



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 №2
по дисциплине «Языки программирования для статистической обработки
данных»

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

(подпись)

Преподаватель *Трушин С.М.*

(подпись)

Москва 2025 г.

1 ЦЕЛЬ И ЗАДАЧИ

Цель практической работы:

Изучить методы загрузки и очистки данных в Python, R и Glarus BI, а также освоить базовые инструменты для подготовки данных к анализу.

Задачи практической работы:

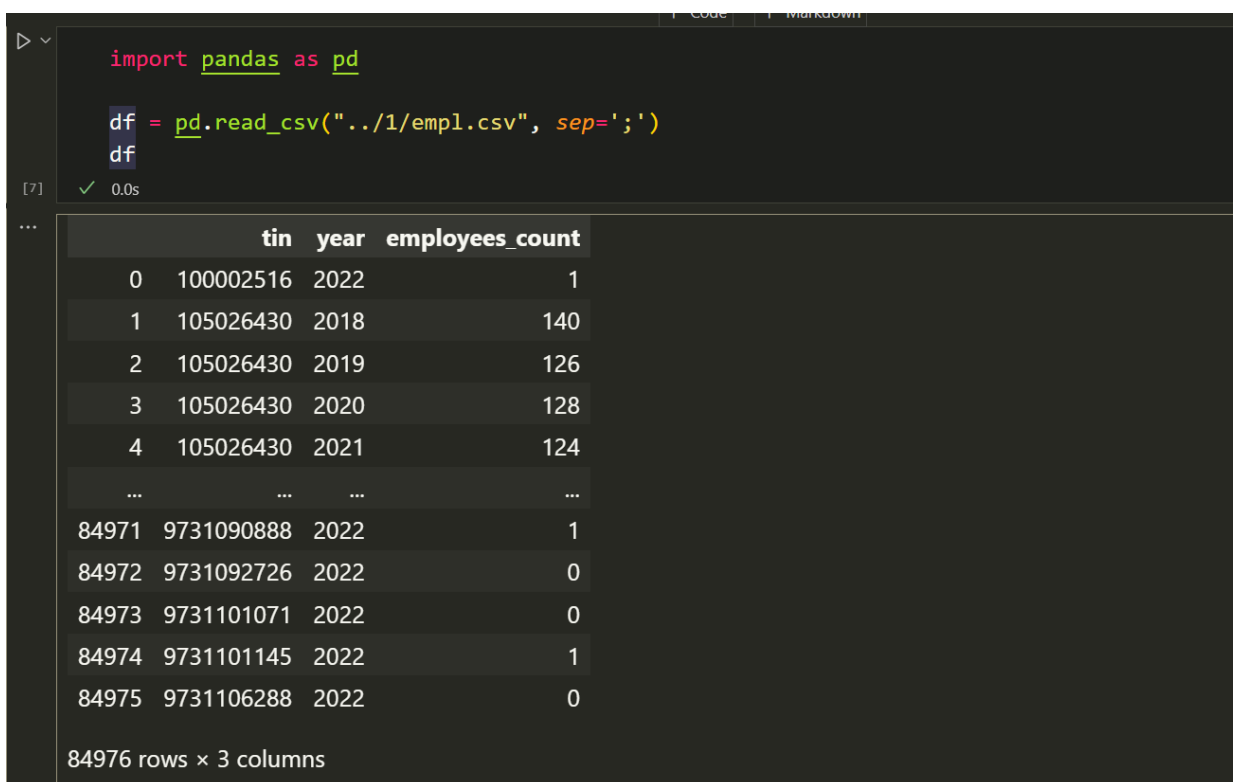
- 1) Загрузить данные из различных источников (CSV, Excel, базы данных) в Python, R
- 2) Выполнить очистку данных:
 - Обнаружить и обработать пропуски (удаление, замена на среднее/медиану).
 - Обнаружить и удалить дубликаты.
 - Преобразовать типы данных (например, текст в даты).
- 3) Сравнить подходы к очистке данных в Python, R.

2 РЕЗУЛЬТАТЫ ПРАКТИКИ

Шаг 1) Загрузка данных из различных источников

1.1) Загрузка данных в Python (VS Code):

Рисунок 1.1 — загрузка данных в Python



```
import pandas as pd

df = pd.read_csv("../1/empl.csv", sep=';')
df
```

[7] ✓ 0.0s

	tin	year	employees_count
0	100002516	2022	1
1	105026430	2018	140
2	105026430	2019	126
3	105026430	2020	128
4	105026430	2021	124
...
84971	9731090888	2022	1
84972	9731092726	2022	0
84973	9731101071	2022	0
84974	9731101145	2022	1
84975	9731106288	2022	0

84976 rows × 3 columns

Загрузили исходную таблицу данных в формате .csv.

Рисунок 1.2 — загрузка данных в Python

```
ds1 = pd.read_excel("../1/empl.xls", engine='xlrd')
ds1
```

[13] ✓ 0.8s

	tin	year	employees_count
0	100002516	2022	1
1	105026430	2018	140
2	105026430	2019	126
3	105026430	2020	128
4	105026430	2021	124
...
65530	7713264418	2020	10
65531	7713264418	2021	8
65532	7713264418	2022	7
65533	7713320197	2018	1
65534	7713320197	2019	1

65535 rows × 3 columns

Загрузка исходного набора данных в формате .xls

Рисунок 1.3 — загрузка данных в Python

```
import sqlite3
conn = sqlite3.connect('data.db')
cursor = conn.cursor() # позволяет выполнять sql-запросы
cursor.execute('SELECT * FROM employees') # sql-запрос
rows = cursor.fetchall() # извлечение строк по результатам работы предыдущего запроса
for row in rows:
    print(row)
conn.close()
```

✓ 0.4s

```
('0100002516', '2022', '1')
('0105026430', '2018', '140')
('0105026430', '2019', '126')
('0105026430', '2020', '128')
('0105026430', '2021', '124')
('0105026430', '2022', '116')
('0105028050', '2018', '19')
('0105028050', '2019', '19')
('0105028050', '2020', '17')
('0105028050', '2021', '31')
('0105028050', '2022', '14')
('0105041910', '2018', '22')
('0105041910', '2019', '28')
('0105041910', '2020', '30')
('0105041910', '2021', '33')
```

Загрузка исходного набора данных в формате SQLite

1.2) Загрузка данных в R (VS Code)

рисунок 1.4 — код для загрузки .csv файла в RStudio

```
2 > R 2.r > ...
1 library(dplyr)
2 library(readr)
3 ds <- read.csv("2/empl.csv", sep=';')
4 print(ds.head())
```

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

```
33317 5022558060 2020 29
33318 5022558060 2021 27
33319 5022558060 2022 27
33320 5022558951 2018 39
33321 5022558951 2019 33
33322 5022558951 2020 29
33323 5022558951 2021 27
33324 5022558951 2022 28
33325 5022559120 2018 52
33326 5022559120 2019 54
33327 5022559120 2020 55
33328 5022559120 2021 68
33329 5022559120 2022 63
33330 5022560735 2018 5
33331 5022560735 2019 5
33332 5022560735 2020 5
33333 5022560735 2021 5
```

Шаг 2) Очистка данных

2.1) Очистка данных в Python:

Найдем количество пустых строк в нашем наборе данных:

Рисунок 2.1.1 – количество пустых строк

```
df.isnull().sum()
[2] ✓ 0.4s
... tin 0
    year 0
    employees_count 0
    dtype: int64
```

2.2) Очистка данных в R:

Рисунок 2.2.1 – находимые пустые строки

```
1 library(dplyr)
2 library(readr)
3 ds <- read.csv("2/empl.csv", sep=';')
4 print(colSums(is.na(ds)))
```

Рисунок 2.2.2 — Набор данных с пустыми строками.

```
tin      year employees_count
0        0                0
```

3 ВЫВОДЫ

Результате практической работы была произведена очистка данных, заменены пустые строки, удалены дубликаты и т. д. В Python чуть-чуть удобнее работать с этим, так как код более тривиальный, нежели в R, но у R есть огромный плюс — очень удобная работа с таблицами и установкой нужных пакетов напрямую из директории программы. Glarus BI неудобен из-за отсутствия нормального функционала, лоу-код щит.