



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИИТ)

Кафедра прикладной математики (ПМ)

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1

по дисциплине «Языки программирования для статистической обработки
данных»

Студент группы *ИМБО-11-23. Журавлев Ф.А.*

(подпись)

Преподаватель *Трушин СМ*

(подпись)

Москва 2025 г.

Содержание

1. Введение.....	3
Цель практики:.....	3
Задачи практики:.....	3
Установка и настройка окружения R, Python, Glarus BI.....	5
2. Результаты работы.....	7
Работа в Glarus BI:.....	7
Посмотрим на реализацию этого алгоритма на языке Python.....	9
Реализация в R:.....	10
3. Выводы:.....	13

Введение

Цель практики:

Научиться настраивать рабочее окружение для работы с Python, R и Glarus BI, а также выполнять базовые операции с данными в каждом из инструментов.

Задачи практики:

1. Установить и настроить окружение: Python (VSCode), R (VSCode), подключение к Glarus BI.
2. Изучить интерфейс Glarus BI, основные элементы и инструменты для анализа данных.
3. Загрузить предоставленный набор данных в Glarus BI.
4. Выполнить базовые операции: Отфильтровать данные по определённым условиям. Рассчитать среднее значение для одного из столбцов.
5. Провести аналогичные операции в Python и R: загрузка данных, подсчёт среднего значения.

Установка и настройка окружения R, Python, Glarus BI

Настройка Python (VSCode):

1. Откроем VScode, запустим программу, убедимся, что библиотеки установлены.

Листинг 1.1. Команды для проверки установки библиотек

```
import pandas as pd
import numpy as np

print("Библиотеки установлены")
```

Настройка R (VSCode):

Установим необходимые для анализа данных библиотеки

dplyr и **readr**.

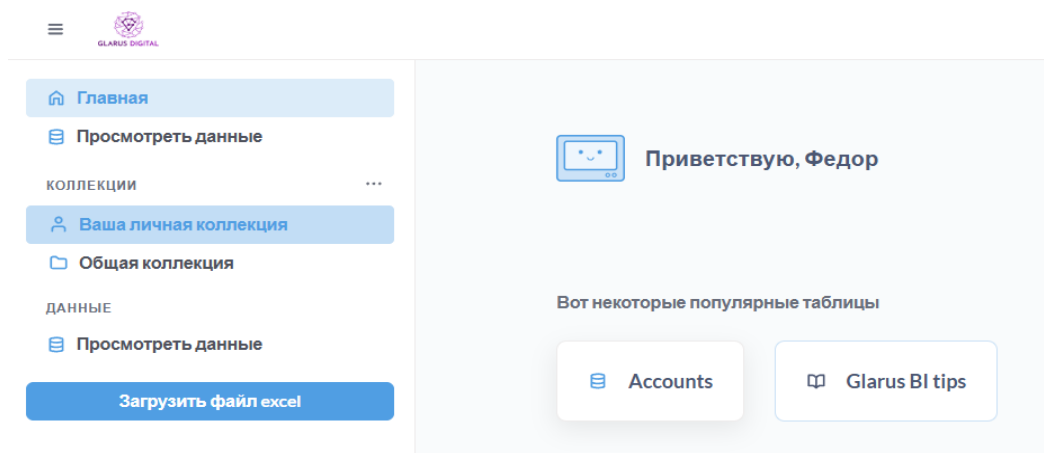
Листинг 1.2. Установка библиотек dplyr и readr

```
VSCode R Session Watcher requires jsonlite, rlang. Please install manually in order to use VSCode-R.  
> install.packages("dplyr")  
--- Please select a CRAN mirror for use in this session ---  
also installing the dependencies 'utf8', 'fansib', 'pkgconfig', 'withr', 'cli', 'generics', 'glue', 'lifecycle', 'magrittr', 'pillar', 'R6',  
'rlang', 'tibble', 'tidyselect', 'vctrs'  
  
trying URL 'https://cran.ma.imperial.ac.uk/bin/windows/contrib/4.4/utf8_1.2.4.zip'  
Content type 'application/zip' length 151107 bytes (147 KB)  
downloaded 147 KB  
  
trying URL 'https://cran.ma.imperial.ac.uk/bin/windows/contrib/4.4/fansib_1.0.6.zip'  
Content type 'application/zip' length 324393 bytes (316 KB)  
downloaded 316 KB  
  
trying URL 'https://cran.ma.imperial.ac.uk/bin/windows/contrib/4.4/pkgconfig_2.0.3.zip'  
Content type 'application/zip' length 23132 bytes (22 KB)  
downloaded 22 KB  
  
trying URL 'https://cran.ma.imperial.ac.uk/bin/windows/contrib/4.4/withr_3.0.2.zip'
```

Настройка Glarus BI:

1. Откроем веб-интерфейс Glarus BI.
2. Войдем в систему с учётными данными, предоставленными администратором.
3. Проверим доступ к функциям загрузки данных и создания отчётов.

Листинг 1.3. Авторизация в Glarus BI



Результаты работы

Работа в Glarus BI:

Загрузим нашу таблицу в формате .xls для работы с ней в Glarus BI:

Листинг 2.1.1. Загрузка набора данных на тему «Количество сотрудников в предприятиях»

ClickHouse_Test / empl / empl

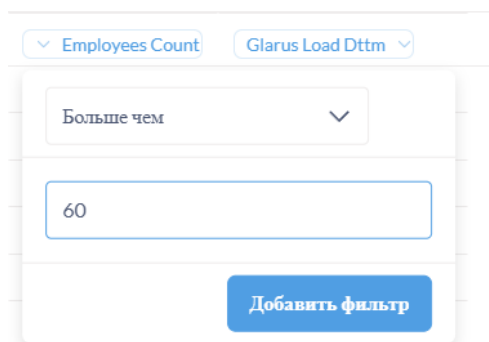
Tin	Year	Employees Count	Glarus Load Dttm
100 002 516	2 022	1	21 февраля, 2025, 21:52
105 026 430	2 018	140	21 февраля, 2025, 21:52
105 026 430	2 019	126	21 февраля, 2025, 21:52
105 026 430	2 020	128	21 февраля, 2025, 21:52
105 026 430	2 021	124	21 февраля, 2025, 21:52
105 026 430	2 022	116	21 февраля, 2025, 21:52
105 028 050	2 018	19	21 февраля, 2025, 21:52
105 028 050	2 019	19	21 февраля, 2025, 21:52
105 028 050	2 020	17	21 февраля, 2025, 21:52
105 028 050	2 021	31	21 февраля, 2025, 21:52
105 028 050	2 022	14	21 февраля, 2025, 21:52
105 041 910	2 018	22	21 февраля, 2025, 21:52
105 041 910	2 019	28	21 февраля, 2025, 21:52
105 041 910	2 020	30	21 февраля, 2025, 21:52
105 041 910	2 021	33	21 февраля, 2025, 21:52
105 041 910	2 022	38	21 февраля, 2025, 21:52
105 048 970	2 018	7	21 февраля, 2025, 21:52
105 048 970	2 019	1	21 февраля, 2025, 21:52

Визуализация

Фильтрация данных:

Выполним фильтрацию данных на примере количества сотрудников в предприятии. Выделим интересующий нас столбец «Employees Count», затем выберем «больше или равно» , поставим в поле значение «60» и нажмем «добавить фильтр».

Листининг 2.1.2. Фильтрация данных



Листининг 2.1.2. Результат фильтрации.

▼ Tin	▼ Year	▼ Employees Count	Glarus Load Dttm ▼
2 147 483 647	2018	603	21 февраля, 2025, 21:52
2 147 483 647	2019	595	21 февраля, 2025, 21:52
2 147 483 647	2020	576	21 февраля, 2025, 21:52
2 147 483 647	2021	554	21 февраля, 2025, 21:52
2 147 483 647	2022	525	21 февраля, 2025, 21:52
2 147 483 647	2018	83	21 февраля, 2025, 21:52
2 147 483 647	2021	63	21 февраля, 2025, 21:52
2 147 483 647	2022	68	21 февраля, 2025, 21:52
2 147 483 647	2018	249	21 февраля, 2025, 21:52
2 147 483 647	2019	240	21 февраля, 2025, 21:52
2 147 483 647	2020	232	21 февраля, 2025, 21:52
2 147 483 647	2021	236	21 февраля, 2025, 21:52
2 147 483 647	2018	115	21 февраля, 2025, 21:52
2 147 483 647	2019	128	21 февраля, 2025, 21:52
2 147 483 647	2020	154	21 февраля, 2025, 21:52
2 147 483 647	2021	159	21 февраля, 2025, 21:52
2 147 483 647	2022	145	21 февраля, 2025, 21:52

Подсчет медианы:

Посчитаем средний балл по математике воспользовавшись специальной функцией Glarus BI:

Листининг 2.1.3. Среднее количество сотрудников

Average of Employees Count
ClickHouse_Test / empl / empl

13,15

Скрипты Python и R для загрузки данных и подсчёта среднего значения.

Посмотрим на реализацию этого алгоритма на языке Python.

Загрузим в Google Colab наш .csv файл прописав специальный код.

Листининг 2.2.1. Среднее количество сотрудников

```
import pandas as pd
import numpy as np

df = pd.read_csv("empl.csv")
print(df.head())
```

Посмотрим на таблицу, которую нам вывело:

Листининг 2.2.2. Таблица в Python

```
tin;year;employees_count
0      0100002516;2022;1
1      0105026430;2018;140
2      0105026430;2019;126
3      0105026430;2020;128
4      0105026430;2021;124
```

Напишем код, который поможет нам отсортировать количество сотрудников на предприятии, которые больше или равно 60.

Листинг 2.2.3. Фильтрация в Python

```
import pandas as pd
import numpy as np

df = pd.read_csv("empl.csv", sep=';')
df[df['employees_count'] >= 60]
```

✓ 0.0s

	tin	year	employees_count
1	105026430	2018	140
2	105026430	2019	126
3	105026430	2020	128
4	105026430	2021	124
5	105026430	2022	116
...
84694	9729025828	2021	79
84764	9729149975	2019	61
84916	9731045772	2020	78
84917	9731045772	2021	83
84918	9731045772	2022	85

листинг 2.2.4. Среднее количество сотрудников

```
avg = df['employees_count'].mean()
print(avg, "среднее количество сотрудников")
```

[10] ✓ 0.0s

... 13.1524548107701 средний балл

Как видно, средний балл такой же, какой и получился в **GLARUS BI**

Реализация на R:

Пропишем код на R, благодаря которому сможем загрузить наш файл .CSV.

Листинг 2.2.5. Загрузка в R

```
1 library(dplyr)
2 library(readr)
3 df <- read.csv("empl.csv", sep=';')
```

После загрузки нашего датасета, напишем код, который отсортирует нашу таблицу по количеству сотрудников, которых больше или равно 60.

Листинг 2.2.6. Фильтрация данных.

```
library(dplyr)
library(readr)
df <- read.csv("empl.csv", sep=';')
f_df <- filter(df, employees_count >= 60)
print(f_df)
```

Листинг 2.2.7. Отфильтрованная таблица в R

3029	7113010436	2018	61
3030	7113010436	2020	60
3031	7113010436	2022	60
3032	7114001321	2018	380
3033	7114001321	2019	384
3034	7114001321	2020	393
3035	7114001321	2021	373
3036	7114023580	2018	70
3037	7115008873	2019	72
3038	7115008873	2020	86
3039	7115008873	2021	82
3040	7115008873	2022	71
3041	7115021930	2018	82
3042	7115021930	2019	82
3043	7115021930	2020	77
3044	7115021930	2021	64
3045	7116018352	2018	77
3046	7116018352	2019	71
3047	7116018352	2020	68
3048	7116018352	2021	68
3049	7116018352	2022	69
3050	7116132930	2018	79

И напишем код, благодаря которому сможем оценить медиану:

Листинг 2.2.8. Медиана в R

```
1 library(dplyr)
2 library(readr)
3 df <- read.csv("empl.csv", sep=';')
4 f_df <- filter(df, employees_count >= 60)
5 print(f_df)
6 avg <- mean(df$employees_count, na.rm = TRUE)
7 print(avg)
```

```
[1] 13.15245
```

По итогам все значения совпали на всех трех платформах.

Выводы

В результате первой практической работы был проведен анализ набора данных «Количество сотрудников на предприятиях» на трех разных платформах: R, Python, Glarus BI. Каждый из них имеет свои достоинства. Код на R более понятный и легкий, нежели на питоне, но в Glarus BI ничего не надо писать и для такой простой обработки он самый быстрый и эффективный вариант. Python в свою очередь имеет самый привычный для обывателя код, который будет понятен всем, кто чуть-чуть с ним знаком.