ГУАП

КАФЕДРА № 41

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

ассистент		В.В. Боженко
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

Предварительный анализ данных

по курсу: Введение в анализ данных

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 4917		Е.А. Ясиновский
	подпись, дата	инициалы, фамилия

Санкт-Петербург 2022

Цель работы: осуществить предварительную обработку данных csv файла, выявить и устранить проблемы в данных, построить сводные таблицы по предоставленным данным

Вариант 2: Файл salary.csv, в котором предоставлены данные о заработной плате разработчиков и инженеров IT сферы разделенные по годам, опыту работы, типу занятости (полная\неполная), валюте, размеру компании и стране компании.

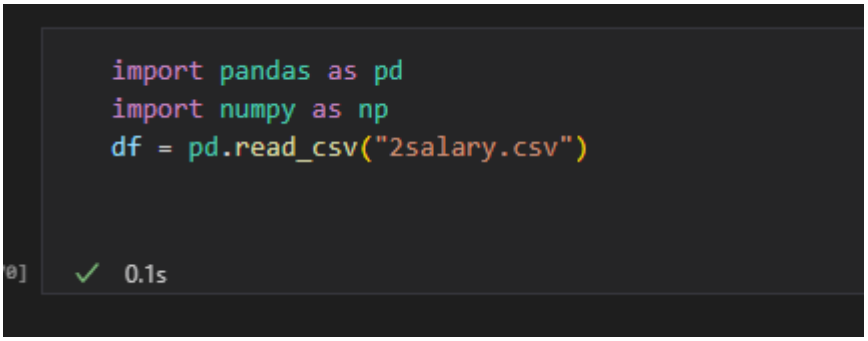


Рисунок 1 — Загрузил данные с помощью pandas

```
df.head(20)
```
✓ 0.7s Python

	Unnamed: 0	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	0	2020	MI	FT	Data Scientist	70000.0	EUR	79833	DE	0	DE	L
1	1	2020	SE	FT	Machine Learning Scientist	260000.0	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000.0	GBP	109024	GB	50	GB	M
3	3	2020	MI	FT	Product Data Analyst	20000.0	USD	20000	HN	0	HN	S
4	4	2020	SE	FT	Machine Learning Engineer	150000.0	USD	150000	US	50	US	L
5	5	2020	EN	FT	Data Analyst	72000.0	USD	72000	US	100	US	L
6	6	2020	SE	FT	Lead Data Scientist	190000.0	USD	190000	US	100	US	S
7	7	2020	MI	FT	Data Scientist	11000000.0	HUF	35735	HU	50	HU	L
8	8	2020	MI	FT	Business Data Analyst	135000.0	USD	135000	US	100	US	L
9	9	2020	SE	FT	Lead Data Engineer	125000.0	USD	125000	NZ	50	NZ	S
10	10	2020	EN	FT	Data Scientist	45000.0	EUR	51321	FR	0	FR	S
11	11	2020	MI	FT	Data Scientist	3000000.0	INR	40481	IN	0	IN	L
12	12	2020	EN	FT	Data Scientist	35000.0	EUR	39916	FR	0	FR	M
13	13	2020	MI	FT	Lead Data Analyst	87000.0	USD	87000	US	100	US	L
14	14	2020	MI	FT	Data Analyst	85000.0	USD	85000	US	100	US	L
15	15	2020	MI	FT	Data Analyst	8000.0	USD	8000	PK	50	PK	L
16	16	2020	EN	FT	Data Engineer	4450000.0	JPY	41689	JP	100	JP	S
17	17	2020	SE	FT	Big Data Engineer	100000.0	EUR	114047	PL	100	GB	S
18	18	2020	EN	FT	Data Science Consultant	423000.0	INR	5707	IN	50	IN	M
19	19	2020	MI	FT	Lead Data Engineer	56000.0	USD	56000	PT	100	US	M

Рисунок 2 — Вывел первые 20 строк

```
💡# Предметная область
Данные о заработной плате разработчиков и инженеров IT сферы разделенные по годам, опыту работы, типу занятости (полная\неполная), валюте, размеру компании и стране компании
#### Столбцы
1. Номер строки
2. Год выплаты ЗП
3. Опыт работы
4. Тип занятости
5. Наименование должности
6. ЗП выраженная в валюте выплаты
7. Валюта выплаты
8. Зп выраженная в долларах USD
9. Страна проживания сотрудника
10. Объем удаленной работы
11. Страна главного офиса компании
12. Размер компании

#### Оценка данных
Оценка данных производится с помощью метода .info() у объекта датафрейма
```

Markdown

Рисунок 3 — Выполнил обзор данных с помощью MD

```
df.info()
✓ 0.1s
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 610 entries, 0 to 609
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Unnamed: 0          610 non-null    int64
 1   work_year           610 non-null    int64
 2   experience_level    610 non-null    object
 3   employment_type     610 non-null    object
 4   job_title           610 non-null    object
 5   salary              607 non-null    float64
 6   salary_currency     609 non-null    object
 7   salary_in_usd       610 non-null    object
 8   employee_residence  610 non-null    object
 9   remote_ratio        610 non-null    int64
 10  company_location    610 non-null    object
 11  company_size        610 non-null    object
dtypes: float64(1), int64(3), object(8)
memory usage: 57.3+ KB
```

Рисунок 4 — Выполнение оценки данных с помощью метода info()

Изходя из полученного выше вывода можно увидеть, что всего у нас есть 610 строк данных, но сами данные корректны только в 607 строках, так как это минимальное число строк, где данные не равны null. В добавок можно заметить, что строковые значение здесь привелись к типу object, а у первого столбца (номера строки) остуствует имя

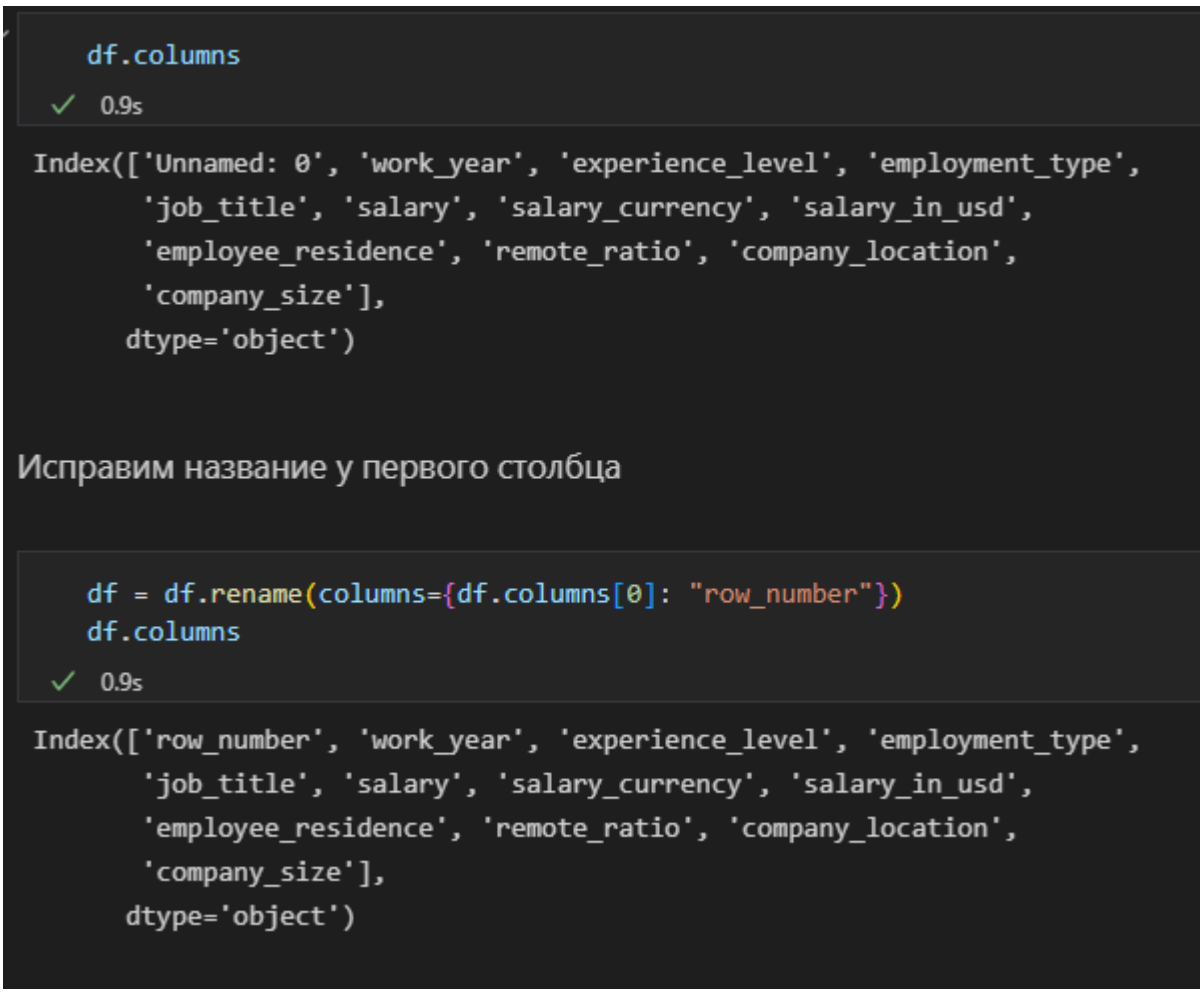


Рисунок 5 — Выявил проблемный столбец и исправил ошибку

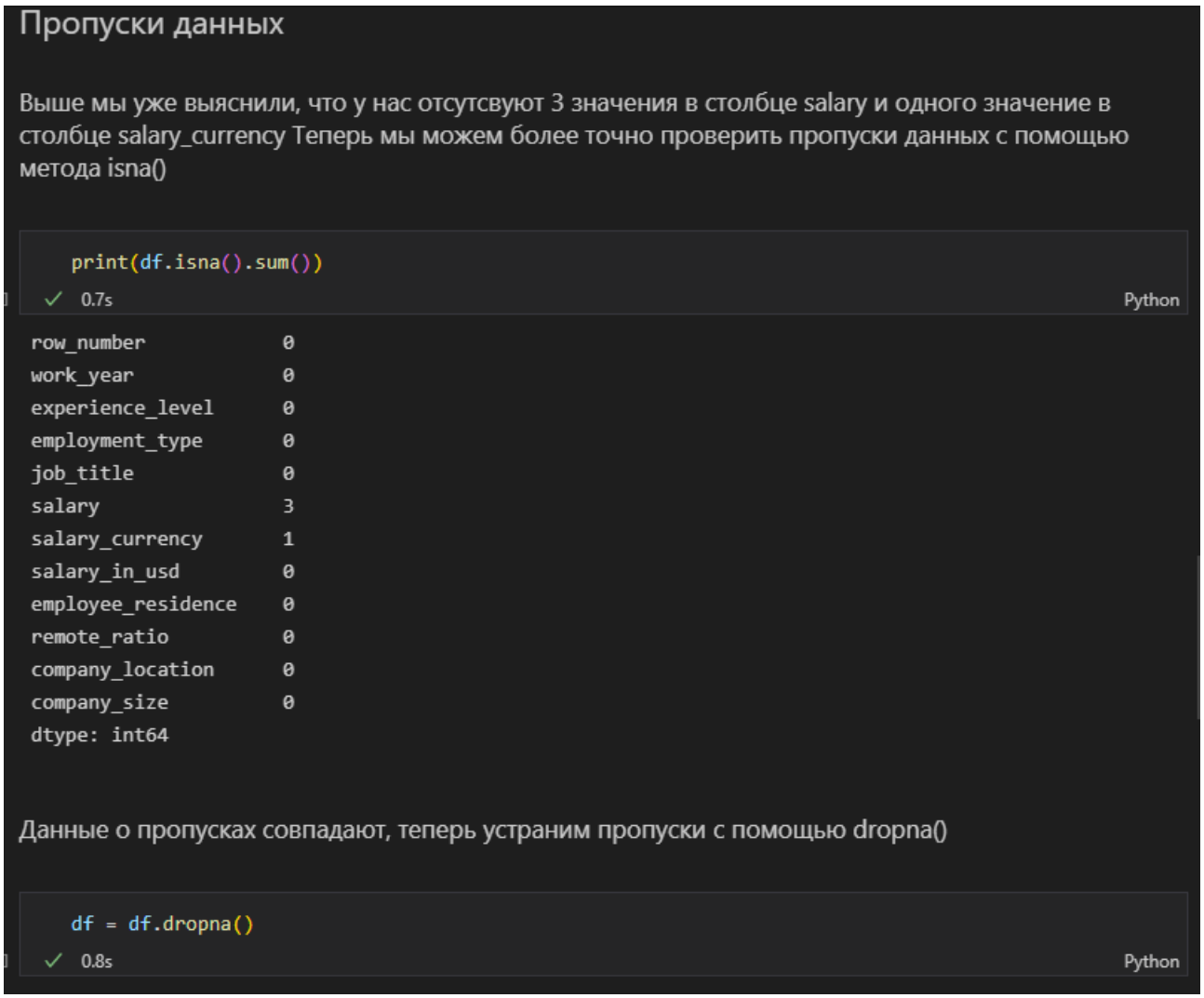


Рисунок 6 — Выявил пропуски в данных и устранил их

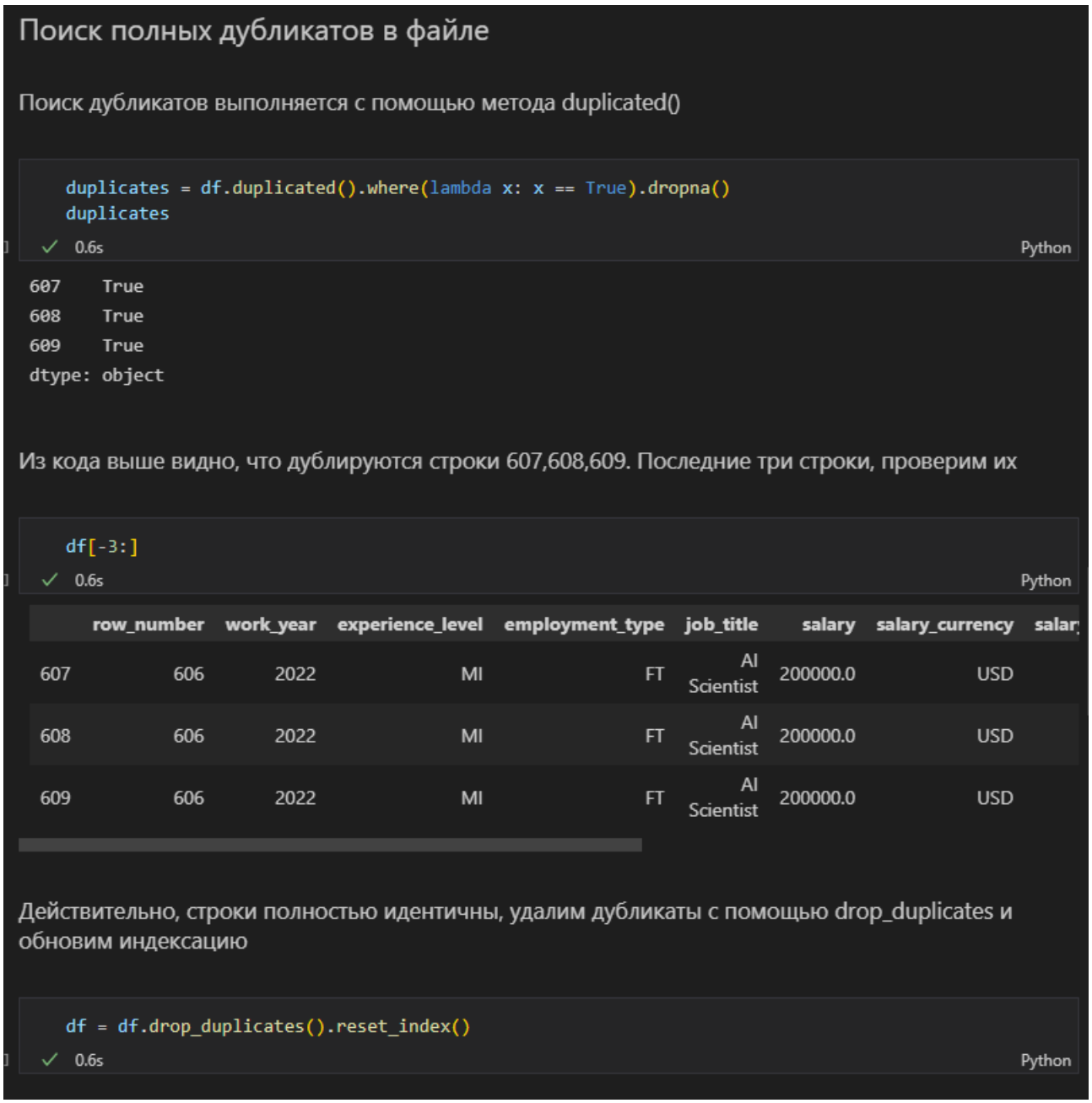


Рисунок 7 — Нашел и устранил явные дубликаты в данных

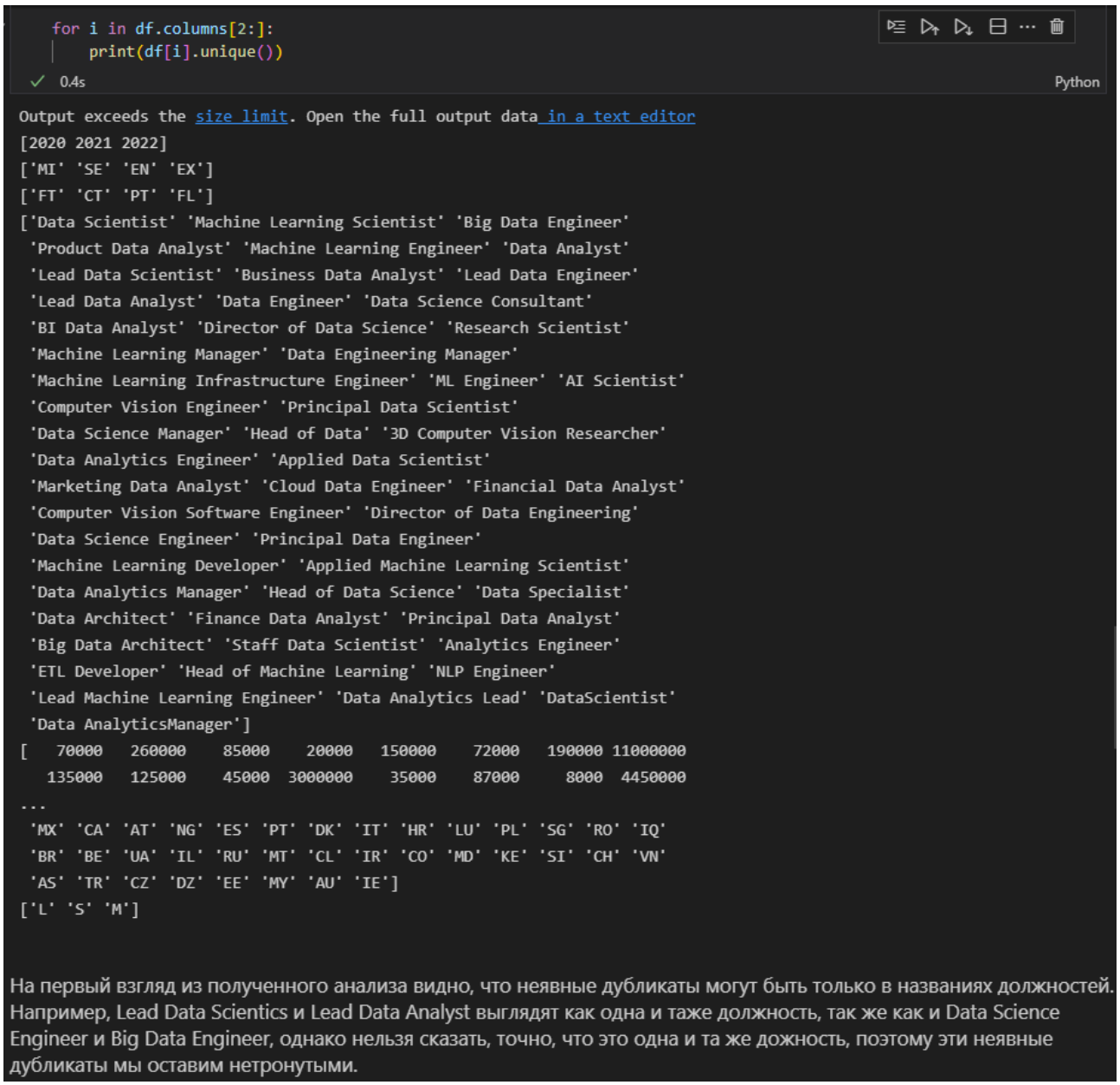


Рисунок 8 — Проверил данные на неявные дубликаты

Проверка типов данных на соотсвествие значениям

В ходе просмотра данных, было выявлено, что столбец salary вместо типа integer является типом float64, что нам не подходит. Изменим тип его данных Также стобец salary_in_usd не соотвествует типу int64, а также нужно убрать одно значение с опечаткой из этого столбца

```
df['salary_in_usd'] = df['salary_in_usd'].replace('d210000','210000')
df =df.astype({'salary':'int64','salary_in_usd':'int64'})
df.dtypes
```

✓ 0.4s Python

```
index                int64
row_number           int64
work_year            int64
experience_level    object
employment_type     object
job_title           object
salary               int64
salary_currency     object
salary_in_usd        int64
employee_residence  object
remote_ratio         int64
company_location    object
company_size        object
dtype: object
```

Рисунок 9 — Проверил типы данных на соответсвие

Сводные таблицы

На основании данных из файла составим сводные таблицы следующего типа: Средняя зп для специалистов разных уровней

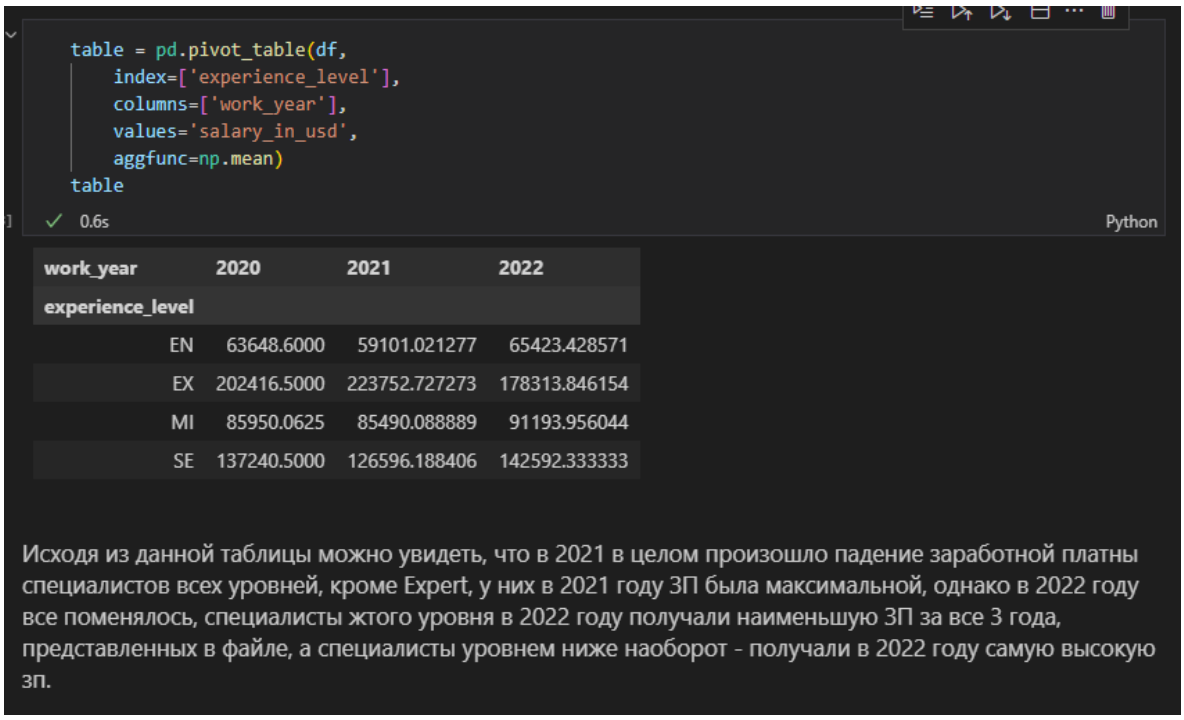
```
table = pd.pivot_table(df,index=["experience_level"],values=['salary_in_usd'],aggfunc=np.mean)
table.sort_values(by='salary_in_usd')
```

✓ 0.6s Python

experience_level	salary_in_usd
EN	61643.318182
MI	87996.056338
SE	138244.264493
EX	199392.038462

Из этой таблицы видна вполне очевидная закономерность: чем больше опыт работы - тем больше средняя зп

Рисунок 10 — Построил первую сводную таблицу



work_year	2020	2021	2022
experience_level			
EN	63648.6000	59101.021277	65423.428571
EX	202416.5000	223752.727273	178313.846154
MI	85950.0625	85490.088889	91193.956044
SE	137240.5000	126596.188406	142592.333333

Исходя из данной таблицы можно увидеть, что в 2021 в целом произошло падение заработной платны специалистов всех уровней, кроме Expert, у них в 2021 году ЗП была максимальной, однако в 2022 году все поменялось, специалисты жтого уровня в 2022 году получали наименьшую ЗП за все 3 года, представленных в файле, а специалисты уровнем ниже наоборот - получали в 2022 году самую высокую зп.

Рисунок 11 — Построил вторую сводную таблицу

```
table = pd.pivot_table(df,index=['employee_residence'],values='remote_ratio',aggfunc=[np.mean,len])
table
```

✓ 0.8s Python

	mean	len
	remote_ratio	remote_ratio
employee_residence		
AE	66.666667	3
AR	100.000000	1
AT	16.666667	3
AU	83.333333	3
BE	75.000000	2
BG	100.000000	1
BO	100.000000	1
BR	66.666667	6
CA	75.862069	29
CH	0.000000	1
CL	100.000000	1
CN	0.000000	1
CO	50.000000	1
CZ	50.000000	1
DE	56.000000	25
DK	50.000000	2
DZ	50.000000	1
EE	100.000000	1
ES	90.000000	15
FR	55.555556	18
GB	45.454545	44
GR	80.769231	13
HK	50.000000	1
HN	0.000000	1
HR	100.000000	1
HU	50.000000	2

Рисунок 12 — Первая часть третьей сводной таблицы

IE	100.000000	1
IN	63.333333	30
IQ	50.000000	1
IR	100.000000	1
IT	50.000000	4
JE	0.000000	1
JP	50.000000	7
KE	100.000000	1
LU	100.000000	1
MD	0.000000	1
MT	50.000000	1
MX	0.000000	2
MY	100.000000	1
NG	100.000000	2
NL	90.000000	5
NZ	50.000000	1
PH	100.000000	1
PK	66.666667	6
PL	100.000000	4
PR	50.000000	1
PT	66.666667	6
RO	25.000000	2
RS	100.000000	1
RU	62.500000	4
SG	75.000000	2
SI	75.000000	2
TN	100.000000	1
TR	50.000000	3
UA	100.000000	1
US	76.981707	328
VN	66.666667	3

Рисунок 13 — Вторая часть третьей сводной таблицы

Так же был произведен анализ третьей таблицы, в ходе которого выяснилось количество работников в каждой стране, и то, как они предпочитают работать (удаленно, полуудаленно, только в офисе).

Исходный файл Jupyter находится на сервере github по ссылке https://github.com/EgorYasinovskiy/Data-Analys/blob/master/JP1/main.ipynb

Вывод: В ходе выполнения лабораторной работы освоил методы предварительно анализа данных с помощью библиотеки pandas. Научился очищать исходные данные от невалидных строк и дубликатов, анализировать столбцы и типы исходных данных на соотвествие действительности. Также научился выполнять простые группировки данных с помощью сводных таблиц.