**Індивідуальне завдання**

…

**Опис виконання завдань згідно з календарним планом**

**03.06** Прибуття на базу практики. Ознайомлення з специфікою роботи. Проходження інструктажу з техніки безпеки.

**04.06**

**05.06**

**06.06**

**07.06**

**08.06** Вихідний день.

**09.06** Вихідний день.

**10.06**

**11.06**

**12.06**

**13.06**

**14.06**

**15.06** Вихідний день.

**16.06** Вихідний день.

**17.06**

**18.06**

**19.06**

**20.06**

**21.06**

**22.06** Вихідний день.

**23.06** Вихідний день.

**Опис технологій**

**Google Cloud.** Google Cloud Platform – запропонований компанією Google набір хмарних служб, які виконуються на тій же самій інфраструктурі, яку Google використовує для своїх продуктів призначених для кінцевих споживачів, таких як Google Search та YouTube. Окрім інструментів для керування, також надається ряд модульних хмарних служб, таких як обчислення, зберігання даних, аналіз даних та машинне навчання..

Google Cloud Platform надає такі оточення як інфраструктура як послуга, платформа як послуга, та безсерверні обчислення. Google Cloud Platform є частиною Google Cloud, який також включає G Suite, корпоративні версії Android та Chrome OS, а також API для машинного навчання та Google Maps.

**Google Cloud database, storage and data analytics services.** Повністю керовані, масштабовані служби баз даних для підтримки всіх ваших додатків сьогодні і завтра. Заряджайте свої програми. Прискорення впровадження з широкою сумісністю баз даних з відкритим кодом. Більше використовуйте свої дані за допомогою інтеграції з аналітикою та ML/AI. Зберігання даних для всіх ваших потреб. Ефективно захоплюйте, обробляйте та аналізуйте дані за допомогою продуктів Google Analytic для аналізу даних.

Найпоширеніші з них є:

* **BigQuery** – Швидке, економічне і повністю кероване сховище даних
* **Cloud Bigtable** – Високопродуктивний сервіс бази даних NoSQL для аналізу в режимі реального;
* **Cloud Datastore** – Масштабна база даних NoSQL для мобільних і веб-додатків;
* **Cloud Spanner –** Перший сервіс глобально узгоджених реляційних баз даних з горизонтальним;
* **Cloud SQL** – Повністю керований сервіс баз даних MySQL і PostgreSQL;
* **Cloud Storage** – Розпочніть роботу зі створення бакета, контейнера для ваших об'єктів у Cloud Storage;
* **Cloud Storage for Firebase** – Просте зберігання контенту;
* **Firebase Hosting** – Швидка доставка веб-контенту;
* **Firebase Realtime Database** – Зберігайте і синхронізуйте дані додатків в режимі реального часу.

**BigQuery.** BigQuery – це серверне, високо масштабоване та економічно ефективне сховище даних з хмарами, з вбудованим вбудованим BI-движком у пам'яті.

BigQuery, безсерверне сховище даних компанії, що не використовує сервер, розроблене для того, щоб зробити аналітиків даних більш продуктивними з неперевершеними ціновими показниками. Оскільки не існує інфраструктури для керування, ви можете зосередитися на розкритті значущої інформації, використовуючи знайомий SQL, без необхідності адміністратора бази даних.

Проаналізуйте всі ваші пакетні та потокові дані, створивши логічне сховище даних над керованим сховищем стовпців, а також дані з сховища об'єктів та електронних таблиць. Створюйте надзвичайно швидкі панелі та звіти з BI-движком у пам'яті. Побудувати та ввести в дію рішення для машинного навчання або провести геопросторовий аналіз, використовуючи простий SQL.

Потужне потокове передавання даних BigQuery дозволяє переглядати та аналізувати дані в режимі реального часу, забезпечуючи постійне поширення інформації. Крім того, ви можете аналізувати до 1 ТБ даних і зберігати 10 ГБ даних безкоштовно щомісяця.

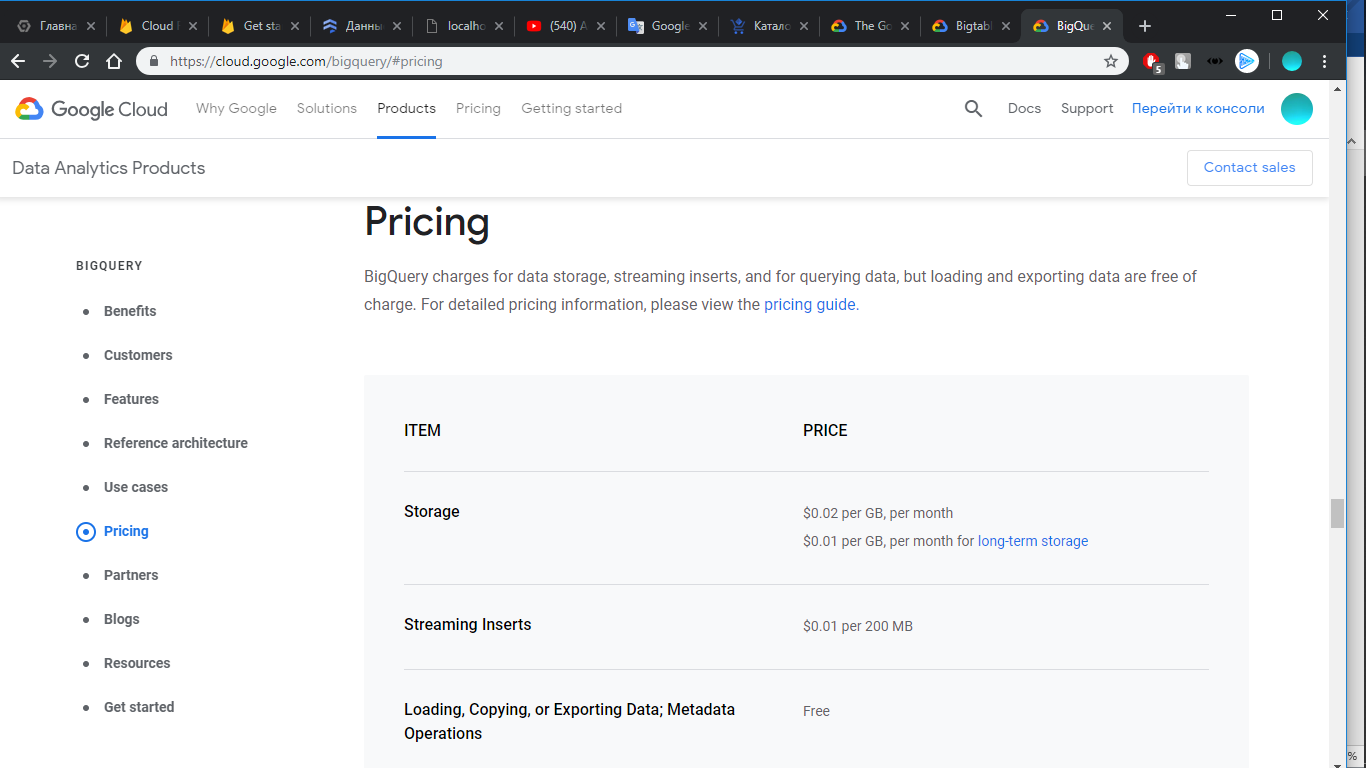


Рис. 1. Ціни на BigQuery

**Cloud Bigtable.** Gовністю керована NoSQL служба баз даних для великих (petabyte) аналітичних та операційних навантажень.

Cloud Bigtable як механізм зберігання даних для великомасштабних додатків з низькою затримкою, а також для обміну даними та аналітики.

Забезпечують і масштабують до сотні петабайт і плавно обробляють мільйони операцій в секунду. Зміни в конфігурації розгортання є негайними, тому під час реконфігурації немає простоїв.

Реплікація додає високу доступність для додатків, які обслуговують у прямому ефірі, та ізоляцію робочого навантаження для обслуговування та аналітики.

Cloud Bigtable легко інтегрується з популярними інструментами великих даних, такими як Hadoop, Cloud Dataflow і Cloud Dataproc. Крім того, Cloud Bigtable підтримує стандартний стандартний інтерфейс HBase API, що полегшує роботу ваших команд розробників.

Оскільки ми керуємо базою даних і керуємо налаштуванням і налаштуванням, можна зосередитися на розробці додатків.

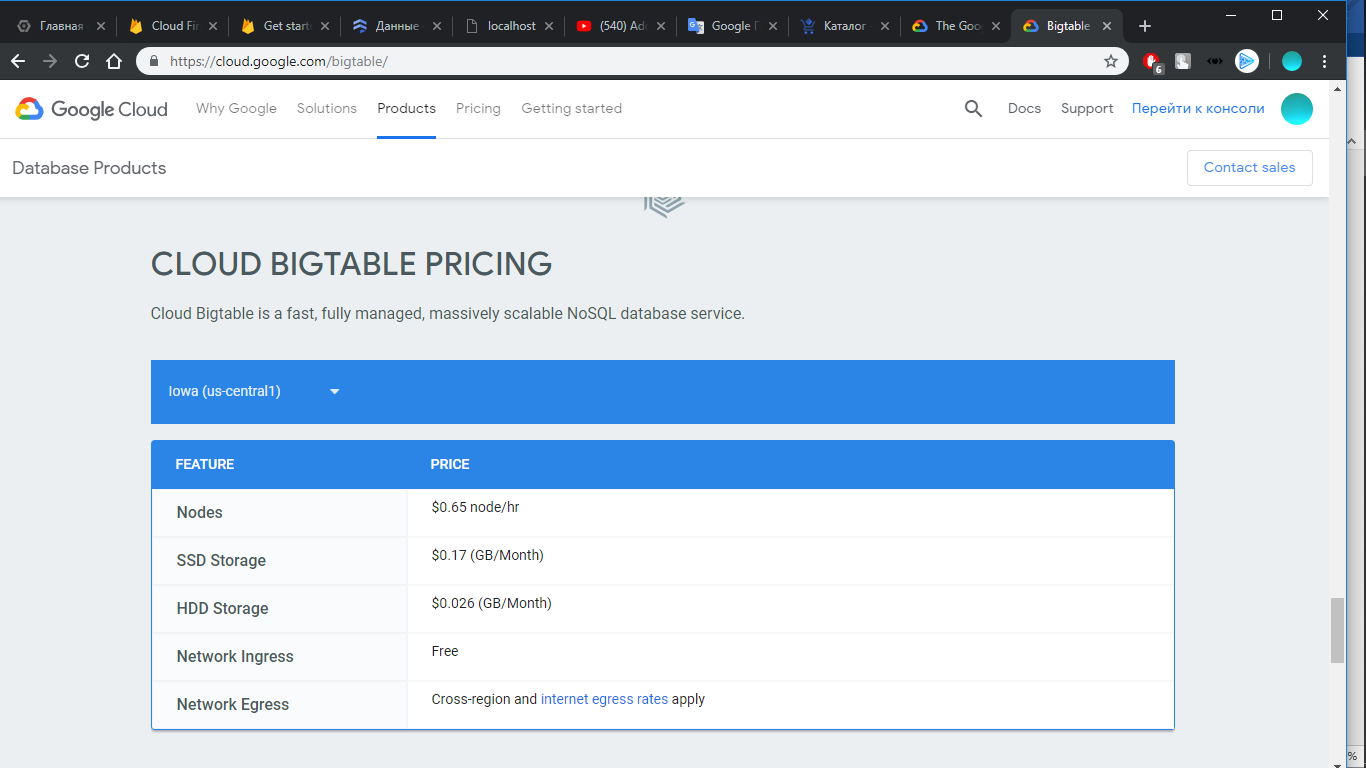


Рис. 2. Ціни на Cloud Bigtable

**Cloud Datastore.** Cloud Datastore - це масштабована база даних NoSQL для ваших веб- і мобільних додатків.

Cloud Datastore - це масштабована база даних NoSQL для ваших додатків. Cloud Datastore автоматично обробляє sharding і реплікацію, надаючи вам доступну і довговічну базу даних, яка автоматично масштабується для обробки навантажень ваших додатків. Cloud Datastore надає безліч можливостей, таких як транзакції ACID, SQL-подібні запити, індекси та багато іншого.

Завдяки інтерфейсу RESTful Cloud Datastore, доступ до даних може здійснюватися будь-якою ціллю розгортання. Ви можете створювати рішення, які охоплюють App Engine і Compute Engine, і покладатися на Cloud Datastore як на точку інтеграції.

Зосередьження на створенні своїх додатків, не турбуючись про надання та завантаження. Cloud Datastore масштабується плавно і автоматично з вашими даними, що дозволяє програмам підтримувати високу продуктивність, оскільки вони отримують більше трафіку.

Datastore - це база даних, яка дозволяє вам не турбуватися про внесення змін до базової структури даних, коли ваша програма розвивається. Datastore надає потужний механізм запитів, який дозволяє здійснювати пошук даних по декількох властивостях і сортувати за необхідності.

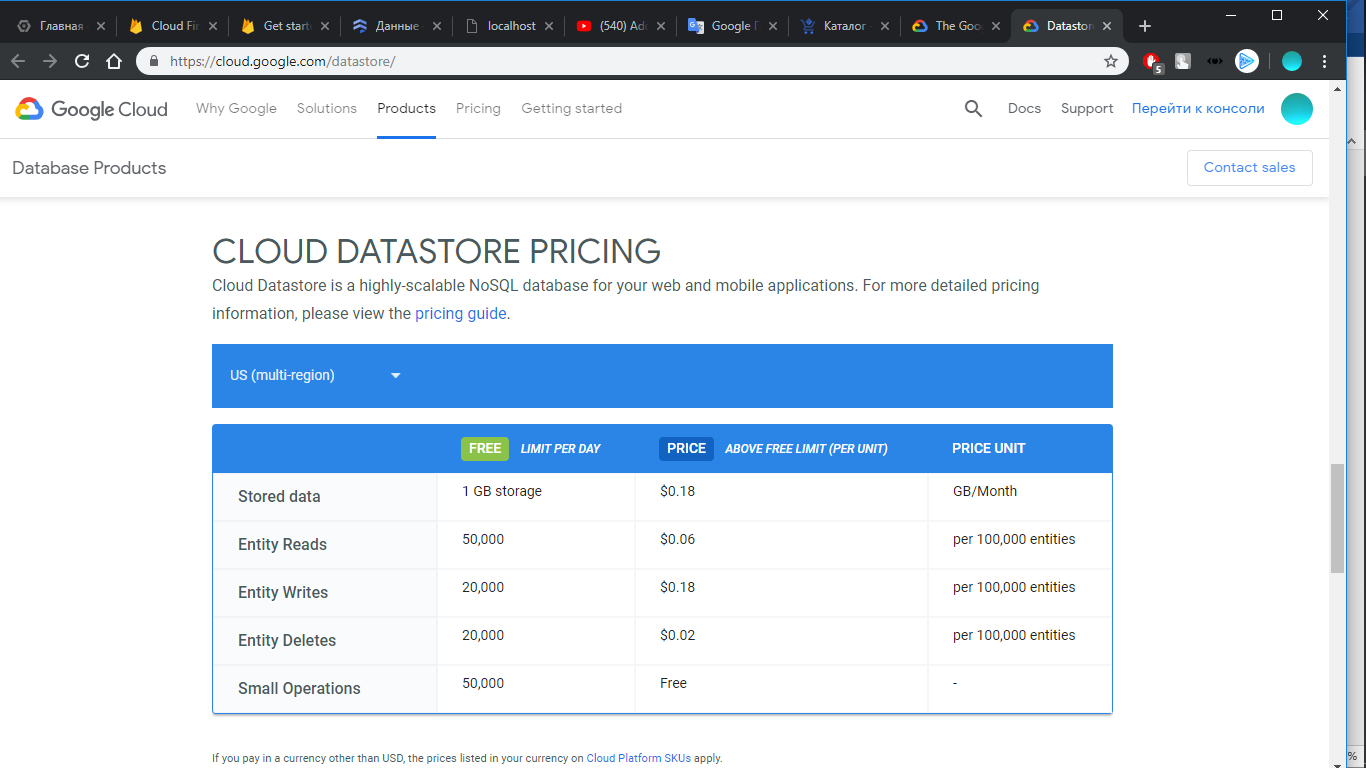


Рис. 3. Ціни на Cloud Datastore

**Cloud Spanner.** Перша горизонтально масштабована, сильно послідовна, реляційна служба бази даних.

Cloud Spanner є єдиним корпоративним, глобально розподіленим і сильно послідовним сервісом баз даних, побудованим для хмари, спеціально для об'єднання переваг реляційної структури бази даних з нереляційною горизонтальною шкалою. Ця комбінація забезпечує високопродуктивні транзакції та високу узгодженість між рядками, регіонами та континентами, що мають провідну угоду SLA про доступність у 99,999%, без планових простоїв та безпеки корпоративного рівня. Cloud Spanner революціонізує адміністрування та управління базами даних і робить розробку додатків більш ефективною.

Більшість баз даних сьогодні вимагають ухвалення компромісів між масштабами та послідовністю. За допомогою Cloud Spanner ви отримуєте найкращу структуру реляційних баз даних та масштаби нереляційних баз даних та продуктивність із зовнішнім сильним узгодженням між рядками, регіонами та континентами.

ІТ-адміністратори та адміністратори баз даних завалені невдячними завданнями. Завдяки Cloud Spanner зосередитися на доданні цінностей та інноваціях замість технічного обслуговування. Створення або масштабування глобально реплікованої бази даних для критично важливих програм тепер займає кілька кліків.

Високодоступні та високошвидкісні системи безпеки Google, як стандартні, а не дорогі додатки, допомагають забезпечити безпеку ваших програм у режимі онлайн. При архітектурі додатків Cloud Spanner має просту модель виставлення рахунків, яка не стягує додаткову плату за вибір конфігурації для високої доступності, реплікації або забезпечення більш безпечних, глобальних сильно узгоджених даних.

Cloud Spanner - це реляційна база даних з повною реляційною семантикою і обробляє зміни схеми як онлайн-операцію без планового простою. Повторне використання наявних навичок SQL для запиту даних у Cloud Spanner з використанням знайомого, стандартного стандарту ANSI 2011 SQL.

Забезпечення кращого досвіду для кінцевих користувачів із зовнішньою, сильною послідовністю. Програми повинні бути побудовані для глобальної бази користувачів і вам потрібна синхронна реплікація для надання правильних, точних і послідовних даних. Альтернативою є, можливо, непослідовне старенне читання, зменшений досвід роботи з кінцевим користувачем і складна логіка застосування, що призводить до складних, трудомістких помилок.

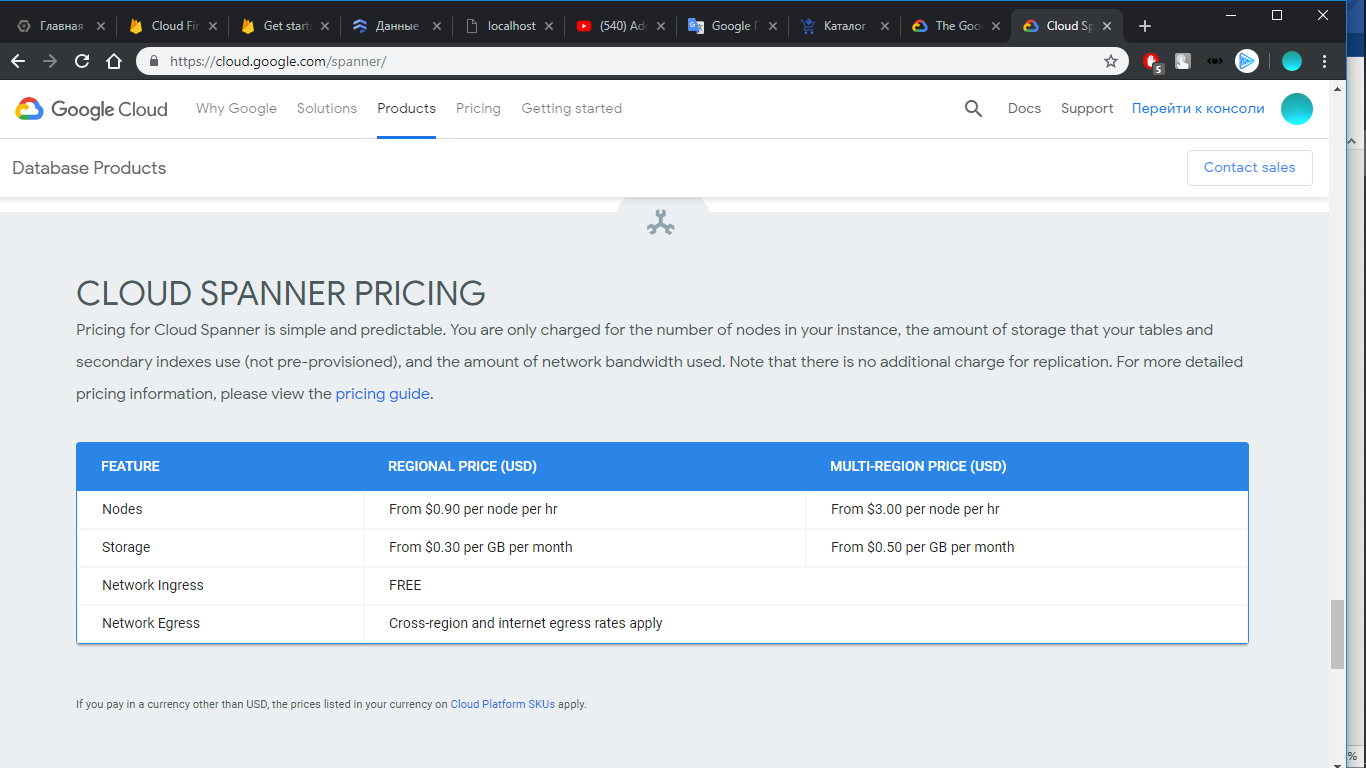


Рис. 4. Ціни на Cloud Spanner

**Cloud SQL.** Хмара SQL для SQL Server.

Cloud SQL - це повністю керована служба баз даних, яка дозволяє легко налаштовувати, підтримувати, керувати та керувати реляційними базами даних PostgreSQL, MySQL і SQL Server у хмарі. Cloud SQL пропонує високу продуктивність, масштабованість і зручність. Розташований на Google Cloud Platform, Cloud SQL надає інфраструктуру бази даних для програм, які працюють де завгодно.

Дозволяє Google керувати вашою базою даних, щоб можна було зосередитися на своїх програмах. Cloud SQL ідеально підходить для сайтів WordPress, додатків електронної комерції, інструментів CRM, геопросторових додатків і будь-яких інших програм, сумісних з MySQL, PostgreSQL або SQL Server.

Cloud SQL забезпечує високу продуктивність і масштабованість з ємністю до 10 ТБ, 40,000 IOPS та 416 Гб оперативної пам'яті на примірник.

Легко налаштувується реплікація та резервні копії для захисту даних. Перейдіть далі, включивши автоматичне перехід на інший, щоб зробити вашу базу даних доступною (HA). Ваші дані автоматично шифруються, а Cloud SQL - SSAE 16, ISO 27001, PCI DSS v3.0 і підтримує відповідність HIPAA.

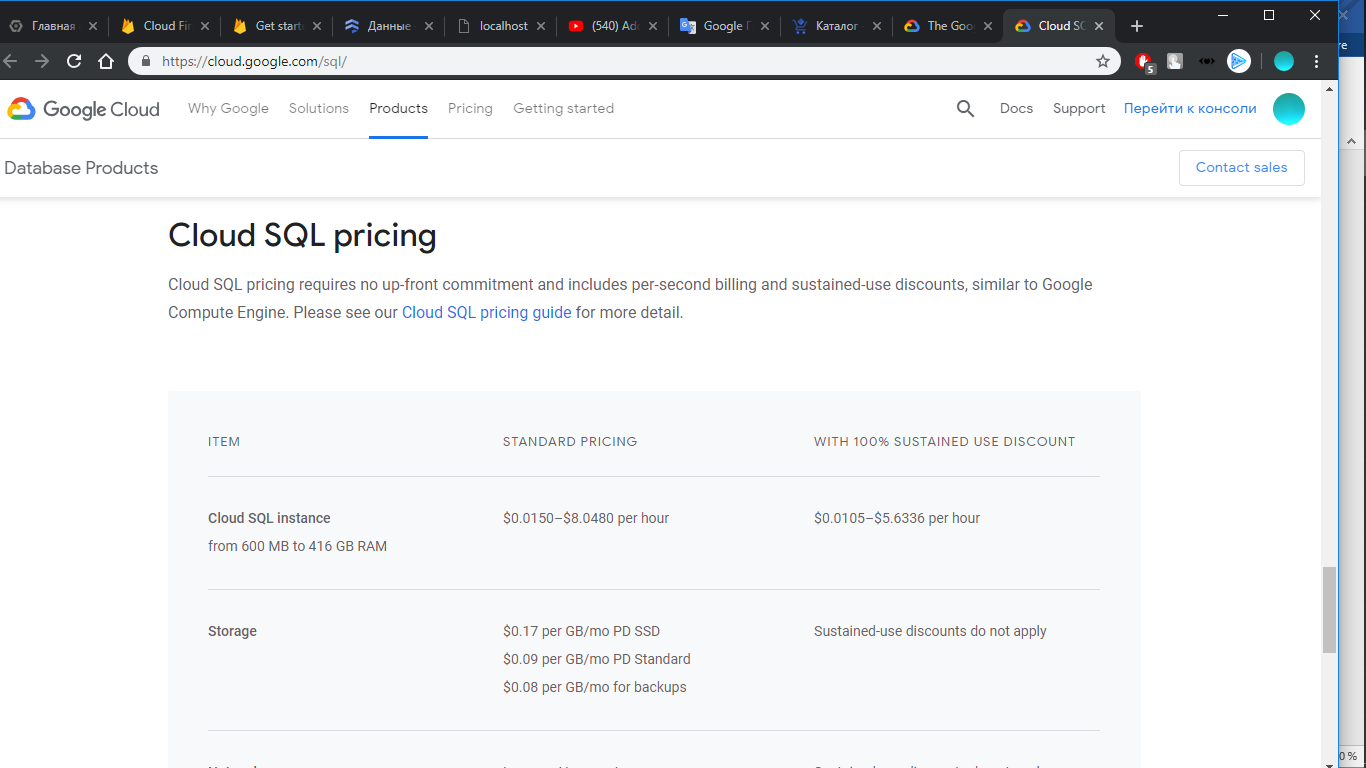


Рис. 5. Ціни на Cloud SQL

**Cloud Storage.** Єдиний об'єкт зберігання для розробників і підприємств.

Послідовний інтерфейс API Cloud Storage, затримка та швидкість у різних класах зберігання спрощують інтеграцію розробки та зменшують складність коду. Реалізуйте керування життєвим циклом об'єктів, щоб встановити час для живих (TTL) об'єктів, архівувати старішу версію об'єктів або знижувати класи зберігання, не порушуючи затримки або доступності. Налаштуйте власні політики для безперешкодного перенесення даних з одного класу зберігання на інший, залежно від потреб у вашій вартості та доступності.

Cloud Storage розрахований на 99,999999999% довговічності. Він зберігає дані надлишково, за допомогою автоматичних контрольних сум для забезпечення цілісності даних. За допомогою мультирегіонального сховища дані зберігаються в географічно різних місцях.

Cloud Storage пропонує необмежений об'єм для зберігання об'єктів, а окремі об'єкти можуть становити до 5TB. Об'єкти можуть бути перезаписані не частіше одного разу в секунду, а частота читання не обмежена. Об'єкти, розмір яких перевищує 5 Мб, слід завантажувати за допомогою багаточастинкового чи повторного завантаження. Щоб отримати докладніші відомості, перегляньте квоти & ліміти Cloud Storage.

Коли запис вдається успішно, остання копія об'єкта гарантовано буде повернута до будь-якого GET, глобально. Це відноситься до PUTs нових або перезаписаних об'єктів і DELETE.

Переміщуючи сховище даних з самостійного центру обробки даних або об'єкта колокації до GCP, викиди, безпосередньо пов'язані з сховищем даних вашої компанії, будуть нульовими.

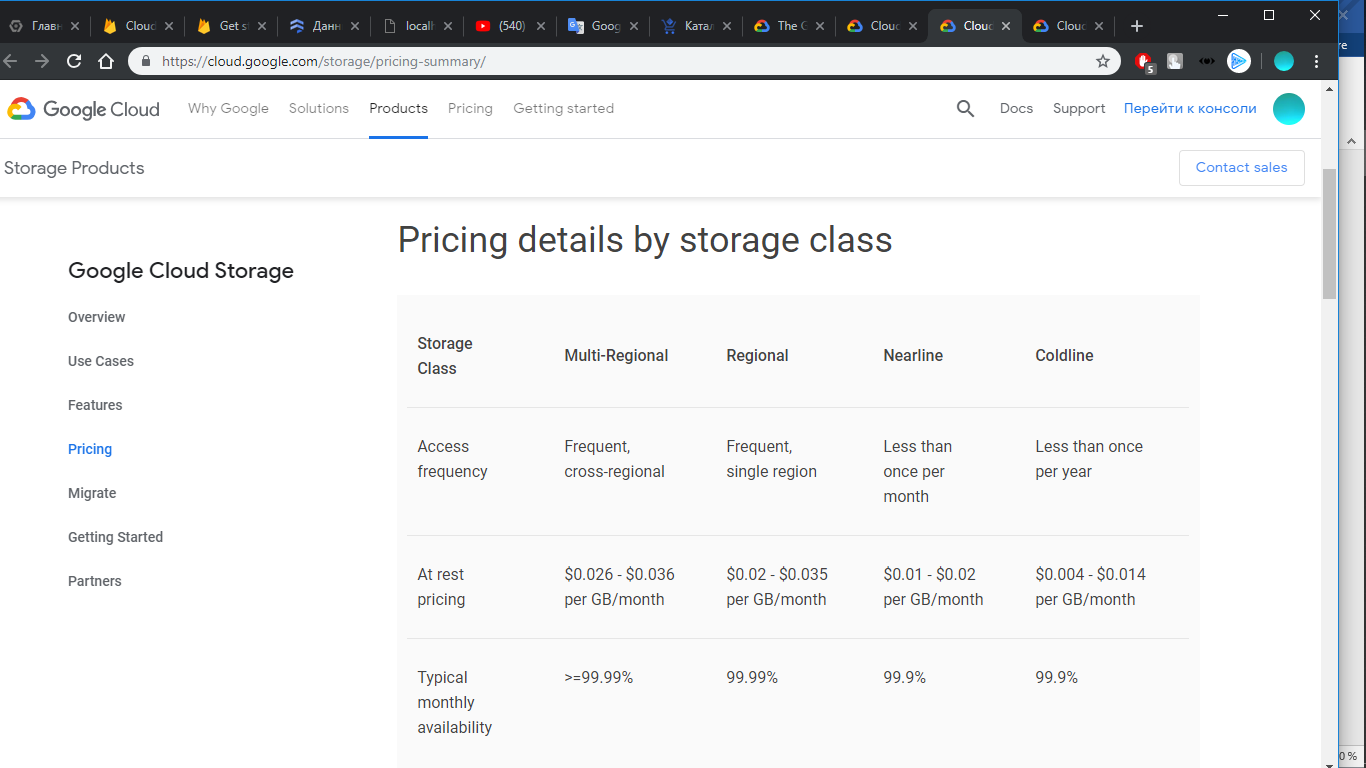


Рис. 6. Ціни на Cloud Storage

**Firebase Hosting.** Зберігати дані програми на основі хмари в глобальному масштабі.

Cloud Firestore - це швидка, повністю керована база даних документів NoSQL без серверів, що спрощує зберігання, синхронізацію та запит даних для мобільних, веб- і IoT-додатків у глобальному масштабі. Її клієнтські бібліотеки забезпечують синхронізацію в реальному часі та підтримку в автономному режимі, а функції безпеки та інтеграції з Firebase та Google Cloud Platform (GCP) прискорюють створення дійсно безсерверних програм.

Cloud Firestore - це база даних, створена у хмарі, яка забезпечує автоматичне масштабування, побудоване з нуля для використання потужної інфраструктури Google Cloud Platform. Він розроблений для того, щоб забезпечити великий досвід розробників і спростити розробку додатків за допомогою синхронізації в реальному часі, підтримки в автономному режимі та транзакцій ACID на сотнях документів і колекцій. Cloud Firestore інтегрована як з Google Cloud Platform (GCP), так і з Firebase, платформою мобільного розвитку Google.

Ви можете розмовляти безпосередньо з Cloud Firestore з мобільного або веб-клієнтів для по-справжньому серверного рішення. Не потрібно встановлювати сервер-посередник для керування доступом до даних. Це відмінно підходить для створення прототипів, ітерації та швидкого запуску виробничої системи.

За допомогою Cloud Firestore ваші програми можуть оновлюватися майже в реальному часі, коли змінюються дані на сервері. Це не тільки зручно для створення спільних багатокористувацьких мобільних додатків, але також означає, що ви можете зберігати свої дані в синхронізації з окремими користувачами, які можуть захотіти використовувати вашу програму з декількох пристроїв.

Cloud Firestore має повну підтримку в автономному режимі, тому ви можете отримувати доступ до своїх даних і вносити зміни, і ці зміни синхронізуються з хмарою, коли клієнт повернеться в онлайн. Вбудована підтримка в режимі офлайн використовує місцевий кеш для обслуговування та зберігання даних, тому програма залишається реагуючою незалежно від затримки мережі або підключення до Інтернету.

Надійні клієнтські бібліотеки Cloud Firestore полегшують оновлення та отримання нових даних, не турбуючись при цьому про встановлення мережевих з'єднань або непередбачувані умови перегонів. Вона може легко масштабуватися, коли ваше додаток зростає. Cloud Firestore дозволяє запускати складні запити щодо ваших даних. Це дає вам більшу гнучкість у способі структурування даних і часто може означати, що ви повинні робити менше фільтрації на клієнті, що зберігає ефективність мережевих дзвінків і використання даних.

Cloud Firestore - це швидка і повністю керована хмарна база даних NoSQL. Вона побудована для масштабування та використання переваг потужної інфраструктури GCP, з автоматичним горизонтальним масштабуванням та виведенням у відповідь на завантаження вашої програми. Вбудовані засоби контролю доступу до даних, які дозволяють здійснювати перевірку даних за допомогою мови налаштування.

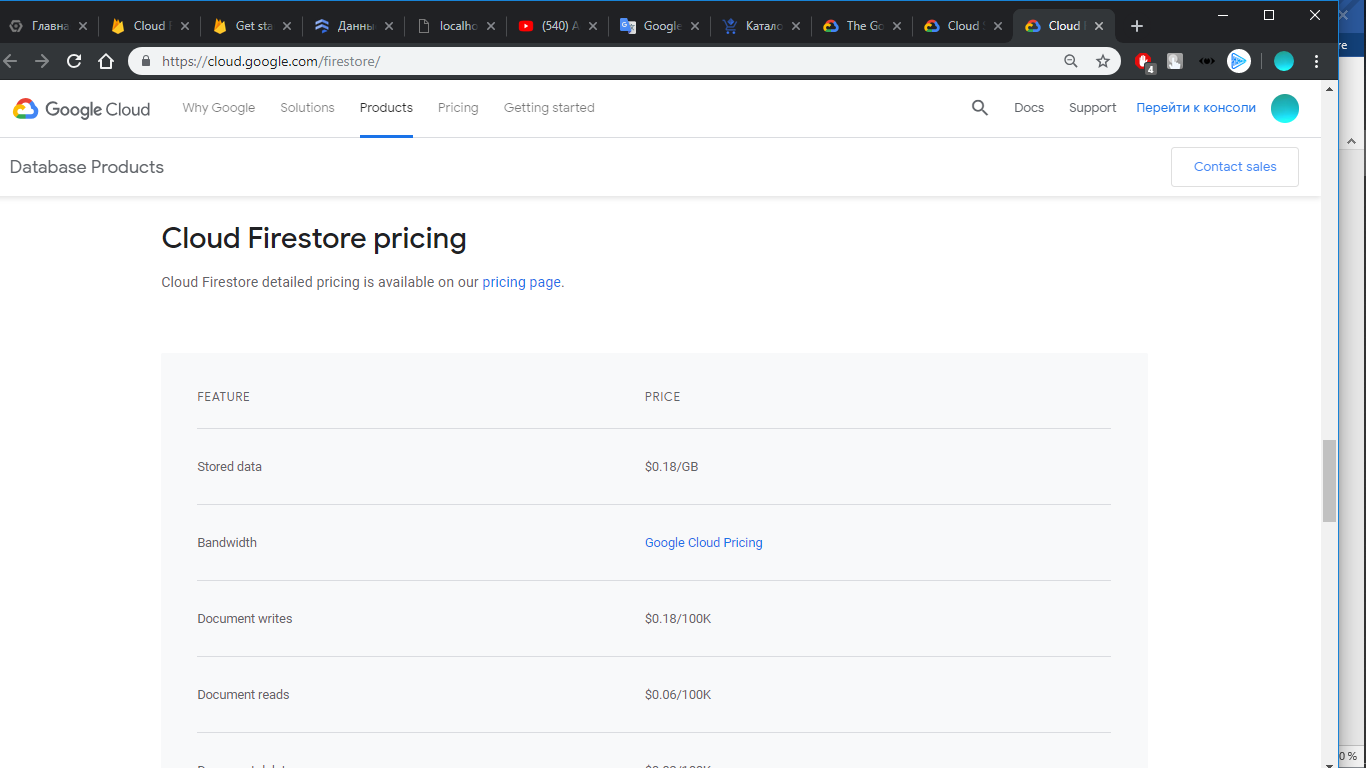


Рис. 7. Ціни на Cloud Firestore

**Реляційна база даних.** Реляційна база даних – це набір даних із зумовленими зв'язками між ними. Ці дані організовані у вигляді набору таблиць, що складаються із стовпців і рядків. У таблицях зберігається інформація про об'єкти, представлених в базі даних. У кожному стовпчику таблиці зберігається певний тип даних, в кожній комірці – значення атрибута. Кожна стоку таблиці являє собою набір пов'язаних значень, що відносяться до одного об'єкту або сутності. Кожен рядок в таблиці може бути позначена унікальним ідентифікатором, званим первинним ключем, а рядки з декількох таблиць можуть бути пов'язані з допомогою зовнішніх ключів. До цих даних можна отримати доступ багатьма способами, і при цьому реорганізовувати таблиці БД не потрібно.

**SQL.** Structured Query Language – основний інтерфейс роботи з реляційними БД. SQL став стандартом Національного інституту стандартів США (ANSI) в 1986 році. Стандарт ANSI SQL підтримується всіма популярними ядрами реляційних БД. Деякі з ядер також включають розширення стандарту ANSI SQL, що підтримують специфічний для цих ядер функціонал. SQL використовується для додавання, оновлення та видалення рядків даних, вилучення наборів даних для обробки транзакцій і аналітичних програм, а також для управління всіма аспектами роботи бази даних.

**Цілісність даних.** Цілісність даних – це повнота, точність і однаковість даних. Для підтримки цілісності даних в реляційних БД використовується ряд інструментів. В їх число входять первинні ключі, зовнішні ключі, обмеження «Not NULL», «Unique», «Default» і «Check». Ці обмеження цілісності дозволяють застосовувати практичні правила до даних в таблицях і гарантувати точність і надійність даних. Більшість ядер БД також підтримує інтеграцію призначеного для користувача коду, який виконується у відповідь на певні операції в БД.

Для дотримання цілісності даних всі транзакції в БД повинні відповідати вимогам ACID, тобто бути атомарними, однаковими, ізольованими і надійними. Атомарність – це умова, при якому або транзакція успішно виконується цілком, або, якщо будь-яка з її частин не виконується, вся транзакція скасовується. Одноманітність - це умова, при якому дані, що записуються в базу даних в рамках транзакції, повинні відповідати всім правилам і обмеженням, включаючи обмеження цілісності, каскади і тригери. Ізольованість необхідна для контролю над узгодженістю і гарантує базову незалежність кожної транзакції. Надійність має на увазі, що всі внесені в базу даних зміни на момент успішного завершення транзакції вважаються постійними.

**Транзакції.** Транзакція в базі даних – це один або кілька операторів SQL, виконаних у вигляді послідовності операцій, що представляють собою єдину логічну задачу. Транзакція являє собою неподільне дію, тобто вона повинна бути виконана як єдине ціле і або повинна бути записана в базу даних цілком, або не повинен бути записаний жоден з її компонентів. У термінології реляційних баз даних транзакція завершується або дією COMMIT, або ROLLBACK. Кожна транзакція розглядається як внутрішньо зв'язний, надійний і незалежний від інших транзакцій елемент.

**MySQL.** MySQL – вільна система керування реляційними базами даних. MySQL був розроблений компанією «ТсХ» для підвищення швидкодії обробки великих баз даних. Ця система керування базами даних (СКБД) з відкритим кодом була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася і зараз MySQL – одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

MySQL виникла як спроба застосувати mSQL до власних розробок компанії: таблиць, для яких використовувалися ISAM — підпрограми низького рівня для індексного доступу до даних. У результаті був вироблений новий SQL-інтерфейс, але API-інтерфейс залишився в спадок від mSQL. Звідки походить назва «MySQL» — достеменно не відомо. Розробники дають два варіанти: або тому, що практично всі напрацювання компанії починалися з префікса My, або на честь дівчинки на ім'я My, дочки Майкла Монті Віденіуса, одного з розробників системи.

Логотип MySQL у вигляді дельфіна носить ім'я «Sakila». Він був обраний з великого списку запропонованих користувачами «імен дельфіна». Ім'я «Sakila» було відправлено Open Source-розробником Ambrose Twebaze.

За час розвитку під орудою Oracle дедалі більше відокремлює MySQL від спільноти і робить процес розробки все менш прозорим. Наприклад, повернута практика поставки власницьких розширених функцій в Enterprise-версії MySQL, спостерігається приховування інформації про вразливості, зі складу виключений тестовий набір, закритий доступ до більшої частини системи відстеження помилок та припинено публікація згрупованого логу змін, що дозволяє судити про прив'язку патчів до конкретних змін.

MySQL вважається гарним рішенням для малих і середніх застосувань. Сирцеві коди сервера компілюються на багатьох платформах. Найповніше можливості сервера виявляються в UNIX-системах, де є підтримка багатопоточності, що підвищує продуктивність системи в цілому.

Можливості сервера MySQL:

* простота у встановленні та використанні;
* підтримується необмежена кількість користувачів, що одночасно працюють із БД;
* кількість рядків у таблицях може досягати 50 млн;
* висока швидкість виконання команд;
* наявність простої і ефективної системи безпеки.

**SQLite.** SQLite – полегшена реляційна система керування базами даних. Втілена у вигляді бібліотеки, де реалізовано багато зі стандарту SQL-92. Сирцевий код SQLite поширюється як суспільне надбання (англ. public domain), тобто може використовуватися без обмежень та безоплатно з будь-якою метою. Фінансову підтримку розробників SQLite здійснює спеціально створений консорціум, до якого входять такі компанії, як Adobe, Oracle, Mozilla, Nokia, Bentley і Bloomberg.

Особливістю SQLite є те, що вона не використовує парадигму клієнт-сервер, тобто рушій SQLite не є окремим процесом, з яким взаємодіє застосунок, а надає бібліотеку, з якою програма компілюється і рушій стає складовою частиною програми. Таким чином, як протокол обміну використовуються виклики функцій (API) бібліотеки SQLite. Такий підхід зменшує накладні витрати, час відгуку і спрощує програму. SQLite зберігає всю базу даних (включаючи визначення, таблиці, індекси і дані) в єдиному стандартному файлі на тому комп'ютері, на якому виконується застосунок. Простота реалізації досягається за рахунок того, що перед початком виконання транзакції весь файл, що зберігає базу даних, блокується; ACID-функції досягаються зокрема за рахунок створення файлу-журналу.

Кілька процесів або потоків можуть одночасно без жодних проблем читати дані з однієї бази. Запис в базу можна здійснити тільки в тому випадку, коли жодних інших запитів у цей час не обслуговується; інакше спроба запису закінчується невдачею, і в програму повертається код помилки. Іншим варіантом розвитку подій є автоматичне повторення спроб запису протягом заданого інтервалу часу.

У комплекті постачання йде також функціональна клієнтська частина у вигляді виконуваного файлу sqlite3, за допомогою якого демонструється реалізація функцій основної бібліотеки. Клієнтська частина працює з командного рядка, і дозволяє звертатися до файлу БД на основі типових функцій ОС.

Завдяки архітектурі рушія можливо використовувати SQlite як на вбудовуваних (embedded) системах, так і на виділених машинах з гігабайтними масивами даних.

Особливості SQLite:

* Транзакції атомарні, послідовні, ізольовані, і міцні (ACID) навіть після збоїв системи і збоїв живлення;
* Встановлення без конфігурації — не потребує ані установки, ані адміністрування;
* Реалізує значну частину стандарту SQL92;
* База даних зберігається в одному крос-платформовому файлі на диску;
* Підтримка терабайтних розмірів баз даних і гігабайтного розміру рядків і BLOBів;
* Малий розмір коду: менше ніж 350KB повністю налаштований, і менш 200KB з опущеними додатковими функціями;
* Швидший за популярні рушії клієнт-серверних баз даних для найпоширеніших операцій;
* Простий, легкий у використанні API;
* Написана в ANSI C, включена прив'язка до TCL; доступні також прив'язки для десятків інших мов;
* Добре прокоментований сирцевий код зі 100% тестовий покриттям гілок;
* Доступний як єдиний файл сирцевого коду на ANSI C, який можна легко вставити в інший проект;
* Автономність: немає зовнішніх залежностей;
* Крос-платформовість: з коробки підтримується Unix (Linux і Mac OS X), OS/2, Windows (Win32 і WinCE). Легко переноситься на інші системи;
* Сирці перебувають в суспільному надбанні;
* Поставляється з автономним клієнтом інтерфейсу командного рядка, який може бути використаний для управління базами даних SQLite.