

Python, pandas va Excel dunyosidan uchta muammoni hal qilish.

Excel ma'lumotlarni tahlil qilish uchun juda keng tarqalgan vositadir. U bilan ishlashni o'rganish oson, u deyarli har bir kompyuterda va uni o'zlashtirgan kishi juda murakkab muammolarni hal qilish uchun foydalanishi mumkin. Python ko'pincha imkoniyatlari deyarli cheksiz bo'lgan, ammo Excelga qaraganda qiyinroq bo'lgan vosita hisoblanadi. Bugungi kunda nashr etadigan materialning muallifi, odatda, Excelda hal qilinadigan uchta vazifani Python bilan hal qilishni istaydi. Ushbu maqola Excelni yaxshi biladiganlar uchun Pythonga kirish kaliti.



Ma'lumotlarni yuklab olish

Keling, Python kutubxonasini import

```
pandas
```

qilish va

```
sales
```

Excel sahifalarida va kitoblarida saqlangan ma'lumotlar bazasiga yuklashdan boshlaylik

```
states
```

. Xuddi shu nomlarni tegishli dataframelarga beramiz.

```
import pandas as pd
sales =
pd.read_excel('https://github.com/datagy/mediumdata/raw/master/pythonexcel
.xlsx', sheet_name = 'sales')
states =
pd.read_excel('https://github.com/datagy/mediumdata/raw/master/pythonexcel
.xlsx', sheet_name = 'states')
```

Endi

```
.head() sales
```

PYTHON DASTURLASH TILINI BIZ BILAN O'RGANING !!!

@PYTHON_UZBEK_CODER

dataframning boshida joylashgan elementlarni olib tashlash uchun dataframe usulidan foydalanamiz:

```
print(sales.head())
```

Excel va Pythonni taqqoslang.

The image shows a comparison between Excel and Python with Pandas. On the left, an Excel spreadsheet is shown with columns A, B, C, and D. The data is as follows:

	A	B	C	D
1	Product	Sales	Date	City
2	Bananas	\$ 121.00	2019-06-13	Atlanta
3	Bananas	\$ 236.00	2019-10-20	Atlanta
4	Apples	\$ 981.00	2019-03-12	Atlanta
5	Bread	\$ 996.00	2019-07-28	New York City
6	Broccoli	\$ 790.00	2019-10-22	New York City

On the right, a Python Pandas DataFrame is shown with columns Product, Sales, Date, and City. The data is as follows:

	Product	Sales	Date	City
0	Bananas	121	2019-06-13	Atlanta
1	Bananas	236	2019-10-20	Atlanta
2	Apples	981	2019-03-12	Atlanta
3	Bread	996	2019-07-28	New York City
4	Broccoli	790	2019-10-22	New York City

Below the tables, it says: "Pretty similar *" and "*but also kind of different".

Excelda olingan ma'lumotlarning ko'rinishini pandas ma'lumotnomasidan olingan ma'lumotlarning ko'rinishi bilan taqqoslash

Bu yerda dataframe ma'lumotlarini ko'rish natijalari Excelda ko'rish mumkin bo'lgan narsalarga juda o'xshashligini ko'rishingiz mumkin. Ammo juda muhim farqlar mavjud:

- Excelda qatorlarni raqamlash 1 bilan boshlanadi va pandasda birinchi satrning raqami (indeksi) 0ga teng.
- Excelda ustunlar harf bilan boshlangan harf belgilariga ega A va pandasda ustunlar nomlari mos keladigan o'zgaruvchilar nomlariga mos keladi.

Pandas imkoniyatlarini o'rganishni davom ettiramiz, bu odatda Excelda hal qilinadigan muammolarni hal qilishga imkon beradi.

Excel xususiyatlarini amalga oshirish-Pythonda IF funktsiyalari

Excelda

IF

, masalan, boshqa hujayradagi narsalarni tekshirish asosida, masalan, kameraga biror narsa yozib olish imkonini beruvchi juda qulay xususiyat mavjud. Excelda yangi ustunni yaratishingiz kerak, hujayralar bizga tegishli ustun hujayralarida qayd etilgan 500 qiymatdan oshib ketadimi-yo'qmi haqida xabar beradi B

. Excelda bunday ustun (bizning holimizda bu ustun E)

MoreThan500

mos keladigan matnni kameraga yozib , sarlavhani belgilashingiz mumkin

PYTHON DASTURLASH TILINI BIZ BILAN O'RGANING !!!

@PYTHON_UZBEK_CODER

E1

. Shundan

E2

so'ng , hujayra ichida quyidagilarni kiritishingiz mumkin:

```
=IF([@Sales]>500, "Yes", "No")
```

	A	B	C	D	E	F	G
1	Product	Sales	Date	City	MoreThan500		
2	Bananas	\$ 121.00	2019-06-13	Atlanta			
3	Bananas	\$ 236.00	2019-10-20	Atlanta			
4	Apples	\$ 981.00	2019-03-12	Atlanta			
5	Bread	\$ 996.00	2019-07-28	New York City			
6	Broccoli	\$ 790.00	2019-10-22	New York City			
7	Apples	\$ 762.00	2019-11-03	Toronto			
8	Bananas	\$ 870.00	2019-11-18	New York City			

Excelda IF funksiyasidan foydalanish

Pandas yordamida ham xuddi shunday qilish uchun siz ro'yxatdan o'tishni (list comprehension) ishlatishingiz mumkin):

```
sales['MoreThan500'] = ['Yes' if x > 500 else 'No' for x in sales['Sales']]
```

List Comprehensions in Python

[expression for item in list]

['Yes' if x > 500 else 'No' for x in sales['Sales']]

Python-da ro'yxatli qo'shimchalar: agar joriy qiymat 500 dan ortiq bo'lsa-ro'yxatda Ha, aks holda-yo'q

Ro'yxatlardagi inkluziyonlar bunday muammolarni hal qilishning ajoyib vositasidir, bu esa if/else turidagi murakkab tuzilmalarga bo'lgan ehtiyojni kamaytirish orqali kodni soddalashtirishga imkon beradi. Xuddi shu muammoni if/else bilan hal qilish mumkin, ammo tavsiya etilgan yondashuv vaqtini tejaydi va kodni biroz tozalaydi. Ro'yxatga kiritilgan qo'shimchalar haqida batafsil

<https://datagy.io/list-comprehensions-in-python/> dan topishingiz mumkin.

Excel xususiyatlarini amalga oshirish-Pythonda VLOOKUP vazifalari

Bizning ma'lumotlar to'plamimizda, Excel sahifalaridan birida shaharlarning nomlari, ikkinchisida esa davlatlar va viloyatlarning nomlari mavjud. Har bir shahar qayerda joylashgani haqida qanday ma'lumot olish mumkin? Buning uchun Excel mos keladi-

VLOOKUP

siz ikkita jadvalning ma'lumotlarini ulashingiz mumkin bo'lgan funksiya. Bu funksiya ifoda chap tomonida joylashgan ma'lumotlar majmui har bir rekord saqlanadi qachon chap munosabati tamoyili bo'yicha ishlaydi. Funktsiyani qo'llash

VLOOKUP

. biz belgilangan varaqning ma'lum bir ustunida ma'lum bir qiymatni qidirishni amalga oshirish uchun tizimni taklif qilamiz, so'ngra topilgan qiymatning o'ng tomonidagi ustunlar soniga berilgan qiymatni qaytaramiz. Bu qanday ko'rinishga ega:

```
=VLOOKUP([@City],states,2,false)
```

sales

Biz ushbu ustunning

FStateVLOOKUP

hujayralarini shaharlar joylashgan davlatlar va viloyatlar nomlari bilan to'ldirish uchun funksiyadan foydalanamiz.

	A	B	C	D	E	F	G	H	I
1	Product	Sales	Date	City	MoreThan500	State			
2	Bananas	\$ 121.00	2019-06-13	Atlanta	No				
3	Bananas	\$ 236.00	2019-10-20	Atlanta	No				
4	Apples	\$ 981.00	2019-03-12	Atlanta	Yes				
5	Bread	\$ 996.00	2019-07-28	New York City	Yes				
6	Broccoli	\$ 790.00	2019-10-22	New York City	Yes				
7	Apples	\$ 762.00	2019-11-03	Toronto	Yes				
8	Bananas	\$ 870.00	2019-11-18	New York City	Yes				
9	Bananas	\$ 852.00	2019-03-21	Atlanta	Yes				
10	Apples	\$ 427.00	2019-05-11	Toronto	No				
11	Bread	\$ 576.00	2019-09-14	Atlanta	Yes				

Excelda VLOOKUP funksiyasidan foydalanish

Pythonda pandas usuli yordamida ham xuddi shunday qilishingiz mumkin

merge

. Bu ikki dataframe oladi va ularni birlashtiradi. Ushbu muammoni hal qilish uchun biz quyidagi kodni talab qilamiz:

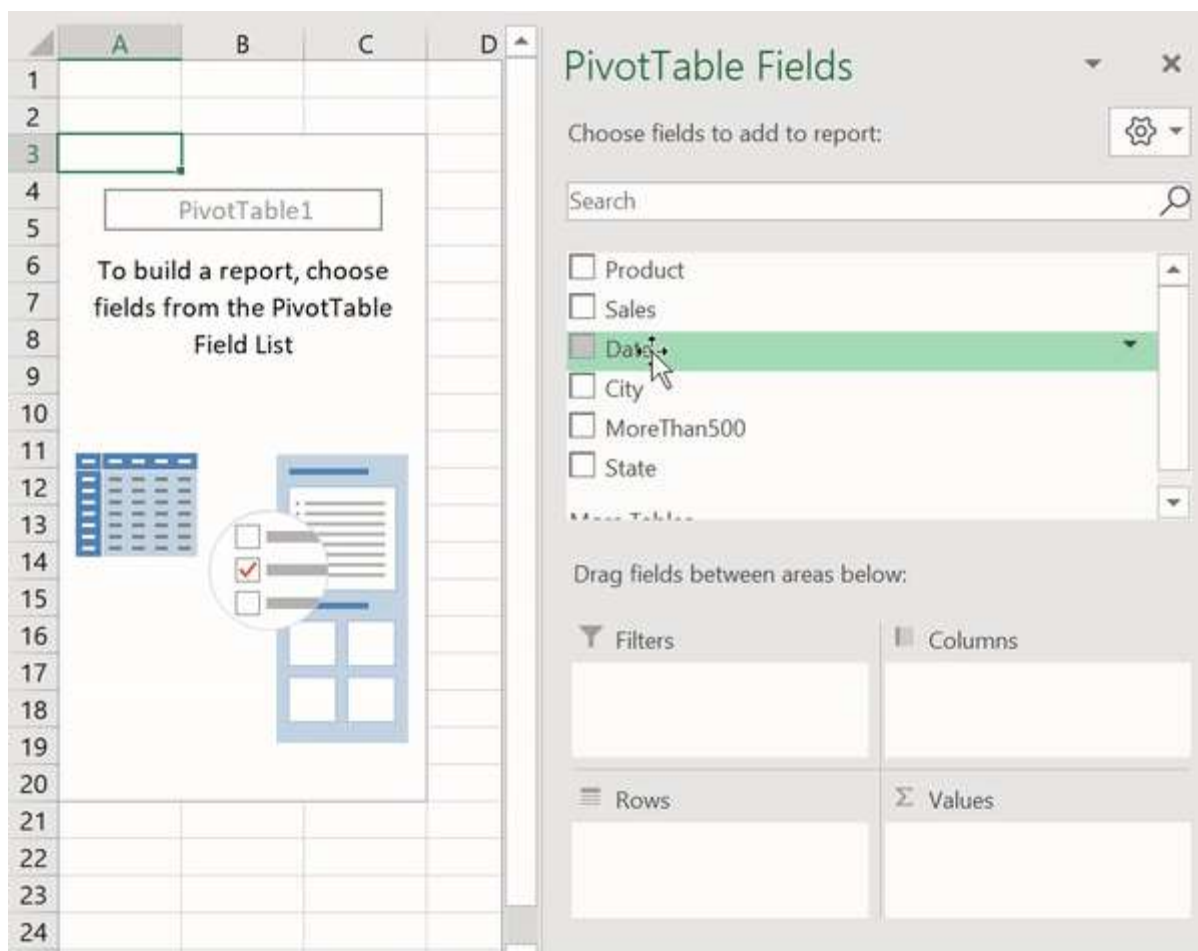
```
sales = pd.merge(sales, states, how='left', on='City')
```

Keling, uni tahlil qilaylik:

1. Usulning birinchi argumenti `mergeas` dataframe hisoblanadi.
2. Ikkinchi dalil-biz qadriyatlarni qidiradigan dataframe.
3. Argument `how` ma'lumotlarni qanday qilib ulashni xohlayotganimizni ko'rsatadi.
4. Argument `onsiz` ulanishni amalga oshirish kerak bo'lgan o'zgaruvchiga ishora qiladi (bu erda siz argumentlarni ishlatishingiz mumkin `left_on` va `right_on` biz qiziqayotgan ma'lumotlar turli xil ma'lumotlar doiralarda boshqacha nomlangan bo'lsa).

Xulosa jadvallari

Pivot jadvallari (Pivot jadvallari) Excelning eng kuchli xususiyatlaridan biridir. Bunday jadvallar katta ma'lumotlar to'plamlaridan qimmatli ma'lumotlarni tezda olish imkonini beradi. Excelda har bir shahar uchun umumiy sotuvlar haqida ma'lumot beruvchi Pivot jadvalini yaratamiz.



Excelda Pivot jadvalini yaratish

Ko'rib turganingizdek , bunday jadvalni yaratish uchun maydonni

City

bo'limga

Rows

va maydonni bo'limga sudrab borish kifoya

SalesValues

. Shundan so'ng, Excel avtomatik ravishda har bir shahar uchun umumiy savdolarni ko'rsatadi.

Pandasda bir xil Pivot jadvalini yaratish uchun quyidagi kodni yozishingiz kerak bo'ladi:

```
sales.pivot_table(index = 'City', values = 'Sales', aggfunc = 'sum')
```

Keling, uni tahlil qilaylik:

1. Bu erda biz `sales.pivot_table` pandasga dataframe-ga asoslangan Pivot jadvalini yaratmoqchi ekanligimizni aytib berish usulidan foydalanamiz `sales`.
2. Argument `index` biz ma'lumotlarni to'plash uchun kerakli ustun ko'rsatadi.
3. Argument `values` biz to'plash uchun boryapmiz nima qadriyatlar ko'rsatadi.
4. Argument `aggfunc` vazifasini belgilaydi, biz qadriyatlarni qayta ishlash foydalanish istayman (bu erda siz vazifalari foydalanishingiz mumkin `mean`, `max` va `min` hokazo).

Natijalar

Ushbu materialdan siz Excel ma'lumotlarini pandasga qanday import qilish, Python va pandas vositalarini Excel funktsiyalari va imkoniyatlarini qanday amalga oshirish haqida bilib oldingiz

IF

VLOOKUP

, shuningdek, pandas bilan Excel Pivot jadvallarining funktsional xususiyatlarini qanday ijro etish haqida. Ehtimol, endi siz Excelda ham xuddi shunday qilishingiz mumkin bo'lsa, pandasni nima uchun ishlatishingiz haqida hayron bo'lasiz. Bu savolga aniq javob yo'q. Python nozik sozlash va chuqur o'rganish uchun javob beradigan kodni yaratishga imkon beradi. Ushbu kod bir necha marta ishlatilishi mumkin. Python vositalari juda murakkab ma'lumotlarni tahlil qilish sxemalarini tasvirlab berishi mumkin. Va Excelning imkoniyatlari, ehtimol, ma'lumotlarning kamroq keng ko'lamli tadqiqotlari uchun etarli. Agar siz shu paytgacha faqat Excelni ishlatgan bo'lsangiz — Python va pandasni boshdan kechirishni tavsiya etaman va undan nima qilish kerakligini bilib oling.

Maqola muallifi: [JaxaDEV](#)