

ПРОГРАММНЫЙ МОДУЛЬ ДЛЯ АНАЛИЗА ФИНАНСОВОЙ СТАБИЛЬНОСТИ КОМПАНИЙ

Карцева В.И., студент,

Лобанов А.В., ассистент,

ФГБОУ ВО "Российский химико-технологический университет имени Д.И.
Менделеева" г. Москва, Россия

Аннотация. В условиях постоянно растущей конкуренции компаниям необходимо отслеживать динамику своих финансовых показателей. Эту задачу может упростить программа с возможностью анализа и визуализации данных по продажам. В рамках данной статьи описывается создание программного модуля на языке программирования Python, который сравнивает прибыль по месяцам, определяя убыточные периоды, с графической визуализацией, а также приводится пример его практического применения.

Ключевые слова: СУБД, БД, прибыль, программный модуль, Python, Excel.

Для работы с данными крупные компании все чаще используют системы управления базами данных (СУБД). Наиболее популярные корпоративные СУБД по статистическим данным — это Oracle и MS SQL Server [1].

Однако, стоит заметить, что в компаниях с небольшим штатом сотрудников с большой вероятностью нет человека, умеющего работать с какими-либо СУБД, и все данные хранятся, как правило, в MS Excel. Для таких компаний актуально создание программного модуля, который поможет

контролировать стабильность продаж, отслеживая месяцы, когда прибыль будет меньше средней за весь период.

Для реализации модуля, блок-схема которого представлена на рис. 1, выбран язык программирования Python, содержащий множество библиотек, в том числе подходящие для выполнения поставленной задачи [2].

Для написания программного кода были использованы следующие библиотеки:

- **Pandas** для обработки данных;
- **Matplotlib** для визуализации данных;
- **NumPy** для работы с массивами;
- **Tabulate** для форматирования таблиц.



Рис. 1. Блок-схема программы

Созданная программа состоит из четырех основных этапов:

- **initialData:** Первым этапом является чтение excel-таблицы с начальными данными (рис. 2) при помощи `df = pd.read_excel(file_path)` с результатом в виде объекта `DataFrame` (основная структура данных в `Pandas`). Далее столбец «Дата продажи» необходимо преобразовать к типу

`datetime` для корректной работы с датами, а также привести к удобному формату, например, "год-месяц" ("2022-01"), так как в нашем случае для определения суммарной прибыли за месяц день продажи не имеет значения;

- **monthlyProfit:** На втором этапе производится расчет суммарной прибыли за каждый месяц при помощи `df.groupby('Месяц')['Итоговая стоимость'].sum()`;
- **plotDiagram:** Третий этап. Из ежемесячных прибылей, рассчитанных в `monthlyProfit`, при помощи метода `mean()` определяется среднее арифметическое прибыли за весь период (`monthly_profit`). Найденная величина отмечается на графике пунктирной линией: `linestyle='--'`. Цвет столбца (красный или зеленый) устанавливается в зависимости от величины месячной прибыли по сравнению со средней: `colors = np.where(monthly_profit < avg_profit, 'red', 'green')`. Подписи месяцев, в которых не было совершено продаж, выделяются красным цветом. На график добавляется название, подписи осей, а также легенда. Полученный график представлен на рис. 3.
- **getTable:** Четвертый этап. С величиной `monthly_profit` (средняя прибыль за весь период) сравниваются величины прибыли за каждый месяц, чтобы определить «убыточные месяцы» (месяцы, в которых прибыль меньше средней). При помощи метода `isin()` выполняется фильтрация, и выводятся строки таблицы только по «убыточным» месяцам. Если за какие-то из месяцев сумма продаж равна 0, добавляем этот месяц в итоговую таблицу с прочерками вместо данных (рис. 4).

Сведения за месяцы с нулевыми продажами не добавляются в диаграмму и таблицу автоматически, так как этих данных не существует в исходной таблице. Важно самостоятельно прописать это в коде программы, так как это самые убыточные месяцы.

	A	B	C	D	E
1	Продукт	Дата продажи	Количество экземпляров	Цена за штуку	Итоговая стоимость
2	Продукт 1	12.01.2022	3	1000	3000
3	Продукт 1	14.01.2022	4	1000	4000
4	Продукт 3	15.01.2022	6	500	3000
5	Продукт 4	01.02.2022	1	6100	6100
6	Продукт 5	20.02.2022	10	235	2350
7	Продукт 2	20.02.2022	3	5400	16200
8	Продукт 2	03.03.2022	2	5400	10800
9	Продукт 4	04.03.2022	1	6100	6100
10	Продукт 3	05.03.2022	4	500	2000
11	Продукт 1	01.05.2022	4	1000	4000
12	Продукт 2	15.05.2022	1	5400	5400
13	Продукт 2	30.05.2022	2	5400	10800
14	Продукт 6	18.06.2022	7	2530	17710
15	Продукт 5	04.07.2022	14	235	3290
16	Продукт 1	08.07.2022	8	1000	8000
17	Продукт 3	09.08.2022	4	500	2000
18	Продукт 4	04.09.2022	6	6100	36600
19	Продукт 4	01.10.2022	2	6100	12200
20	Продукт 4	03.10.2022	1	6100	6100
21	Продукт 5	07.11.2022	15	235	3525
22	Продукт 1	02.12.2022	3	1000	3000
23	Продукт 1	20.12.2022	4	1000	4000
24	Продукт 2	31.12.2022	2	5400	10800

Рис. 2. Пример таблицы с входными данными

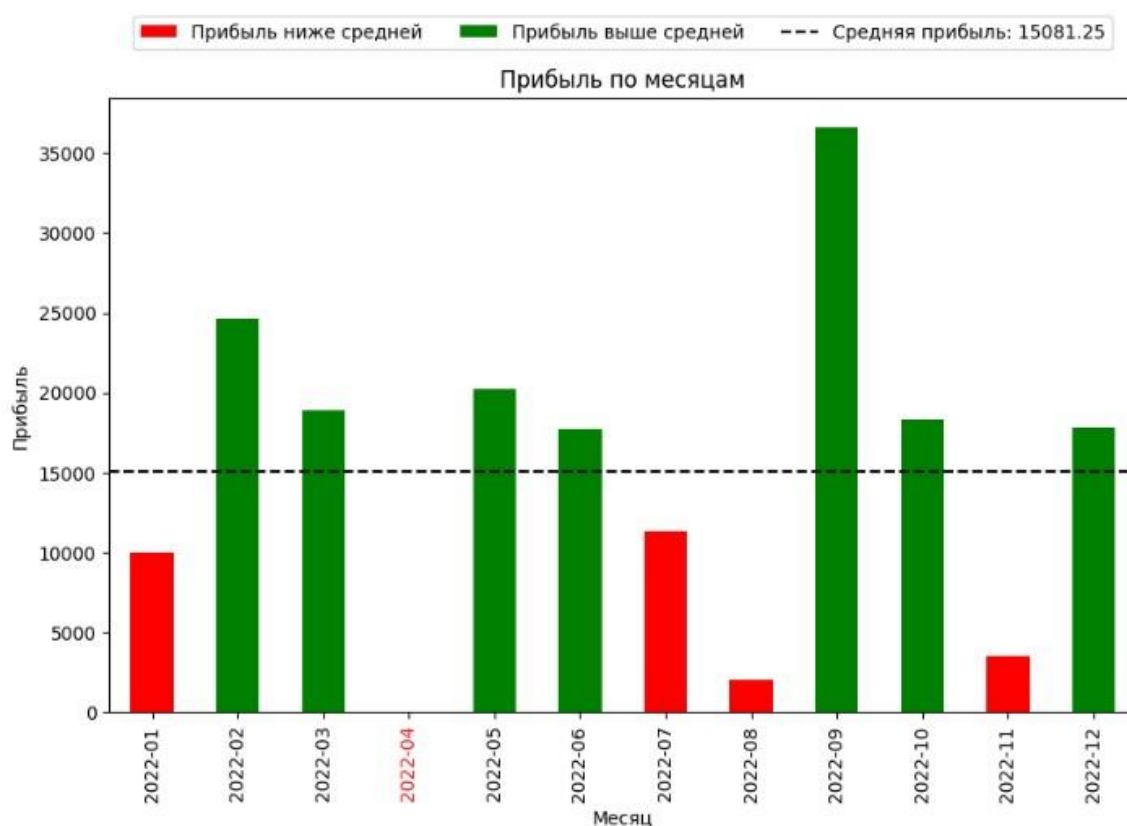


Рис. 3. Результат работы программы: диаграмма с ежемесячной прибылью

Продукт	Дата продажи	Количество экземпляров	Цена за штуку	Итоговая стоимость	Месяц
Продукт 1	2022-01-12	3	1000	3000	2022-01
Продукт 1	2022-01-14	4	1000	4000	2022-01
Продукт 3	2022-01-15	6	500	3000	2022-01
-	-	-	-	-	2022-04
Продукт 5	2022-07-04	14	235	3290	2022-07
Продукт 1	2022-07-08	8	1000	8000	2022-07
Продукт 3	2022-08-09	4	500	2000	2022-08
Продукт 5	2022-11-07	15	235	3525	2022-11

Рис. 4. Результат работы программы: подробная выписка за месяцы с прибылью ниже среднего

При наличии сведений о продажах за несколько лет программный модуль поможет определить месяцы, которые ежегодно являются убыточными по причине сезонности спроса [3].

Таким образом, в статье была описана разработка программного модуля, который успешно выполняет расчеты прибыли за каждый месяц и средней прибыли, а также выделяет убыточные месяцы. А визуализация данных в виде столбчатой диаграммы и подробная табличная выписка помогают выделить периоды для дальнейшего более глубокого анализа.

Литература

1. Атаев Р.Ш., Музалевская А.А. Выбор СУБД для создания системы автоматизации бизнес-процессов учреждения высшего образования // Формирование цифровой инфраструктуры предприятий: теоретические и прикладные аспекты. Сборник научных трудов национальной научно-практической конференции. 2020. С. 4-10.
2. Усачева А.Б., Галиаскаров Д.Р. Библиотеки Python для анализа данных // Colloquium-Journal. 2020. № 29-1 (81). С. 58-59.
3. Фурик М.М. Виды сезонного спроса и стимулирование продаж в несезонный период // Уральский федеральный университет имени первого Президента России Б. Н. Ельцина, Екатеринбург, 2022.