GIS-driftsättning i GitHub Actions

Automatiserade driftsättningen av en GIS-applikation som SaaS, databearbetningen och rapporteringssystemet med GitHub Actions CI/CD, Python, Bash och SQL.

Ledde över 30 GIS-projekt från bygge till support

Ledde utveckling, driftsättning och support av över 30 GIS-projekt och visade expertis inom PostgreSQL, Bash, Python, JavaScript, GDAL, ArcGIS, PostGIS och Mapbox.

Automatiserade databearbetning i Python, PL/pgSQL

Automatiserade uppgifter inom databearbetning med Shell-skript, PL/pgSQL, Python och Transact-SQL och ökade produktiviteten och effektiviteten.

Byggde 600 ETL/ELT-pipelines i PL/pgSQL

Byggde 600 PL/pgSQL-baserade ETL/ELT-pipelines för att effektivisera komplexa arbetsflöden för databearbetning i flera utvecklings- och produktionsmiljöer för PostgreSQL.

Utformade en JSON-väg rakt från Postgres till API

Utformade en arkitektur för direkt vidarebefordran av JSON, där PostgreSQL-funktioner returnerar komplett JSON som Go-API:t skickar vidare ordagrant, vilket tog bort mellanliggande deserialisering och frikopplade frontenden från schemaändringar.

Utformade ett funktionsbaserat skikt i PostgreSQL

Utformade ett funktionsbaserat dataskikt i PostgreSQL — 1 275 lagrade funktioner i 34 scheman — så att varje läsning och skrivning går via en funktion som databasen kan ge behörighet till, i stället för via en tabell.

Modellerade 348 tabeller i 34 Postgres-scheman

Modellerade sjöfartsdomänen i 348 normaliserade tabeller i 34 PostgreSQL-scheman — yrkesverksamma, företag, fartyg, jobb, recensioner och resten — med SMALLINT-uppslag och UUID v7-nycklar.

Byggde egen analys inuti PostgreSQL

Byggde egen produktanalys i PostgreSQL — 47 funktioner över partitionerade händelsetabeller som rensar sig själva — pseudonymiserad bakom ett roterande salt och villkorad av besökarens samtycke.

Byggde en lojalitetsplånbok, ligor och märken

Byggde lojalitets- och ryktessystemet — 67 funktioner över en huvudbok i 31 tabeller, med ligor, märken och en inlösenbutik — som tar ett radlås på saldot för att stänga fönstret för dubbelspendering.

Genererade 495 sidor från en SQLite-export

Genererade 495 resultatsidor på tre språk från en skrivskyddad SQLite-export, med sidadressen angiven som data så att en rättad mening inte längre flyttade sidan och bröt länken.

Byggde en databasdriven CV-generator

Byggde en databasdriven generator för CV, referenser, portfolio och personliga brev i Python — 41 moduler, 10 580 rader — som renderar sex utdataformat från en SQLite-källa med 23 tabeller, sammanställd av en idempotent pipeline i 19 steg.

Fokusera den tekniska insatsen

Ta med ett representativt dataset, frågorna det ska besvara och dagens flaskhals. Vi kan avtala mått på kvalitet, genomströmning och återställning.

Se vad det kostar Hör av er →

Sätt ämnet i ert sammanhang

Berätta hur ”SQL” ingår i ert system. Vi kan hjälpa er att välja rätt tjänst och avgränsa arbetet.

Se våra tjänster

welcome@engineer.company +45 50 10 61 84 Köpenhamn, Danmark

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/categories/sql/