

Penggunaan Redis Untuk Meningkatkan *Performance Retrieve Data* Dari *Database*

Frans Andrew Sinaga¹, Manarul Hidayat², Riyantoro Djati³

Program Studi Bisnis Digital, Universitas Indraprasta PGRI
Jalan Nangka No. 58 C (TB. Simatupang), Tanjung Barat, Jagakarsa, Jakarta Selatan

¹franspoenya@gmail.com

²manarulhidayat.id@gmail.com

³roof.toro@gmail.com

Abstract— *In today's world, application performance is crucial. If an application or system is slow to access, it will negatively impact the user experience and could lead to a decrease in visitors. As the need for systems or applications to have high data access speeds increases, caching technology such as Redis has become a relevant and efficient solution. This study aims to compare the performance of data retrieval from a MySQL database with data retrieval from Redis, through a simple program created in the Python programming language. The method used in this study is an experimental method, which consists of several stages: problem identification, hypothesis, design and implementation, data preparation, experimentation, analysis, and conclusions. At the end of this study, after conducting 3 trials, retrieving data from MySQL took an average of 2.083 seconds, while retrieving data from Redis was much faster, taking an average of 0.018 seconds.*

Keywords— *Redis, Database, MySQL, Python*

Abstrak— Di era sekarang, performa aplikasi sangat penting. Jika aplikasi atau sistem lambat saat diakses, pengalaman pengguna akan buruk dan bisa membuat pengunjung berkurang. Seiring meningkatnya kebutuhan sistem ataupun aplikasi terhadap kecepatan akses data yang tinggi, teknologi *caching* seperti Redis menjadi salah satu solusi yang relevan dan efisien. Penelitian ini bertujuan untuk membandingkan performa *retrieve data* dari *database* MySQL dengan *retrieve data* dari Redis, melalui program sederhana yang dibuat dari bahasa pemrograman Python. Metode yang digunakan pada penelitian ini yaitu metode eksperimental, yang terdiri dari beberapa tahapan yaitu identifikasi masalah, hipotesis, desain dan implementasi, persiapan data, percobaan, analisis, dan kesimpulan. Di akhir penelitian ini, setelah dilakukan 3 kali percobaan, *retrieve data* dari MySQL membutuhkan waktu rata-rata 2,083 detik, sedangkan *retrieve data* dari Redis jauh lebih cepat, yaitu membutuhkan waktu rata-rata 0,018 detik.

Kata kunci— *Redis, Database, MySQL, Python*

I. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi yang begitu pesat dalam beberapa dekade terakhir telah mengubah pola penggunaan aplikasi, baik pada sektor bisnis, pendidikan, kesehatan, maupun hiburan. Aplikasi

tidak lagi hanya berfungsi sebagai alat bantu sederhana, melainkan sudah menjadi sistem yang kompleks dengan kebutuhan pemrosesan data secara cepat, akurat, dan dapat diakses oleh banyak pengguna secara bersamaan. Dalam kondisi ini, kinerja (*performance*) dan skalabilitas (*scalability*) menjadi faktor penentu keberhasilan suatu aplikasi.

Salah satu tantangan utama yang dihadapi aplikasi *modern* adalah bagaimana mengelola permintaan *retrieve data* untuk ditampilkan ke dalam aplikasi secara cepat dan efisien. Penggunaan sistem basis data tradisional berbasis *disk* sering kali menjadi *bottleneck*, karena keterbatasan dalam kecepatan baca/tulis (*input/output* atau *I/O*). Hal ini mengakibatkan aplikasi lambat dalam memberikan respons, menurunkan kualitas *user experience*, bahkan dapat menyebabkan kegagalan layanan pada saat terjadi beban tinggi.

Dalam menjawab tantangan tersebut, solusi proses *retrieve data* dapat dilakukan dengan cepat dengan menggunakan Redis. Redis (*Remote Dictionary Server*) hadir sebagai salah satu teknologi yang dapat memenuhi kebutuhan ini. Redis merupakan *in-memory data store* berkecepatan tinggi yang mampu melakukan operasi baca dan tulis dalam hitungan milidetik, karena seluruh data disimpan di dalam *memory* (RAM/*Random Access Memory*). Redis menyimpan data dalam *memory* agar dapat diakses lebih cepat dan memastikan data tetap ada di *storage* [1].

Oleh karena itu, peneliti bermaksud melakukan penelitian penggunaan Redis ini, untuk membuktikan peningkatan performa *retrieve data* dengan menggunakan Redis. Redis akan dijalankan pada Sistem Operasi (OS) Linux Ubuntu, tetapi di dalam OS Windows, sehingga dibutuhkan Windows Subsystem for Linux (WSL). Hal ini dilakukan karena peneliti tidak memiliki OS Linux pada satu *environment* tersendiri.

WSL adalah fitur dari sistem operasi Windows yang memungkinkan pengguna untuk menjalankan lingkungan Linux langsung di dalam Windows tanpa memerlukan mesin virtual atau *dual-boot*. Solusi untuk menjalankan Linux di dalam sistem operasi Windows, seperti menggunakan WSL, mencerminkan upaya untuk mengatasi batasan-batasan antar platform. WSL dan

VirtualBox menjadi dua pilihan utama yang menarik perhatian, masing-masing dengan pendekatan unik dalam menyediakan lingkungan Linux di dalam lingkungan Windows [2].

Penelitian ini bertujuan untuk melakukan perbandingan performa durasi *retrieve* data dari Redis dan *retrieve* data dari *database*, yaitu menggunakan *database* MySQL. Perbandingan tersebut dilakukan dengan menggunakan program sederhana yang dibuat dari bahasa pemrograman Python. Python adalah bahasa pemrograman tingkat tinggi yang bersifat *interpreted*, interaktif, dan berorientasi objek. Python dirancang untuk memudahkan penulisan kode yang mudah dibaca dan dipelihara, dengan sintaksis yang jelas dan sederhana [3].

Database adalah beberapa data yang tersimpan serta terorganisir di dalam suatu sistem komputer, sehingga *user*/pengguna dapat mengakses, memanipulasi hingga menyimpan data dengan mudah, serta penggunaannya untuk mengelola informasi dengan efektif dan efisien [4]. MySQL merupakan *software* RDBMS (*Relational Database Management System*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah yang sangat besar, dapat diakses oleh banyak pengguna dan dapat melakukan suatu proses secara sinkron atau bersamaan [5]. MySQL akan diakses menggunakan *phpMyAdmin*, sehingga dibutuhkan Apache dan *Hypertext Preprocessor* (PHP).

II. METODE PENELITIAN

Penelitian ini menggunakan metode *Experimental Research*, dengan langkah-langkah sesuai yang terlihat pada Gambar 1.



Gambar. 1 Langkah-Langkah Penelitian

Berikut penjelasan dari langkah-langkah tersebut.

A. Perumusan Masalah

Perumusan masalah dilakukan dengan membandingkan 2 variabel performa *retrieve* data, yaitu performa *retrieve* data dari *database* dan performa *retrieve* data dari Redis. Adapun instrumen yang digunakan untuk mendapatkan waktu *retrieve* data tersebut, yaitu dengan menggunakan program sederhana yang dibuat menggunakan bahasa pemrograman Python. Pada penelitian ini, proses *retrieve*

data dari Redis akan dibuktikan lebih cepat dibandingkan *retrieve* data dari *database*. Redis adalah sebuah *in-memory data store* bersifat *open-source* yang digunakan sebagai *database*, *cache*, dan *message broker*. Redis menyimpan data langsung di *memory* (RAM), sehingga memungkinkan kecepatan akses data yang sangat tinggi, jauh lebih cepat dibandingkan dengan *database* konvensional yang menyimpan data di *disk* [6].

B. Formulasi Hipotesis

Hipotesis sebagai jawaban sementara dari permasalahan yang perlu untuk diuji. Hipotesis sebelum dilakukan percobaan pada penelitian ini, didapat dari beberapa sumber yang menyatakan Redis dapat meningkatkan *request* data dan membuat *down server time* berkurang. Hal tersebut sesuai kajian teoretis terhadap beberapa hasil penelitian terlebih dahulu mengenai penelitian ini.

Sebagai contoh karya ilmiah Gandhi mengenai Redis pada sistem nusANTARA yang sebelumnya tidak bisa *handle user* lebih dari 500 *users* pada saat percobaan berlangsung. Hal ini diatasi dengan mengimplementasikan strategi Redis, sehingga dapat membuat *down server time* dari sistem nusANTARA menjadi berkurang [7]. Di samping itu, pada implementasi Redis, menghasilkan rata-rata waktu *request* data kedua dan seterusnya hanya dibutuhkan waktu yang singkat dalam memuat data yang semakin besar selama belum ada perubahan data [8].

Berdasarkan hasil hipotesis tersebut, disimpulkan bahwa proses *retrieve* data dari Redis dapat dilakukan dan memiliki performa waktu yang lebih cepat.

C. Perancangan dan Implementasi

Sebelum dilakukan pencarian ataupun persiapan data, terlebih dahulu ditetapkan rancangan desain. Dilakukan desain arsitektur *input* data dari *database* MySQL ke Redis, serta proses *retrieve* data dari *database* MySQL dan *retrieve* data dari Redis.

Pada tahapan ini, dilakukan pengunduhan *software*/perangkat lunak yang dibutuhkan untuk menjalankan penelitian. Perangkat lunak merupakan satu *set item* atau objek yang membentuk "konfigurasi" yang mencakup program, dokumen, dan data [9].

Setelah itu dilakukan instalasi/*setup* per masing-masing *software*. Lalu dibuatkan program Python sederhana dan desain *database* menyimpan data untuk diuji. Program Python tersebut yang berfungsi untuk mengakses dan menampilkan data dari *database* dan Redis.

D. Persiapan Data

Persiapan analisis data dibutuhkan untuk menjalankan percobaan penelitian dan persiapan analisis data yang akan memberikan berbagai pertimbangan atas teknik dan prosedur statistik yang akan digunakan untuk meyakinkan bahwa kondisi yang diperlukan adalah valid dan reliabel. Peneliti mencari sampel data melalui *website* Kaggle dan menyimpannya ke dalam *database* MySQL.

E. Pelaksanaan Percobaan

Percobaan dapat dilakukan dengan mengikuti secara ketat apa yang sudah ditetapkan. Program Python yang telah dibuat, dijalankan untuk melakukan percobaan dengan dua konsep, yaitu percobaan *retrieve* data dari *database* dan percobaan *retrieve* data dari Redis.

F. Analisis Data

Didasarkan atas percobaan yang diimplementasikan tersebut, dilakukan pengukuran untuk memperoleh data perbandingan antara durasi *retrieve* data dari *database* dan *retrieve* data dari Redis.

G. Kesimpulan

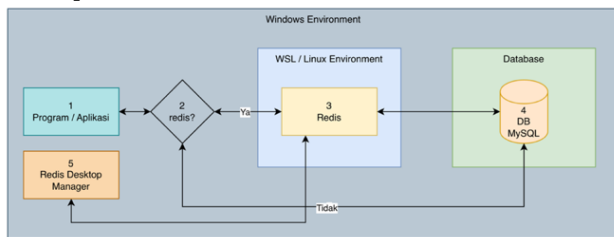
Berdasarkan hasil percobaan dan analisis yang dilakukan secara statistik, serta perbandingan data performa *retrieve* data dari percobaan yang telah dilakukan, sehingga dapat ditarik kesimpulan yang tepat.

III. HASIL DAN PEMBAHASAN

Pada bagian ini dijelaskan mengenai hasil dan pembahasan dari penelitian ini.

A. Desain

Adapun desain atau topologi dari penelitian ini, dapat dilihat pada Gambar 2.



Gambar 2. Desain Penelitian

Penelitian ini dijalankan dengan menggunakan satu laptop yang menggunakan sistem operasi Windows 11. Penjelasan mengenai desain arsitektur di atas sesuai nomornya, yaitu:

1) Program kecil/ sederhana yang hanya terdiri dari beberapa baris *coding* untuk memanggil dan menerima respons dari Redis ataupun *database*, dibuat menggunakan bahasa pemrograman Python. Program ini nanti akan dijalankan dengan menggunakan *tool* "Visual Studio Code".

2) Program memiliki fitur, yaitu mengambil data dari Redis atau langsung dari *database*.

3) Redis dijalankan pada *environment* Linux dengan menggunakan *feature* WSL yang memungkinkan Linux dapat dijalankan pada sistem operasi Windows.

4) *Database* menggunakan MySQL yang digunakan untuk menyimpan sampel data yang digunakan untuk membandingkan performa *retrieve* data melalui Redis ataupun dari *database*.

5) Redis Desktop Manager (RDM) digunakan untuk memonitor data yang ada pada Redis, sehingga dapat diketahui data apakah sudah tersimpan di Redis atau tidak.

B. Implementasi

Adapun cara instalasi dan *setup tools* ataupun program-program yang digunakan pada penelitian ini akan dijelaskan satu-persatu pada bagian ini.

1) Instalasi Python

Python digunakan untuk membuat program sederhana mengambil data dari Redis ataupun dari *database*. Instalasi dilakukan dengan men-download *installer* Python pada *website* Python "<https://www.python.org/downloads/windows/>".

2) Instalasi Visual Studio Code (VS Code)

VS Code digunakan sebagai editor untuk membuat dan menjalankan program Python. Instalasi dilakukan dengan men-download *installer* VS Code pada *website* Visual Studio berikut "<https://code.visualstudio.com/download>".

3) Instalasi dan Konfigurasi WSL (Windows Subsystem for Linux), Linux, Redis, Apache, MySQL

Pada tahapan ini akan dilakukan proses setup dan instalasi WSL dan Linux Ubuntu pada sistem operasi (OS) Windows 11, lalu dilakukan *setup* dan instalasi Redis, Apache dan *database* MySQL, pada OS Linux Ubuntu tersebut. Tahapan instalasi dan *setup* WSL dan Linux Ubuntu [10]. Tahapan instalasi Redis pada Linux Ubuntu [11]. Tahapan instalasi dan setup Apache, MySQL dan phpMyAdmin [12].

4) Instalasi aplikasi RDM

RDM digunakan untuk melihat dan *me-maintain* data yang ada pada Redis secara visual. Proses instalasi *tool* tersebut dilakukan dengan men-download *installer*-nya pada *website* RDM berikut : <https://docs.rdm.dev/en/2019/install/>.

C. Program

Program dibuat menggunakan bahasa pemrograman Python menggunakan dan menjalankannya dengan *tool* VS Code. Tahapan menjalankan Python pada VS Code [13]. Berikut daftar fungsi yang ada dalam program:

1) "connectRedis" digunakan untuk melakukan koneksi ke server Redis.

2) "connectMySQL" digunakan untuk melakukan koneksi ke *database* MySQL

3) "inputDataIkeaProdukMysqlToRedis" digunakan untuk meng-*input* data dari MySQL ke Redis, dengan "redis_key" digunakan sebagai *key* dari Redis.

4) "getDataIkeaProdukFromMysql(key)" digunakan untuk *retrieve* atau mengambil data 1 baris data dari *database* MySQL, lalu memberikan informasi waktu yang dibutuhkan untuk proses tersebut.

5) "getDataIkeaProdukFromMysql_All" digunakan untuk *retrieve* atau mengambil semua baris data dari *database* MySQL, lalu memberikan informasi waktu yang dibutuhkan untuk proses tersebut.

6) "getDataFromRedis(kName)" digunakan untuk *retrieve* atau mengambil data 1 baris data dari Redis, lalu memberikan informasi waktu yang dibutuhkan untuk proses tersebut.

7) "getDataFromRedis_All(kName)" digunakan untuk *retrieve* atau mengambil semua baris data dari Redis, lalu memberikan informasi waktu yang dibutuhkan untuk proses tersebut.

8) "deleteKeysValueRedis(kName)" digunakan untuk menghapus semua baris data dari Redis.

9) "pMenu" merupakan fungsi utama, untuk menampilkan tampilan program dan menerima *input* kode yang digunakan untuk menjalankan fungsi yang ingin dijalankan oleh pengguna.

Pada saat menjalankan program, fungsi utama "pMenu" akan dipanggil, lalu meminta *input* kode untuk menjalankan fungsi yang ingin dijalankan oleh pengguna. Adapun kode tersebut, yaitu:

- 81, digunakan untuk memanggil dan menjalankan fungsi "inputDataIkeaProdukMysqlToRedis"
- 82, digunakan untuk memanggil dan menjalankan fungsi "getDataIkeaProdukFromMysql(key)"
- 83, digunakan untuk memanggil dan menjalankan fungsi "getDataFromRedis(kName)"
- 52, digunakan untuk memanggil dan menjalankan fungsi "getDataIkeaProdukFromMysql_All"
- 53, digunakan untuk memanggil dan menjalankan fungsi "getDataFromRedis_All(kName)"
- 91, digunakan untuk memanggil dan menjalankan fungsi "deleteKeysValueRedis(kName)"
- 00, digunakan untuk keluar dari program

D. Persiapan Data

Penelitian ini menggunakan Data Produk IKEA, yang di-download dari situs web Kaggle melalui URL (*Uniform Resource Locator*) berikut : "https://www.kaggle.com/datasets/thedevastator/ikea-product". Data tersebut dimasukkan ke dalam *database* MySQL melalui proses *import* data dari phpMyAdmin [14]. Populasi data terdiri dari 3694 baris data dengan 14 kolom data. Peneliti melakukan 3 kali percobaan dengan *retrieve* 1 baris data dari populasi tersebut secara *random*. Proses *retrieve* data dilakukan baik dari *database* MySQL maupun Redis. Pada akhir penelitian, dilakukan percobaan untuk *retrieve* semua baris data dari populasi tersebut pada kedua *database*.

E. Percobaan

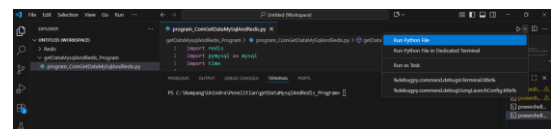
Program yang dibuat menggunakan bahasa pemrograman Python tersebut dijalankan dengan aplikasi Visual Studio Code (VS Code). Dengan menjalankan program tersebut, uji coba penelitian dilakukan dengan

proses *input* data dari *database* MySQL ke dalam Redis terlebih dahulu, lalu dilanjutkan dengan proses *retrieve* 1 (satu) baris data dari *database* MySQL dan Redis, dengan menggunakan *id* atau *key* data yang sama. Proses tersebut dilakukan sebanyak 3 (tiga) kali agar sampel data uji coba yang diambil akurat. Setelah itu, di akhir dilakukan *retrieve* semua baris data dari baik dari *database* MySQL dan Redis. Saat proses *retrieve* data, program akan menampilkan waktu yang dibutuhkan untuk proses *retrieve* data tersebut. Waktu tersebut akan dicatat dijadikan sebagai variabel pembandingan dalam penelitian ini.

Di akhir, dilakukan perhitungan secara manual, untuk membandingkan durasi *retrieve* data dari *database* MySQL dengan durasi *retrieve* data dari Redis dari setiap percobaan. Berikut prosesnya:

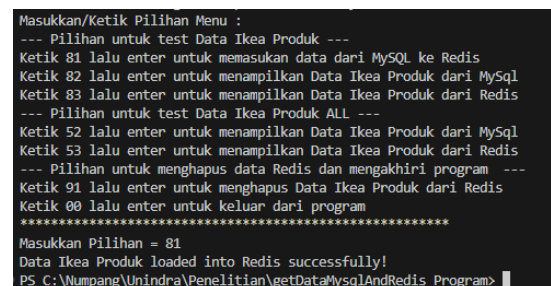
1) Input data dari database MySQL ke dalam Redis

- *Running* program dengan mengeklik "Run Python File" pada ikon kepala panah pada kanan atas VS Code, seperti yang terlihat pada Gambar 3.



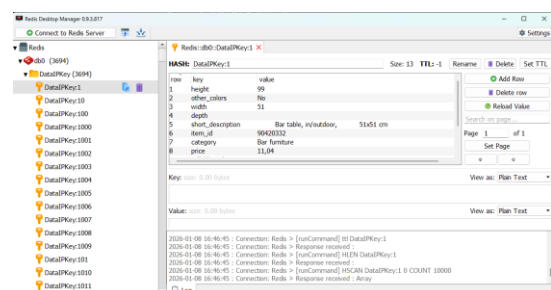
Gambar. 3 Proses Input Data – Tahap 1

- Setelah tampilan program muncul, masukkan kode "81" lalu tekan tombol Enter



Gambar. 4 Proses Input Data – Tahap 2

- Jalankan program RDM, untuk melihat data sudah ada di server Redis



Gambar. 5 Proses Input Data – Tahap 3

2) Percobaan pertama perbandingan durasi *retrieve* satu baris data dengan *key* data "3123"

- *Retrieve* data dari *database* MySQL

Jalankan program, lalu masukkan kode "82", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 82
```

Gambar. 6 Proses Percobaan Pertama – Tahap 1

Ketik "3123", lalu tekan tombol Enter

```
Masukkan Key Data (1 s.d 3964) = 3123
[{"Item_id": "00356341", "name": "KLIPPEN", "category": "Tables & desks", "price": "19,79", "old_price": "No old price", "available_online": "TRUE", "link": "https://www.ikea.com/sa/en/p/klippen-table-top-grey-light-grey-00356341/", "other_colors": "Yes", "short_description": "Table top, 120x80 cm", "designer": "Ikea Christensen", "depth": "", "height": "", "width": "60"}]
Time: 2.1239066620608
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program
```

Gambar. 7 Proses Percobaan Pertama – Tahap 2

Waktu yang dibutuhkan adalah **2,12390 detik**.

- *Retrieve* data dari Redis
Jalankan program, lalu masukkan kode "83", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 83
```

Gambar. 8 Proses Percobaan Pertama - Tahap 3

Ketik "3123", lalu tekan tombol Enter

```
Masukkan Key Data (1 s.d 3964) = 3123
[{"Item_id": "00356341", "name": "KLIPPEN", "category": "Tables & desks", "price": "19,79", "old_price": "No old price", "available_online": "TRUE", "link": "https://www.ikea.com/sa/en/p/klippen-table-top-grey-light-grey-00356341/", "other_colors": "Yes", "short_description": "Table top, 120x80 cm", "designer": "Ikea Christensen", "depth": "", "height": "", "width": "60"}]
Time: 0.0249725983206079
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program
```

Gambar. 9 Proses Percobaan Pertama – Tahap 4

Waktu yang dibutuhkan adalah **0,02497 detik**.

3) Percobaan kedua perbandingan durasi *retrieve* satu baris data dengan key data "21"

- *Retrieve* data dari *database* MySQL
Jalankan program, lalu masukkan kode "82", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 82
```

Gambar. 10 Proses Percobaan Kedua – Tahap 1

Ketik "21", lalu tekan tombol Enter

```
Masukkan Key Data (1 s.d 3964) = 21
[{"Item_id": "00340759", "name": "EKEDALEN", "category": "Bar furniture", "price": "41,46", "old_price": "No old price", "available_online": "TRUE", "link": "https://www.ikea.com/sa/en/p/ekedalen-bar-table-white-00340759/", "other_colors": "No", "short_description": "Bar table, 120x80 cm", "designer": "Erik Johansson", "depth": "", "height": "", "width": "80"}]
Time: 2.001346259878013
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program
```

Gambar. 11 Proses Percobaan Kedua – Tahap 2

Waktu yang dibutuhkan adalah **2,06114 detik**.

- *Retrieve* data dari Redis
Jalankan program, lalu masukkan kode "83", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 83
```

Gambar. 12 Proses Percobaan Kedua – Tahap 3

Ketik "21", lalu tekan tombol Enter

```
Masukkan Pilihan = 83
Masukkan Key Data (1 s.d 3964) = 21
[{"Item_id": "00340759", "name": "EKEDALEN", "category": "Bar furniture", "price": "41,46", "old_price": "No old price", "available_online": "TRUE", "link": "https://www.ikea.com/sa/en/p/ekedalen-bar-table-white-00340759/", "other_colors": "No", "short_description": "Bar table, 120x80 cm", "designer": "Erik Johansson", "depth": "", "height": "80", "width": "80"}]
Time: 0.010081768035888672
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program
```

Gambar. 13 Proses Percobaan Kedua – Tahap 4

Waktu yang dibutuhkan adalah **0,01008 detik**.

4) Percobaan ketiga perbandingan durasi *retrieve* satu baris data dengan key data "1500"

- *Retrieve* data dari *database* MySQL
Jalankan program, lalu masukkan kode "82", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 82
```

Gambar. 14 Proses Percobaan Ketiga – Tahap 1

Ketik "1500", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 82
```

Gambar. 15 Proses Percobaan Ketiga – Tahap 2

Waktu yang dibutuhkan adalah **2,06611 detik**.

- *Retrieve* data dari Redis
Jalankan program, lalu masukkan kode "83", lalu tekan tombol Enter

```
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\pythoncore-3.14-64\python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_ConnectDataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk memasukan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengakhiri program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 83
```

Gambar. 16 Proses Percobaan Ketiga – Tahap 3

Lalu ketik "1500", lalu tekan tombol Enter

```

Masukkan Key Data (1 s.d 3964) = 1500
[{"sellable_online": "TRUE", "b'item_id': 'b'69229217', 'b'width': 'b'', 'b'depth': 'b'', 'b'short_description': 'b'
salle and 6 chairs, 180x240 cm', 'b'link': 'b'https://www.ikea.com/sa/en/p/ekedalen-bernard-table-and-6-chairs-b
room-yuk-white-s69229217', 'b'price': 'b'227,71', 'b'name': 'b'EKEDALEN / BERNARD', 'b'height': 'b'', 'b'other_colors': 'b'N
o', 'b'category': 'b'Chairs', 'b'designer': 'b'EH17h Johansson/Ola Wihlborg', 'b'old_price': 'b'No old price'}}
[time: 0.01924049329581953]
PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program >

```

Gambar. 17 Proses Percobaan Ketiga – Tahap 4

Waktu yang dibutuhkan adalah **0,01924 detik**.

5) Percobaan keempat perbandingan durasi *retrieve* semua baris data

- *Retrieve* data dari *database* MySQL

Jalankan program, lalu masukkan kode "52", lalu tekan tombol Enter

```

PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\Python314-64\
python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_comGetdataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk masukkan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengaktifkan program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 52

```

Gambar. 18 Percobaan Keempat – Tahap 1

Berikut tampilan setelah tekan tombol Enter

```

program_ComGetdataMySQLAndRedis.py K
getdataMySQLAndRedis_Program > program_ComGetdataMySQLAndRedis.py > getdataFromRedis_All
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Gambar. 19 Percobaan Keempat – Tahap 2

Waktu yang dibutuhkan adalah **2,58651 detik**.

- *Retrieve* data dari Redis

Jalankan program, lalu masukkan kode "53", lalu tekan tombol Enter

```

PS C:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program & C:\Users\USER\AppData\Local\Python\Python314-64\
python.exe c:\Vumpang\Unindra\Pemelitian\getdataMySQLAndRedis_Program\program_comGetdataMySQLAndRedis.py
*** Selamat Data Program Compare Get Data MySQL dan Redis ***
Masukkan/Ketik Pilihan Menu :
--- Pilihan untuk test Data Ikea Produk ---
Ketik 81 lalu enter untuk masukkan data dari MySQL ke Redis
Ketik 82 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 83 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk test Data Ikea Produk ALL ---
Ketik 52 lalu enter untuk menampilkan Data Ikea Produk dari MySQL
Ketik 53 lalu enter untuk menampilkan Data Ikea Produk dari Redis
--- Pilihan untuk menghapus data Redis dan mengaktifkan program ---
Ketik 91 lalu enter untuk menghapus Data Ikea Produk dari Redis
Ketik 80 lalu enter untuk keluar dari program
*****
Masukkan Pilihan = 53

```

Gambar. 20 Percobaan Keempat – Tahap 3

Berikut tampilan setelah tekan tombol Enter

```

program_ComGetdataMySQLAndRedis.py K
getdataMySQLAndRedis_Program > program_ComGetdataMySQLAndRedis.py > getdataFromRedis_All
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Gambar. 21 Percobaan Keempat – Tahap 4

Waktu yang dibutuhkan adalah **1,99364 detik**.

F. Analisis

Dari hasil percobaan di atas maka hasil perbandingan *retrieve* data dari *database* MySQL dan Redis dapat dirangkum pada Tabel 1.

TABEL I
HASIL ANALISIS DATA

Percobaan	Total Retrieve Data	MySQL (detik)	Redis (detik)
1	1 baris data	2,12390	0,02497
2	1 baris data	2,06114	0,01008
3	1 baris data	2,06611	0,01924
4	3694 baris data	2,58651	1,99364

Rata-rata durasi dari hasil percobaan *retrieve* 1 baris data dengan yang terdiri dari 14 kolom data, yaitu :

- *Database* MySQL :
(2,12390 + 2,06114 + 2,06611)/3 = **2,083 detik**.
- Redis :
(0,02497 + 0,01008 + 0,01924)/3 = **0,018 detik**.

Dari hasil tersebut, penelitian ini hanya dijalankan pada sebuah laptop dengan kapasitas *memory* 4 *gigabyte*, *processor* Core i5, dan OS Windows 11. Dengan demikian peneliti menyampaikan, bahwa penelitian ini belum mempertimbangkan pengaruh jumlah *memory* dan *processor* yang digunakan, serta populasi data yang digunakan masih tergolong kecil. Percobaan juga dilakukan dengan satu permintaan *retrieve* data dengan satu pengguna atau *user*, sehingga belum mencakup pengaruh bagaimana jika dilakukan dengan banyak permintaan *retrieve* data dan dengan banyak pengguna pada saat yang bersamaan.

IV. KESIMPULAN

Penelitian penggunaan Redis untuk meningkatkan performa *retrieve* data dari *database* ini, dilakukan dengan percobaan *retrieve* data menggunakan program Python. Dari hasil percobaan tersebut, dapat dibandingkan durasi *retrieve* data dari *database* MySQL dan Redis. Dari hasil perbandingan tersebut, disimpulkan *retrieve* data dari

Redis memiliki performa durasi lebih cepat dari dibandingkan *retrieve* data dari *database* MySQL.

Pada percobaan pertama, kedua dan ketiga untuk *retrieve* 1 baris data dengan 14 kolom data dari *database* memiliki rata-rata **2,083 detik**, sedangkan *retrieve* 1 baris data dengan 14 kolom dari Redis memiliki rata-rata **0,018 detik**.

Oleh karena itu, peneliti menyarankan bagi para pengembang *web* untuk memanfaatkan Redis sebagai *layer cache* dalam proses pengambilan data, terutama untuk data yang sering diakses dan jarang berubah. Pendekatan ini memberikan beberapa keuntungan utama dibandingkan mengambil data langsung dari *database* setiap saat.

Dengan demikian menggunakan Redis dapat disarankan untuk meningkatkan performa aplikasi. Hal ini karena Redis menyimpan data di *memory* (*in-memory*), sehingga waktu akses jauh lebih cepat dibandingkan *database* relasional berbasis *disk*. Hal ini sangat efektif untuk mengurangi *latency* pada *endpoint* yang memiliki *traffic* tinggi.

Namun penelitian ini dilakukan dengan populasi data yang berskala kecil, *environment* yang digunakan memiliki spesifikasi yang kecil, serta percobaan dilakukan hanya untuk satu permintaan *retrieve* data dengan satu pengguna. Peneliti menyampaikan, hal tersebut menjadi pertimbangan untuk para pengembang ataupun peneliti lainnya.

REFERENSI

- [1] S. Suehring, *Redis For Dummies*, 2nd Limited Edition, Hoboken, New Jersey: John Wiley & Sons, Inc., 2021, p. 7.
- [2] R. A. Wijayanti, R. A. Firdaus, N. B. N. Putra dan A. R. Kardan, "Analisis perbandingan penggunaan Kali Linux pada Windows Subsystem for Linux (WSL) dan VirtualBox terhadap OpenSSL benchmark testing," *Journal on Education*, vol. 6, no. 1, pp. 1-2, Jul. 2023, doi: 10.31004/joe.v6i1.4664.
- [3] M. Lutz, *Learning Python*, Fifth Edition, R. Roumeliotis, Penyunt., Sebastopol, CA, California: O'Reilly Media, Inc., 2013, pp. 18-19.
- [4] G. Maulani, Nirsal dan R. Sari, *Sistem Basis Data*, W. Yuliani, Penyunt., Padang Pariaman, Sumatera Barat: *Lingkar Edukasi Indonesia*, 2024, pp. 1-2.
- [5] M. L. A. Nurachman, U. M. H. Tamyiz dan Minarto, "Rancang bangun aplikasi penyewaan skuter online berbasis mobile menggunakan metode extreme programming," *JATIKOM: Jurnal Aplikasi dan Teori Ilmu Komputer*, vol. 7, no. 2, p. 70, Sep. 2024, doi: 10.17509/jatikom.v7i2.53464.
- [6] J. L. Carlson, *Redis in Action*, Shelter Island, New York: Manning Publications, 2013, pp. 67-68.
- [7] R. J. Iskandar, Kartono dan Gandhi, "Penerapan Redis dalam sistem e-office Desa Sutura," *Jurnal InTekSis*, vol. 9, no. 1, p. 50, May 2022, doi: 10.66003/inteksis.v9i1.1017.
- [8] D. R. Prasetyo, "Implementasi Redis untuk mempercepat pengambilan data (studi kasus: sistem Dasawisma PKK Sumsel)," *Explore : Jurnal Sistem Informasi dan Telematika*, vol. 15, no. 2, p. 214, Dec. 2024, doi: 10.36448/jsit.v15i2.3948.
- [9] L. P. Sumirat, *Dasar Dasar Rekayasa Perangkat Lunak*, Malang, Jawa Timur: Madza Media, 2023, pp. 1-2.
- [10] Microsoft, "Cara menginstal Linux di Windows dengan WSL," Microsoft, 8 Agustus 2025. [Online]. Available: <https://learn.microsoft.com/id-id/windows/wsl/install>. [Diakses Agustus 2025].
- [11] DigitalOcean, "How to install and secure Redis on Ubuntu," Digital Ocean, 9 October 2025. [Online]. Available: <https://www.digitalocean.com/community/tutorials/how-to-install-and-secure-redis-on-ubuntu>. [Diakses October 2025].
- [12] Manu, "Cara install Apache, MySQL, PHP (LAMP) di Ubuntu," Plasa Web Host, 29 April 2025. [Online]. Available: <https://plasawebhost.com/panduan/cara-install-apache-mysql-php-lamp-di-ubuntu-22-04.html>. [Diakses Oktober 2025].
- [13] K. Chen, "Cara menjalankan Python pada Visual Studio Code," Medium, 18 Juni 2020. [Online]. Available: <https://medium.com/@kelvin.chen996/cara-menjalankan-python-pada-visual-studio-code-75fd1ff6dace>. [Diakses Oktober 2025].
- [14] A. Prawoto, "Impor file CSV dengan phpMyAdmin," Jago Web Hosting, 11 August 2016. [Online]. Available: <https://jagowebdev.com/impor-file-csv-dengan-phpmyadmin/>. [Diakses Oktober 2025].