

**Міністерство освіти і науки України  
Національний технічний університет України «Київський політехнічний  
інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки**

**Кафедра інформатики та програмної інженерії**

**Звіт  
з лабораторної роботи № 1 з дисципліни  
«Проектування інтелектуальних інформаційних систем»**

**Викладач:  
Баришич Л.М.**

**Виконав:  
студент 3 курсу  
групи ІП-12 ФІОТ  
Дулов Денис**

**Київ 2023**

## Скрипт створення баз даних та таблиць:

```
#!/usr/bin/env bash
set -euo pipefail # Enable strict mode
MARIADB=$(which mariadb)
CPIMPORT=$(which cpimport)
SCHEMA_DIR=$(readlink -f ./schema)
NAME1='airports'
NAME2='airlines'
NAME3='flights'
GREEN='\033[0;32m'
RED='\033[0;31m'
NC='\033[0m'
if $MARIADB <"${SCHEMA_DIR}"/columnstore_schema.sql &>/dev/null; then
    echo -e "Creating 'columnstore_bts' schema..." "${GREEN}done${NC}"
else
    echo -e "Creating 'columnstore_bts' schema..." "${RED}fail${NC}"
    exit 1
fi
# Loading data into ColumnStore tables.
for CSV_FILE in ${NAME1} ${NAME2} ${NAME3}; do
    echo -e "\nLoading '${CSV_FILE}.csv' with cpimport ..."
    if ! $CPIMPORT -m 1 -s ',' -E "" columnstore_bts "${CSV_FILE}" -l
"${SCHEMA_DIR}/${CSV_FILE}.csv"; then
        echo -e "loading '${CSV_FILE}.csv' ... ${RED}fail${NC}"
        exit 1
    fi
done
printf "\nDo you want to include an InnoDB comparison schema? (Y/N) "
read -r COMPARE
if [[ $COMPARE == [yY] ]]; then
    if $MARIADB <"${SCHEMA_DIR}"/innodb_schema.sql &>/dev/null; then
        echo -e "\nCreating 'innodb_bts' schema..." "${GREEN}done${NC}\n"
        for CSV_FILE in "${NAME1}" "${NAME2}" "${NAME3}"; do
            echo -e "\nLoading '${CSV_FILE}.csv' with LDI ..."
            if ! $MARIADB --database="innodb_bts" --init-command="SET sql_mode=''"
"LOAD DATA INFILE '${SCHEMA_DIR}/${CSV_FILE}.csv'
INTO TABLE ${CSV_FILE}
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '\"
LINES TERMINATED BY '\n';" | awk '!/Bye/'; then
                echo -e "loading '${CSV_FILE}.csv' ... ${RED}fail${NC}"
                exit 1
            fi
        done
    else
        echo -e "Creating 'innodb_bts' schema..." "${RED}fail${NC}"
        exit 1
    fi
fi
elif [[ $COMPARE == [nN] ]]; then
```

```
    exit 0
else
    echo -e "Invalid input. Please enter Y or N.\n"
    exit 1
fi
```

### Запит 1. Розрахувати сумарну затримку по містах

```
SELECT origin AS city, SUM(arr_delay + dep_delay) AS total_delay
FROM flights
GROUP BY origin
UNION ALL
SELECT dest AS city, SUM(arr_delay + dep_delay) AS total_delay
FROM flights
GROUP BY dest;
```

**Колумнарна БД:**

*1 запит: 0,484 сек. \*/*

**Звичайна:**

*запит: 5,922 сек. \*/*

### Запит 2. Порахувати кількість польотів по містах

```
SELECT origin AS city, COUNT(*) AS flight_count
FROM flights
GROUP BY origin
UNION ALL
SELECT dest AS city, COUNT(*) AS flight_count
FROM flights
GROUP BY dest;
```

**Колумнарна БД:**

*запит: 0,360 сек.*

**Звичайна:**

*запит: 3,625 сек.*

### Запит 3. Знайти місто з найменшою і найбільшою затримкою

```
SELECT
    city,
    CASE
        WHEN avg_delay IS NULL THEN 'Немає даних'
        ELSE avg_delay
    END AS avg_delay
FROM (
    SELECT
        city,
        AVG(arr_delay) AS avg_delay
    FROM flights
    JOIN airports ON flights.origin = airports.iata_code
    GROUP BY city
    ORDER BY avg_delay DESC
    LIMIT 1
```

```

) AS min_delay_city
UNION ALL
SELECT
  city,
  CASE
    WHEN avg_delay IS NULL THEN 'Немає даних'
    ELSE avg_delay
  END AS avg_delay
FROM (
  SELECT
    city,
    AVG(arr_delay) AS avg_delay
  FROM flights
  JOIN airports ON flights.origin = airports.iata_code
  GROUP BY city
  ORDER BY avg_delay ASC
  LIMIT 1

```

) AS max\_delay\_city

**Колумнарна БД:**

*запрос: 0,875 сек.*

**Звичайна:**

*запрос: 8,156 сек.*

**Запит 4. Знайти всі польоти з затримкою більше за середній час затримки**

```

SELECT *
FROM flights
WHERE (arr_delay + dep_delay) > (
  SELECT AVG(arr_delay + dep_delay)
  FROM flights
);

```

**Колумнарна БД:**

*запрос: 0,250 сек.*

**Звичайна:**

*запрос: 2,844 сек.*

**Виміри**

Запит №	Стовпчикова БД	Звичайна БД
1	0,484	5,922
2	0,360	3,625
3	0,875	8,156
4	0,250	2,844

Загалом, швидкість виконання запитів у стовпчиковій БД швидше приблизно в 10 разів, що значно покращує роботу з базою даних.

**Об'єм БД**

Виміряємо об'єм колумнарної бази даних:

call columnstore\_info.total\_usage();

TOTAL_DATA_SIZE	TOTAL_DISK_USAGE
232.76 MB	61.00 MB

Після цього виміряємо об'єм звичайної БД за допомогою команди:

```
sh-4.4# du innodb_bts -h
413M    innodb_bts
```

Як бачимо, стовпчаста база даних займає значно менше простору на жорсткому диску.