**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

«Програмування інтелектуальних інформаційних систем»

„**Стовпчакові бази даних**”

**Виконав(ла)**

(шифр, прізвище, ім'я, по батькові)

*ІП-11 Головня Олександр Ростиславович*

**Перевірив**

(прізвище, ім'я, по батькові)

*Баришич Л. М.*

Київ 2023

1. **Розрахувати сумарну затримку по містах**

SELECT a.city, SUM(f.arr\_delay) AS total\_delay

FROM columnstore\_bts.flights f

JOIN columnstore\_bts.airports a ON f.dest = a.iata\_code

GROUP BY a.city ORDER BY a.city ASC LIMIT 10;

**Изображение выглядит как текст, Шрифт, снимок экрана, чек

Автоматически созданное описание**

**Рисунок – 1.1 Columnstore**

Изображение выглядит как текст, Шрифт, снимок экрана, чек

Автоматически созданное описание

**Рисунок – 1.2 Innodb**

**2. Порахувати кількість польотів по містах**

SELECT a.city, COUNT(\*) AS total\_flights

FROM columnstore\_bts.flights f

JOIN columnstore\_bts.airports a ON f.dest = a.iata\_code

GROUP BY a.city ORDER BY a.city LIMIT 10;

Изображение выглядит как текст, чек, снимок экрана, Шрифт

Автоматически созданное описание

**Рисунок – 2.1 Columnstore**

Изображение выглядит как текст, чек, снимок экрана, Шрифт

Автоматически созданное описание

**Рисунок – 2.2 Innodb**

**3. Знайти місто з найменшою і найбільшою затримкою**

SELECT a.city, MIN(f.arr\_delay) as min\_delay FROM columnstore\_bts.flights f JOIN columnstore\_bts.airports a ON f.dest = a.iata\_code GROUP BY a.city ORDER BY min\_delay LIMIT 1;

SELECT a.city, MAX(f.arr\_delay) as max\_delay FROM columnstore\_bts.flights f JOIN columnstore\_bts.airports a ON f.dest = a.iata\_code GROUP BY a.city ORDER BY max\_delay DESC LIMIT 1;

Изображение выглядит как текст, Шрифт, чек, алгебра

Автоматически созданное описаниеИзображение выглядит как текст, чек, Шрифт, белый

Автоматически созданное описание

**Рисунок – 3.1 Columnstore**

Изображение выглядит как текст, чек, Шрифт, белый

Автоматически созданное описание Изображение выглядит как текст, чек, Шрифт, белый

Автоматически созданное описание

**Рисунок – 3.2 Innodb**

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

SELECT f.\*

FROM columnstore\_bts.flights f

WHERE f.arr\_delay > (SELECT AVG(f.arr\_delay) FROM columnstore\_bts.flights f)

LIMIT 10;

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

**Рисунок – 4.1 Columnstore**

Изображение выглядит как текст, снимок экрана, линия, Шрифт

Автоматически созданное описание

**Рисунок – 4.2 Innodb**

**5. Заміряти вбудованими методами об`єм БД та швидкість виконання запитів. Порівняти звичайну і стовпчикову**

CALL columnstore\_info.table\_usage('columnstore\_bts', NULL);

Изображение выглядит как текст, чек, Шрифт, снимок экрана

Автоматически созданное описание

**Рисунок – 5.1 Columnstore**

Об’єм стовпчикової близько 60 мб, а звичайної – 413 мб

|  |  |  |
| --- | --- | --- |
| ЗАПИТ | Стовпчакова | Звичайна |
| 1. | 0.571 sec | 7.202 sec |
| 2. | 0.344 sec | 1.558 sec |
| 3. | 0.385/0.349 sec | 5.109/4.926 sec |
| 4. | 0.594 sec | 1.078 sec |

Висновок: У ході проведеної лабораторної роботи я створив дві бази даних, а саме стовпчастої та звичайної. Виконав різні запити до обох видів баз даних та порівняв їхню продуктивність, вимірявши час виконання різних запитів і встановив, що ефективність стовпчастої в рази більша ніж звичайної, а об’єм є майже у 7 разів менший.

DROP DATABASE IF EXISTS `columnstore\_bts`;

CREATE DATABASE `columnstore\_bts`;

USE `columnstore\_bts`;

CREATE TABLE `airlines` (

`iata\_code` varchar(2) DEFAULT NULL,

`airline` varchar(30) DEFAULT NULL

) ENGINE=Columnstore DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3\_general\_ci;

CREATE TABLE `airports` (

`iata\_code` varchar(3) DEFAULT NULL,

`airport` varchar(80) DEFAULT NULL,

`city` varchar(30) DEFAULT NULL,

`state` varchar(2) DEFAULT NULL,

`country` varchar(30) DEFAULT NULL,

`latitude` decimal(11,4) DEFAULT NULL,

`longitude` decimal(11,4) DEFAULT NULL

) ENGINE=Columnstore DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3\_general\_ci;

CREATE TABLE `flights` (

`year` smallint(6) DEFAULT NULL,

`month` tinyint(4) DEFAULT NULL,

`day` tinyint(4) DEFAULT NULL,

`day\_of\_week` tinyint(4) DEFAULT NULL,

`fl\_date` date DEFAULT NULL,

`carrier` varchar(2) DEFAULT NULL,

`tail\_num` varchar(6) DEFAULT NULL,

`fl\_num` smallint(6) DEFAULT NULL,

`origin` varchar(5) DEFAULT NULL,

`dest` varchar(5) DEFAULT NULL,

`crs\_dep\_time` varchar(4) DEFAULT NULL,

`dep\_time` varchar(4) DEFAULT NULL,

`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,

`cancellation\_code` varchar(20) DEFAULT NULL,

`diverted` decimal(13,2) 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,

`distance` decimal(13,2) DEFAULT NULL,

`carrier\_delay` decimal(13,2) DEFAULT NULL,

`weather\_delay` decimal(13,2) DEFAULT NULL,

`nas\_delay` decimal(13,2) DEFAULT NULL,

`security\_delay` decimal(13,2) DEFAULT NULL,

`late\_aircraft\_delay` decimal(13,2) DEFAULT NULL

) ENGINE=Columnstore DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3\_general\_ci;  
  
  
  
  
  
  
  
  
DROP DATABASE IF EXISTS `innodb\_bts`;

CREATE DATABASE `innodb\_bts`;

USE `innodb\_bts`;

CREATE TABLE `airlines` (

`iata\_code` varchar(2) NOT NULL,

`airline` varchar(30) DEFAULT NULL,

PRIMARY KEY (`iata\_code`),

KEY `airline` (`airline`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3\_general\_ci;

CREATE TABLE `airports` (

`iata\_code` varchar(3) NOT NULL,

`airport` varchar(80) DEFAULT NULL,

`city` varchar(30) DEFAULT NULL,

`state` varchar(2) DEFAULT NULL,

`country` varchar(30) DEFAULT NULL,

`latitude` decimal(11,4) DEFAULT NULL,

`longitude` decimal(11,4) DEFAULT NULL,

PRIMARY KEY (`iata\_code`),

KEY `state` (`state`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3\_general\_ci;

CREATE TABLE `flights` (

`year` smallint(6) DEFAULT NULL,

`month` tinyint(4) DEFAULT NULL,

`day` tinyint(4) DEFAULT NULL,

`day\_of\_week` tinyint(4) DEFAULT NULL,

`fl\_date` date DEFAULT NULL,

`carrier` varchar(2) DEFAULT NULL,

`tail\_num` varchar(6) DEFAULT NULL,

`fl\_num` smallint(6) DEFAULT NULL,

`origin` varchar(5) DEFAULT NULL,

`dest` varchar(5) NOT NULL,

`crs\_dep\_time` varchar(4) DEFAULT NULL,

`dep\_time` varchar(4) DEFAULT NULL,

`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,

`cancellation\_code` varchar(20) DEFAULT NULL,

`diverted` decimal(13,2) 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,

`distance` decimal(13,2) DEFAULT NULL,

`carrier\_delay` decimal(13,2) DEFAULT NULL,

`weather\_delay` decimal(13,2) DEFAULT NULL,

`nas\_delay` decimal(13,2) DEFAULT NULL,

`security\_delay` decimal(13,2) DEFAULT NULL,

`late\_aircraft\_delay` decimal(13,2) DEFAULT NULL,

KEY `carrier` (`carrier`),

KEY `year` (`year`),

KEY `carrier\_delay` (`carrier\_delay`),

KEY `weather\_delay` (`weather\_delay`),

KEY `nas\_delay` (`nas\_delay`),

KEY `security\_delay` (`security\_delay`),

KEY `late\_aircraft\_delay` (`late\_aircraft\_delay`),

KEY `arr\_delay` (`arr\_delay`),

KEY `month` (`month`),

KEY `dest` (`dest`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3\_general\_ci;