Pandas для анализа данных

Вызов библиотеки pandas

Вызов библиотеки pandas

```
In  import pandas
    import pandas as pd
```

Конструктор DataFrame() для создания таблицы

```
In  pd.DataFrame(data = data, columns =
    columns)
    # аргумент data – список с данными,
    # аргумент columns – список с
    # названиями столбцов
```

Метод tail() для вывода последних строк таблицы

```
In  df.tail() # последние 5 строк
    df.tail(15) # последние 15 строк
```

Метод read_csv() для чтения файлов формата CSV

```
In  df = pd.read_csv('путь к файлу')
```

Метод head() для вывода первых строк таблицы

```
In  df.head() # первые 5 строк
    df.head(10) # первые 10 строк
```

Атрибут columns для вывода названий столбцов

```
In  df.columns
```

Атрибут shape для вывода размера таблицы

```
In  df.shape
```

Атрибут dtypes для получения информации о типах данных в таблице

```
In  df.dtypes
```

Метод info() для просмотра сводной информации о таблице

```
In  df.info()
```

Атрибут loc[строка, столбец] даёт доступ к элементу в DataFrame по строке и столбцу

```
In  df.loc[:, 'column']
```

```
Out
```

Вид	Реализация
Одна ячейка	`.loc[7, 'column']`
Один столбец	`.loc[:, 'column']`
Несколько столбцов	`.loc[:, ['column_1', 'column_4']]`
Несколько столбцов подряд (срез)	`.loc[:, 'column_5': 'column_8']`
Одна строка	`.loc[1]`
Все строки, начиная с заданной	`.loc[1:]`
Все строки до заданной	`.loc[:3]`
Несколько строк подряд (срез)	`.loc[2:5]`

Логическая индексация для получения элементов по определенному условию

```
Out
```

Вид	Реализация	Сокращенная запись
Все строки, удовлетворяющие условию	`'df.loc[df.loc[:,'column'] == 'X']'`	`'df[df['column'] == 'X']'`
Столбец, удовлетворяющий условию	`'df.loc[df.loc[:,'column'] == 'X']['column']'`	`'df[df['column'] == 'X']['column']'`
Применение метода	`'df.loc[df.loc[:,'column'] == 'X']['column'].count()'`	`'df[df['column'] == 'X']['column'].count()'`

Индексация в Series

```
Out
```

Вид	Реализация	Сокращенная запись
Один элемент	`` `df.loc[7]` ``	`` `df[7]` ``
Несколько элементов	`` `df.loc[[5, 7, 10]]` ``	`` `df[[5, 7, 10]]` ``
Несколько элементов подряд (срез)	`` `df.loc[5:10]` `` включая 10	`` `df[5:10]` `` не включая 10
Все элементы, начиная с заданного	`` `df.loc[1:]` ``	`` `df[1:]` ``
Все элементы до заданного	`` `df.loc[:3]` `` включая 3	`` `df[:3]` `` не включая 3

Словарь

Библиотека
Это набор готовых методов для решения распространенных задач

CSV
Формат файла (от англ. Comma-Separated Values, «значения, разделённые запятой»). Каждая строка представляет собой одну строку таблицы, где данные разделены запятыми. В первой строке собраны заголовки столбцов (если они есть)

Кортеж
Одномерная неизменяемая последовательность данных. Она похожа на список, её тоже можно сохранять в переменной.

Series
Одномерная структура данных Pandas, её элементы можно получить по индексу. Каждый индекс представляет собой номер отдельного наблюдения, и поэтому несколько различных Series вместе составляют DataFrame.

- В Series хранятся данные одного типа.
- У Series есть имя (Name), информация о количестве данных в столбце (Length) и тип данных, которые хранятся в ней (dtype).
- Индексация в Series аналогична индексации элементов столбца в DataFrame.

DataFrame
Это двумерная структура данных Pandas, где
у каждого элемента есть два индекса:
по строке и по столбцу.

- Каждая строка — это одно наблюдение, запись об объекте исследования. А столбцы — признаки этого объекта.
- DataFrame() – это конструктор библиотеки Pandas, который используется для создания **DataFrame**. Перед именем конструктора стоит обращение к переменной, в которой библиотека хранится: pd.DataFrame().
- У DataFrame есть неотъемлемые свойства, значения которых можно запросить. Они называются атрибуты. Например, это размер таблицы df.shape или количество столбцов df.columns.
- К каждой ячейке с данными в DataFrame можно обратиться по её индексу и названию столбца. Этот процесс называется индексация и для DataFrame его проводят разными способами.