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

Звіт № 1 з дисципліни «Програмування інтелектуальних інформаційних систем»

Виконав студент П	П-13 Романюк Діана Олексіївна	
	(шифр, прізвище, ім'я, по батькові)	
Перевірив	Баришич Лука Маріянович	

( прізвище, ім'я, по батькові)

## Лабораторна робота 1

## Постановка задачі:

- 1. Створити стовпчикову і звичайну бд
- 2. Розрахувати сумарну затримку по містах
- 3. Порахувати кількість польотів по містах
- 4. Знайти місто з найменшою і найбільшою затримкою
- 5. Знайти всі польоти з затримкою більше за середній час затримки
- 6. Заміряти вбудованими методами об"єм БД та швидкість виконання запитів. Порівняти звичайну і стовпчикову

## Завдання 1

Створення звичайної та стовпчикової бд

```
key `mas_delay` (`mas_delay`),
key `security_delay` (`security_delay`),
key `late_aircraft_delay` (`late_aircraft_delay`),
key 'arr_delay' ('arr_delay'),
key 'month' ('month'),
key 'dest' ('dest')
```

```
`dep_delay` decimal(13,2) default null,
    `taxi_out` decimal(13,2) default null,
    wheels_off` varchar(4) default null,
    wheels_on` varchar(4) default null,
    `taxi_in` decimal(13,2) default null,
    `crs_arr_time` varchar(4) default null,
    `arr_time` varchar(4) default null,
    `arr_delay` decimal(13,2) default null,
    `cancelled` decimal(13,2) default null,
    `cancelled` decimal(13,2) default null,
    `cancellation_code` varchar(20) default null,
    `crs_elapsed_time` decimal(13,2) default null,
    `actual_elapsed_time` decimal(13,2) default null,
    `air_time` decimal(13,2) default null,
    `carrier_delay` decimal(13,2) default null,
    `veather_delay` decimal(13,2) default null,
    `nas_delay` decimal(13,2) default null,
    `securrity_delay` decimal(13,2) default null,
    `securrity_delay` decimal(13,2) default null,
    `late_aircraft_delay` decimal(13,2) default null)
```

## Завантаження даних

```
USE `innodb_bts`;
USE `columnstore_bts`;

TRUNCATE TABLE airlines;
LOAD DATA INFILE '../../../data/airlines.csv'
    INTO TABLE airlines
    FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

TRUNCATE TABLE airports;
LOAD DATA INFILE '../../../data/airports.csv'
    INTO TABLE airports
    FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

TRUNCATE TABLE flights;
SET SESSION sql_mode = '';
LOAD DATA INFILE '../../../data/flights.csv'
    INTO TABLE flights
    FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
    LINES TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
    LINES TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '\n';
```

## Завдання 2

Розрахунок сумарної затримки по містах

```
-- task2
-- total delay for each city
SELECT a.city, SUM(f.arr_delay) AS 'total delay'
FROM flights AS f
JOIN airports AS a ON f.dest = a.iata_code
GROUP BY a.city;
```

	<b>■</b> city	<b>■</b> `total delay` ÷	
	Aberdeen	-525.00	
	Abilene	5952.00	
	Adak	-293.00	
	Agana	994.00	
	Aguadilla	2634.00	
	Akron	6390.00	
	Albany	12775.00	
8	Albuquerque	18043.00	

	<b>I</b> city	<b>\$</b>	<b>I</b> ∄ `total delay` ‡
1	Boston		241473.00
2	Chicago		804105.00
3	Ft. Myers		36346.00
4	Phoenix		100955.00

## Завдання 3

# Розрахунок кількості польотів по містах

```
-- task3
-- Count the number of flights for each city
SELECT a.city, COUNT(*) AS 'amount of flights'
FROM flights f
JOIN airports a ON f.dest = a.iata_code
GROUP BY a.city;
```

	I≣ city ▲	■ `amount of flights` ÷
	Aberdeen	121
	Abilene	568
	Adak	21
	Agana	65
5	Aguadilla	224
6	Akron	1282
	Albany	1523
	Albuquerque	3462

			_	′				
	<b>■</b> city	<b>‡</b>		圃	`amount	of	flights	<b>,</b>
1	Chicago						69	097
2	Baltimore						15	169
3	Ft. Lauderdale						16	717
4	Raleigh						5	947
_	New Onleans						-	7//5

## Завдання 4

## Знайти місто з найменшою і найбільшою затримкою

```
-- task4
-- city with the highest and lowest delays
-- city with the lowest delay

SELECT a.city, f.arr_delay

FROM airports a

JOIN flights f ON a.iata_code = f.dest

WHERE f.arr_delay = (SELECT MIN(arr_delay) FROM flights)
```

```
■ city  
■ arr_delay 

1 Seattle -82.00
```

```
ORDER BY f.arr_delay;

-- city with the highest delay

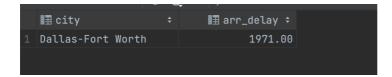
SELECT a.city, f.arr_delay

FROM airports a

JOIN flights f ON a.iata_code = f.dest

WHERE f.arr_delay = (SELECT MAX(arr_delay) FROM flights)

ORDER BY f.arr delay;
```



### Завдання 5

Знайти всі польоти з затримкою більше за середній час затримки

```
-- task5
-- find flights with a delay greater than the average delay
-- calculate and display the average delay
SELECT AVG(arr_delay) AS 'Average Delay'
FROM flights;

SELECT fl_date, tail_num, fl_num, dest, arr_delay
FROM flights
WHERE arr_delay > (SELECT AVG(arr_delay) FROM flights);
```

```
| SELECT AVG(arr_delay) AS 'Average Delay'
| FROM flights;
| SELECT fl_date, tail_num, fl_num, dest, arr_delay
| FROM flights
| AVG(arr_delay) FROM flights);
| FROM flights | | FROM
```

## Завдання 6

Заміряти вбудованими методами об'єм БД та швидкість виконання запитів.

```
-- task6
-- database size

CALL columnstore_info.table_usage('columnstore_bts', NULL);

SELECT
    table_schema AS 'Database',
    SUM(data_length + index_length) / 1024 / 1024 AS 'Database Size (MB)'

FROM information_schema.tables

WHERE table_schema = 'innodb_bts'

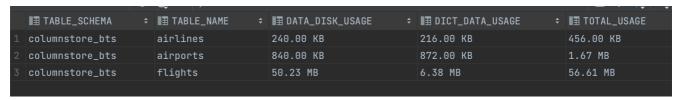
GROUP BY table_schema;
```

#### Висновок:

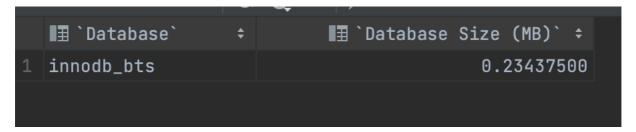
Порівняємо звичайну і стовпчикову бд

#### Об'єм

Стовпчикова бд colmnstore\_bts:



## Innodb bts:



### Швидкість виконання запитів

Стовпчикова бд colmnstore bts:

```
₌ab1.sql 73 ms
```

## Innodb bts:

