Часть 1: Введение в Pandas

Pandas - это библиотека Python для работы с данными. Она предоставляет мощные инструменты для анализа и манипуляции данными.

Установка Pandas

Для начала убедитесь, что у вас установлена библиотека Pandas. Если ее нет, установите ее с помощью команды:

!pip install pandas

Импорт библиотеки

Давайте начнем с импорта библиотеки Pandas:

Ответ:

import pandas as pd

print("Библиотека Pandas была успешно импортирована")

Часть 2: Работа с данными

Чтение данных

Мы будем использовать датасет Titanic. Давайте прочитаем данные из CSV файла.

Ответ:

import pandas as pd

data = pd.read\_csv('titanic.csv')

print(data.head())

Предпросмотр данных

Для первого ознакомления с данными давайте выведем первые несколько строк.

Ответ:

import pandas as pd

data = pd.read\_csv('titanic.csv')

print(data.head())

Информация о данных

Чтобы получить общую информацию о данных, воспользуйтесь методом .info().

Ответ:

print(data.info())

Работа с NaN

Часто данные содержат пропущенные значения, которые представляются как NaN (Not a Number). Pandas предоставляет удобные методы для работы с ними.

Проверка на наличие NaN. Методы fillna и dropna возвращают новые Dataframe, проверьте нет ли в них NaN

Ответ:

print(data.isna())

Заполнение NaN

Ответ:

filled\_data = data.fillna(0)

Удаление строк с NaN

Ответ:

cleaned\_data = data.dropna()

Часть 3: Обработка DataFrame

Выбор данных

Pandas позволяет выбирать данные по индексам, меткам столбцов и условиям.

Ответ:

# Выбор столбца по метке

selected\_column = data['column\_label']

# Выбор нескольких столбцов

selected\_columns = data[['column\_1', 'column\_2']]

# Выбор строк по индексу

selected\_row = data.loc[index\_label]

# Выбор строк и столбцов по условию

conditioned\_selection = data[(data['gender'] == 'male') & (data['age'] > 30)]

Сортировка данных

Сортировка данных по значениям столбцов.

Ответ:

sorted\_data = data.sort\_values(by='столбец\_1')

Группировка данных

Pandas также позволяет группировать данные и выполнять агрегирующие операции.

Ответ:

survival\_rate = data.groupby('PClass')['Survived'].mean()

Часть 4: Задания для практики

Прочитайте данные из файла 'titanic.csv'. Проверьте, есть ли пропущенные значения в данных и заполните их нулями. Выведите первые 10 строк данных. Выберите только те строки, где значение в столбце 'Age' больше 30. Отсортируйте данные по столбцу 'Fare' в порядке убывания. Сгруппируйте данные по столбцу 'Pclass' и вычислите средний возраст ('Age') для каждого класса.

Ответ:

import pandas as pd

data = pd.read\_csv('titanic.csv')

data.fillna(0, inplace=True)

print(data.head(10))

age\_over\_30 = data[data['Age'] > 30]

sorted\_data = data.sort\_values(by='Fare', ascending=False)

average\_age\_by\_class = data.groupby('Pclass')['Age'].mean()