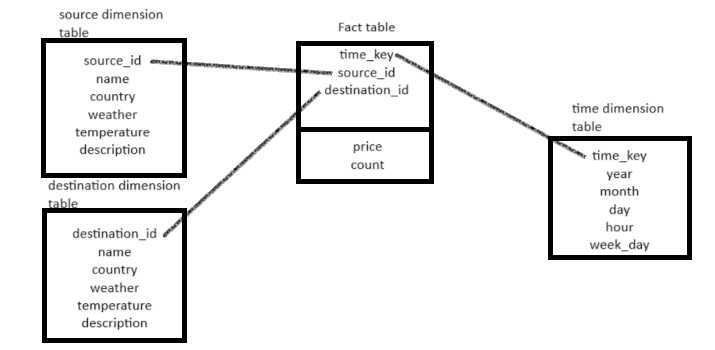
**تمرین دوم درس داده کاوی محمدباربد امیرمزلقانی – 810102348**

**بخش تشریحی**

**سوال اول)**

الف) \*منظور از name همان city است.



ب)ابتدا تهران به میلان در خرداد 1402 :

* roll-up بر روی مقصد تا لول name و دایس بر روی میلان
* Roll-up بر روی مبدا تا لول name و دایس بر روی تهران
* Slice بر روی تایم یکی برای خرداد 1402 و یکی هم برای فروردین 1401
* Drill-down تا ماه (در بعد تایم)
* Aggregate بر روی price

**سوال 2)**

الف) چون هدف مینیمایز کردن سایز اولیه است پس ابتدا B و C که کوچکترین ابعاد هستند، سپس B و A و در نهایت C و A.

ب)

BC شامل 100 \* 1000 سلول

BA شامل 100 \* 1000000 سلول

CA شامل 1000 \* 1000000 سلول

که مجموع آن ها را در 4 بایت ضرب میکنیم ( طبق گفته سوال سایز هر سلول 4 بایت است )

که اگر این جمع و ضرب را انجام بدهیم به 4,400,400,000 بایت میرسیم که حدود 4200 مگابایت است.

ج) اگریگیت بر روی بعدی که نمیخواهیم بهترین روش خواهد بود، منظور جمع میانگین یا ... است برای مثال از کیوبید BA اگریگیت میکنیم بر روی B ( مثلا جمع یا میانگین بر روی کل Bها به ازای هر A) که در همین روش هم بهترین کار، کوچک کردن تعداد عملیات ها تا حد ممکن است مثلا برای A میتوانیم از BA یا CA استفاده کنیم که B در اینجا با توجه به سایز کوچکتر ارجحیت دارد(همین راه حل را میتوان برای B و C هم اعمال کرد)

**سوال 3)**

الف)

در اینجا 2 به توان n خواهد بود که با توجه به 9 بودن میشود 9^2

ب)

برای هر سلول این تعداد برابر 2 به توان n منهای یک میشود (بدون خودش) از اونجایی که دو تا بیس داریم میشود ولی باید اورلپ ها را هم کم کنیم که میشود 3 سلول که حالت های مختلف آن میشود پس در کل میشود :

ج)

دو تا بیس و یکی هم این سلول بسته اگریگیت شده : (\_,*a*2​,\_,*a*4​,*a*5​,\_,*a*7​,*a*8​,\_)

د)

اگر سه بعد مشترک را در نظر بگیریم، که بتوانیم شرط را هم ارضا کنیم3^2 داریم که کلا 8 تا میشود.

**سوال چهارم)**

الف) برای محاسبه ترتیب بهینه لازم است کاردینالیتی هر بعد را حساب کنیم( که تعداد مقادیر منحصر به فرد هر بعد میشود)

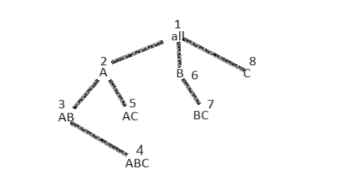
* Gender : 2
* Education : 3
* Occupation : 4

حال بر مبنای این مقادیر ترتیب اپتیمال به صورت 1.شغل 2.تحصیلات 3.جنسیت میشود.

دلایل هم برای این روش بدین صورت است که باعث بهبود پرون کردن میشود به این صورت که فضای جستجو زودتر پرون شود چون ترکیب آن ها مثلا با بعدهای دیگر شرط مینیمم ساپورت را رعایت نمیکند زودتر حذف میشوند .

ب)

با فرض اینکه A=occupation ، B=education و C=gender به این درخت میرسیم :



حال برای هر کدام :

A: استاد:3 ، برنامه نویس :4

AB: برنامه نویس و دانشجو : 3

B: دانشجو 5 ، دبیرستان : 3

BC: دانشجو و زن : 3

C: زن : 5 ، مرد : 5

**بخش عملی**

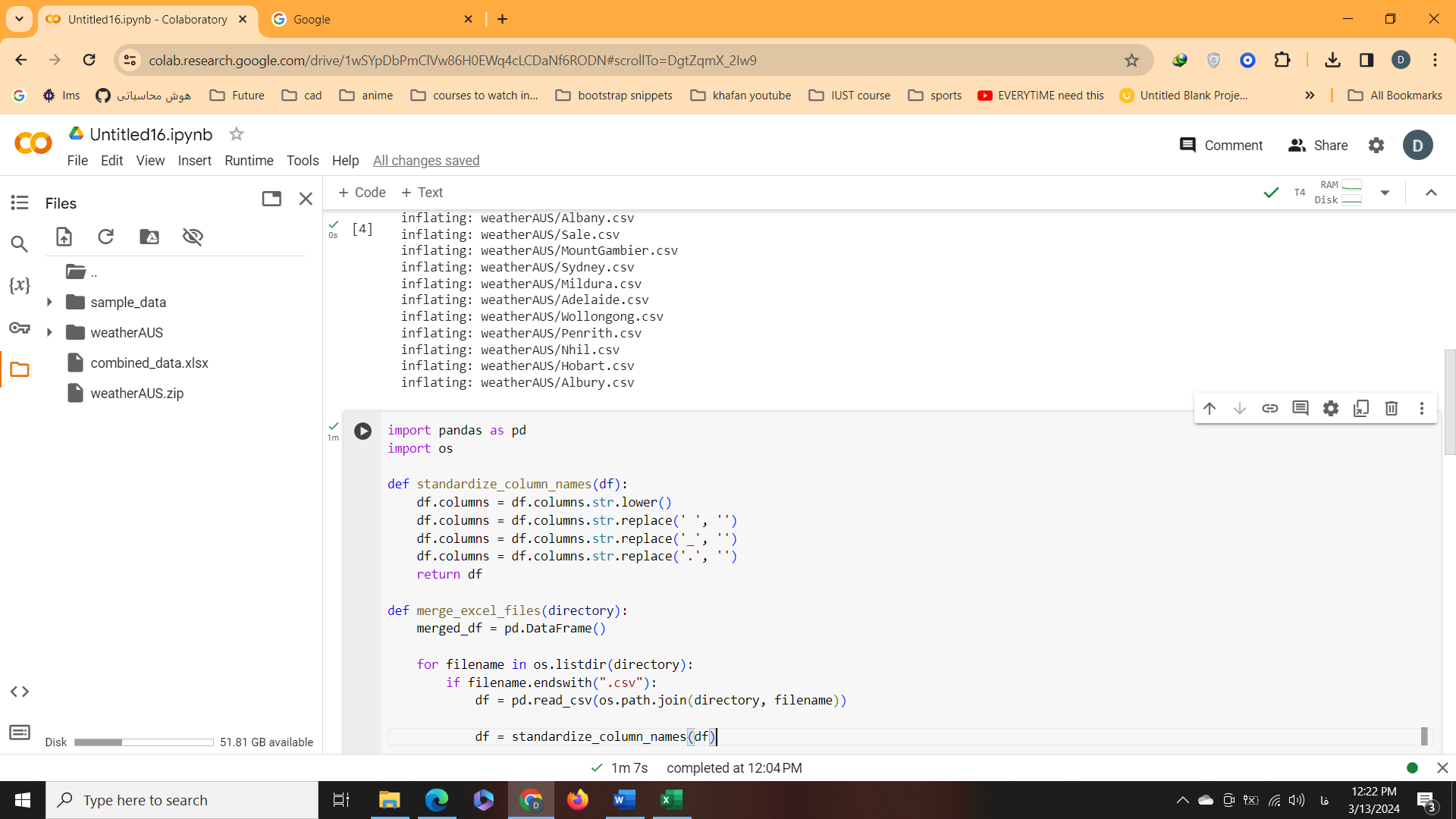
**(بر روی colab ران میشود و تنها لازم است فایل زیپ دیتاست در کنار فایل آپلود شود)**

**سوال یک)**

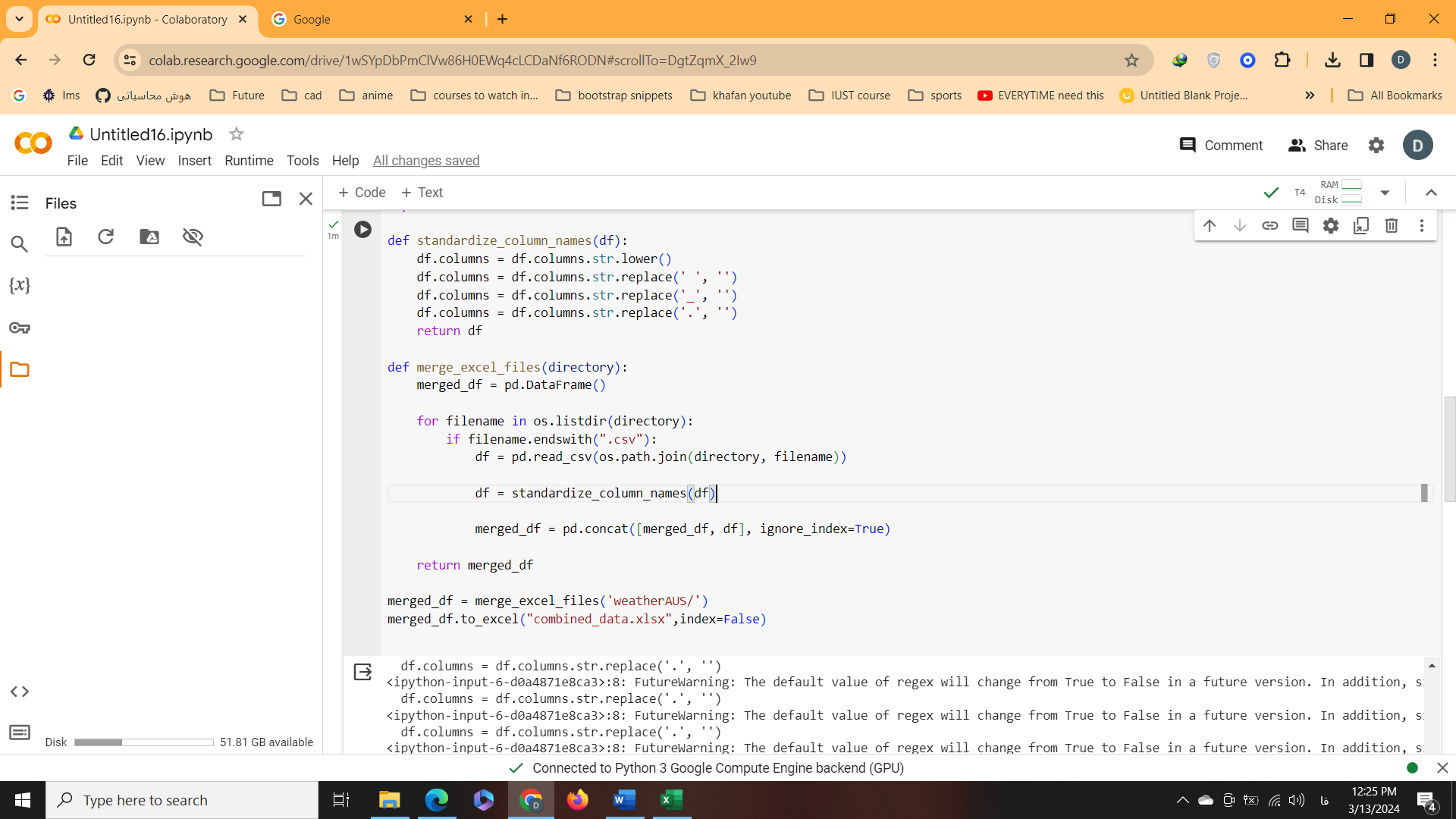
در این بخش ابتدا پیش پردازش را کامل طبق 12 مرحله تمرین اول انجام میدهیم، در ادامه داک و کد این بخش ها موجود است.

1.ابتدا در رابطه با مشکلات ادغام کردن فایل ها و سپس راه حل های آن میپردازیم، همانطور که در pdf هم اشاره شد اصلی ترین مشکل تفاوت های موجود در نام ستون‌ها بود، مشکل بعدی هم وجود یک سری پارامترها در یک سری فایل ها بود که در دیگر ایستگاه‌ها حضور نداشتند.

با نگاه دقیق به فایل‌ها دریافتیم که این تفاوت ها معمولا آنچنان قوی نیستند که نام پارامتر را به طور کلی تغییر دهند مثلا صرفا \_ اضافه شده است یا حروف اول کلمات بزرگ هستند یا بین بخش های مختلف . گذاشته است، که این موارد به راحتی به کمک پایتون قابل حل است(اگر تفاوت در نام ها هم موجود بود دو روش میتوانستیم استفاده کنیم 1.یک فانکشن استخراجگر بنویسیم که مثلا به دنبال تکه temp و 3 برود که هر ترکیبی از آن میتواند بیانگر دما در ساعت 3 باشد 2.یک فانکشن تعریف کنیم برای تشخیص نزدیکی کلمات بدین معنی که نزدیک ترین کلمه در فایل های مختلف یک ستون را تشکیل دهند و برای فاصله بین کلمات هم میتوانیم مثلا از cosine similarity استفاده کنیم.) ولی الان با استفاده از فانکشن های زیر که ابتدا همه را lower case کنیم سپس \_ و . را حذف کنیم مشکلمان حل میشود :

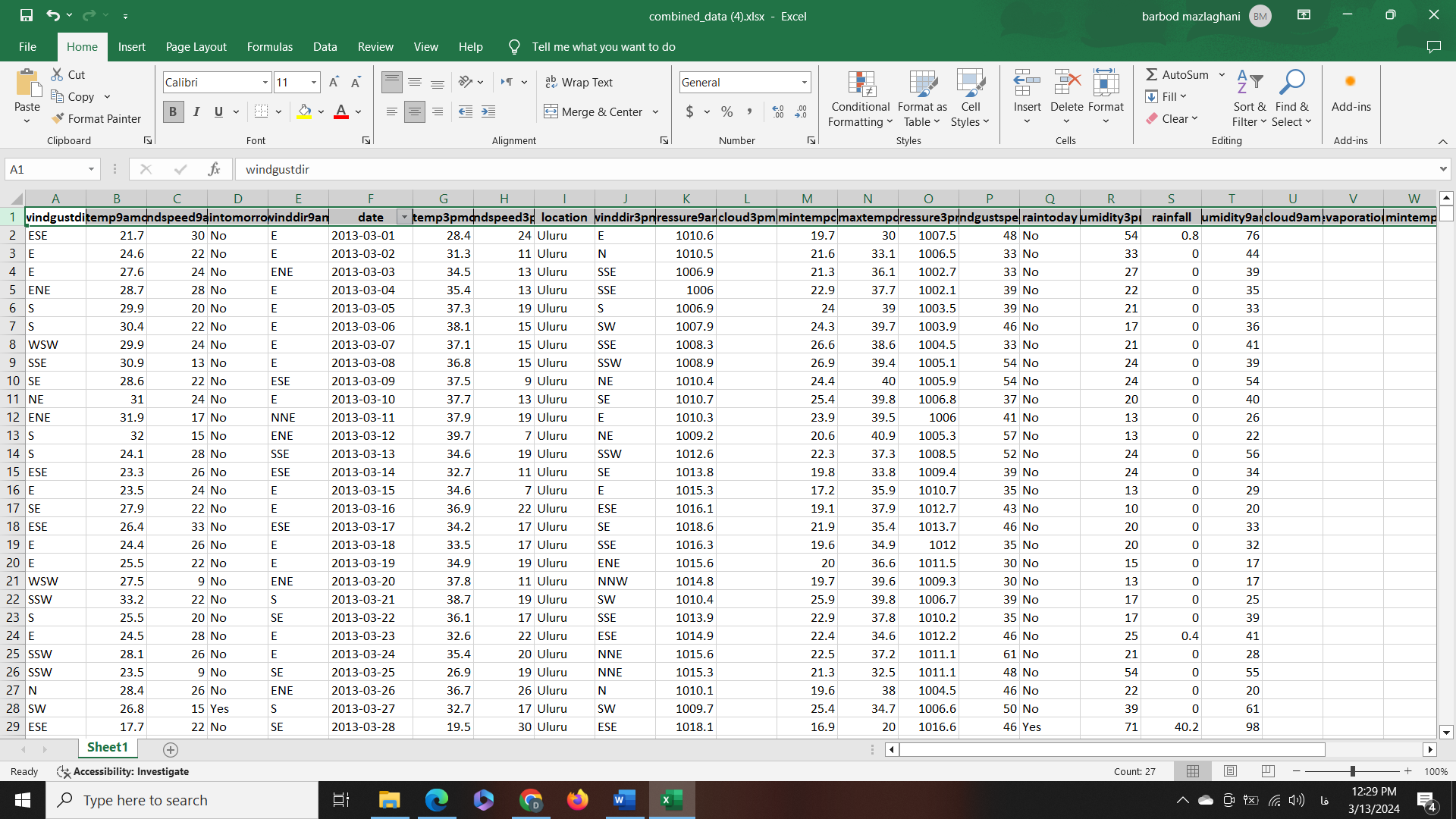


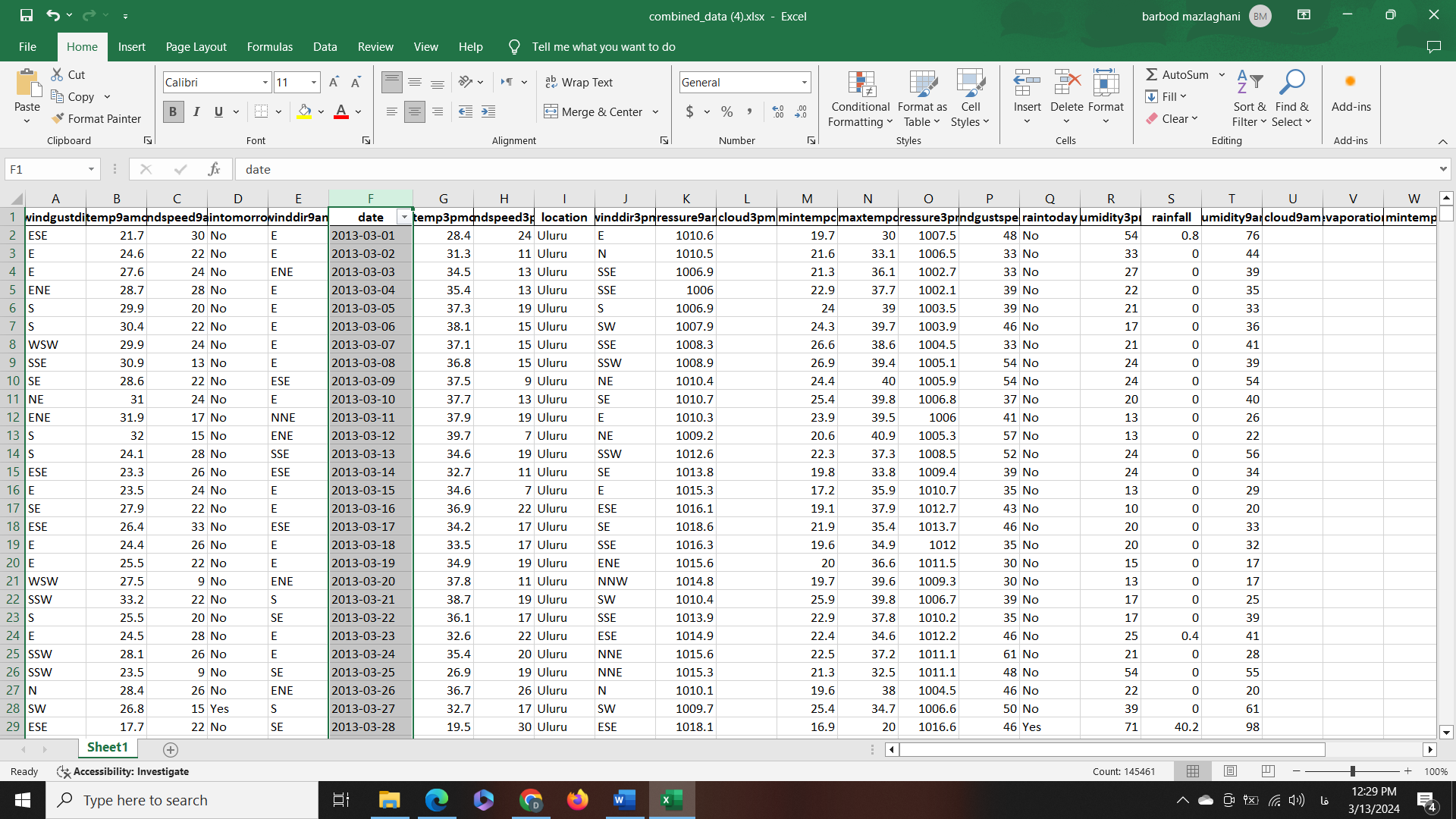
در گام بعدی برای هندل کردن ستون های متفاوت در ایستگاه های متفاوت تنها موقع مرج کردن دیتافریم ignore\_index را True میگذاریم که دیتافریم جدید را دوباره index گذاری میکند و ستون های اضافی هم هندل میشود.



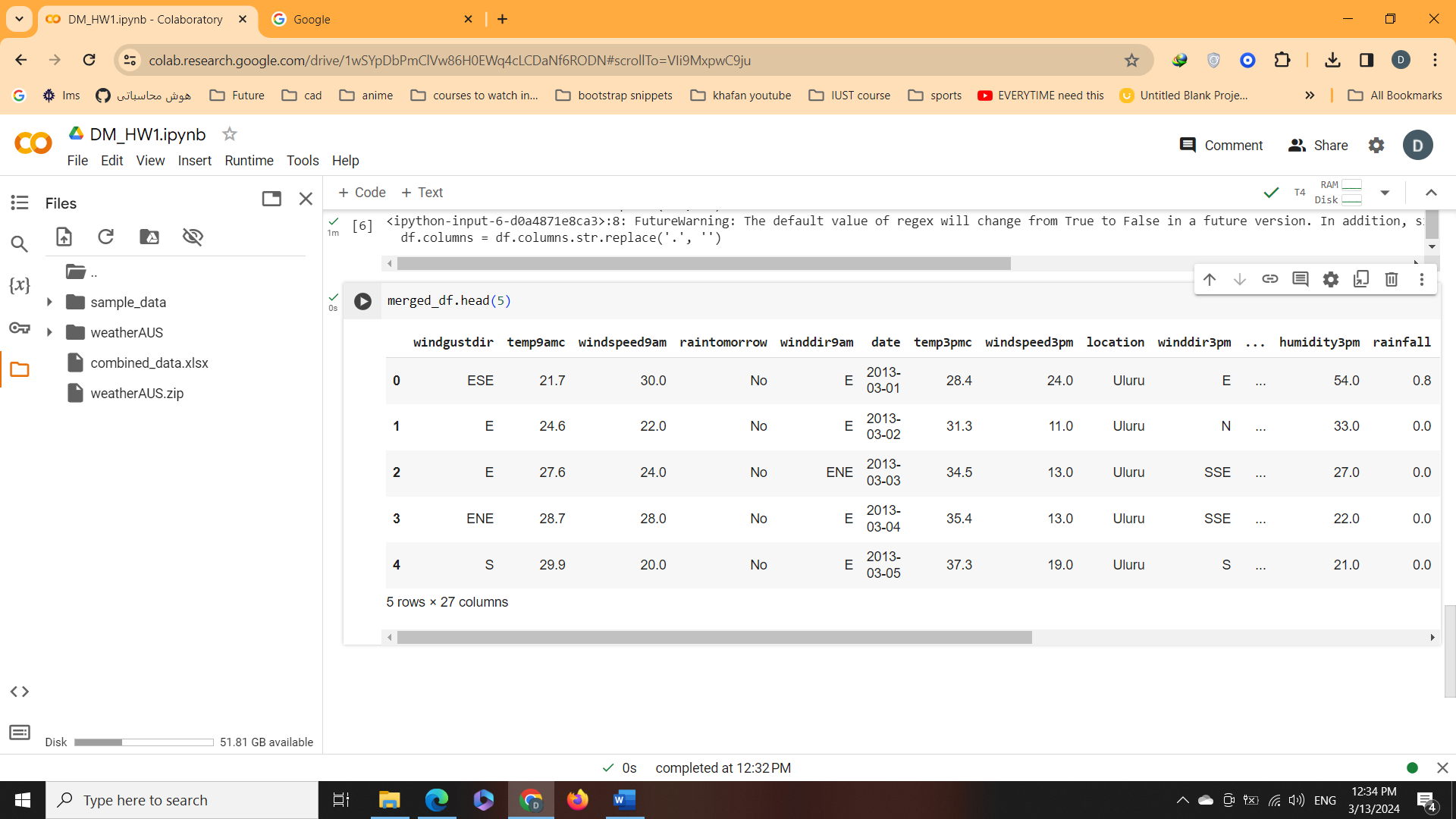
کد را هم از این جا به بعد روی google colab ران میکنیم چون حجم محاسبات زیاد شده است و میتوان از TPU GPU استفاده کرد.

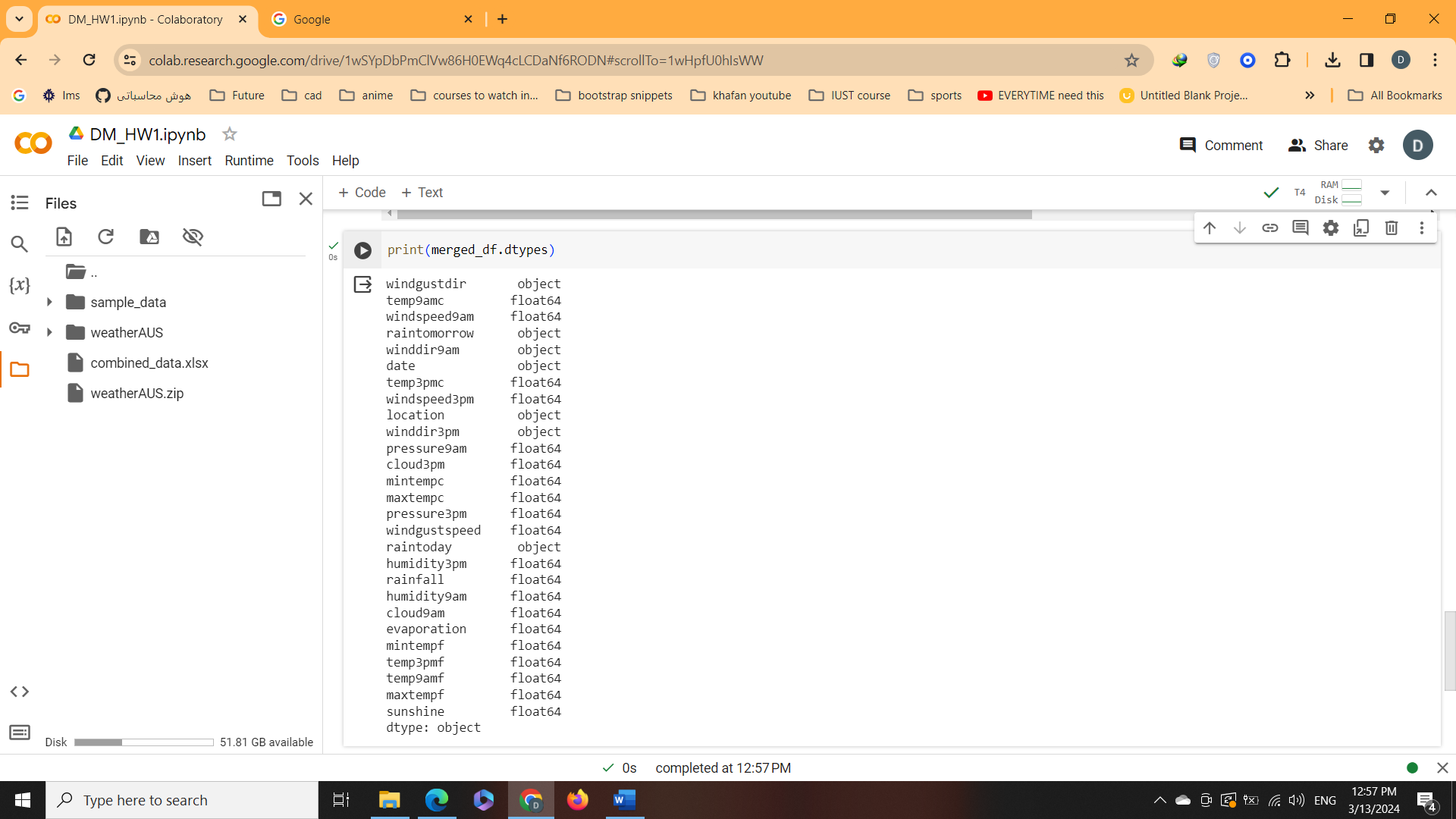
در مورد توضیح کد هم ابتدا تمامی فایل های csv را میخواند نام ستون ها را نرمال میکند و سپس با کمک concat آنها را با هم ترکیب میکند و در نهایت در یک فایل excel ذخیره میکند خروجی هم ساختاری مانند شکل زیر دارد:



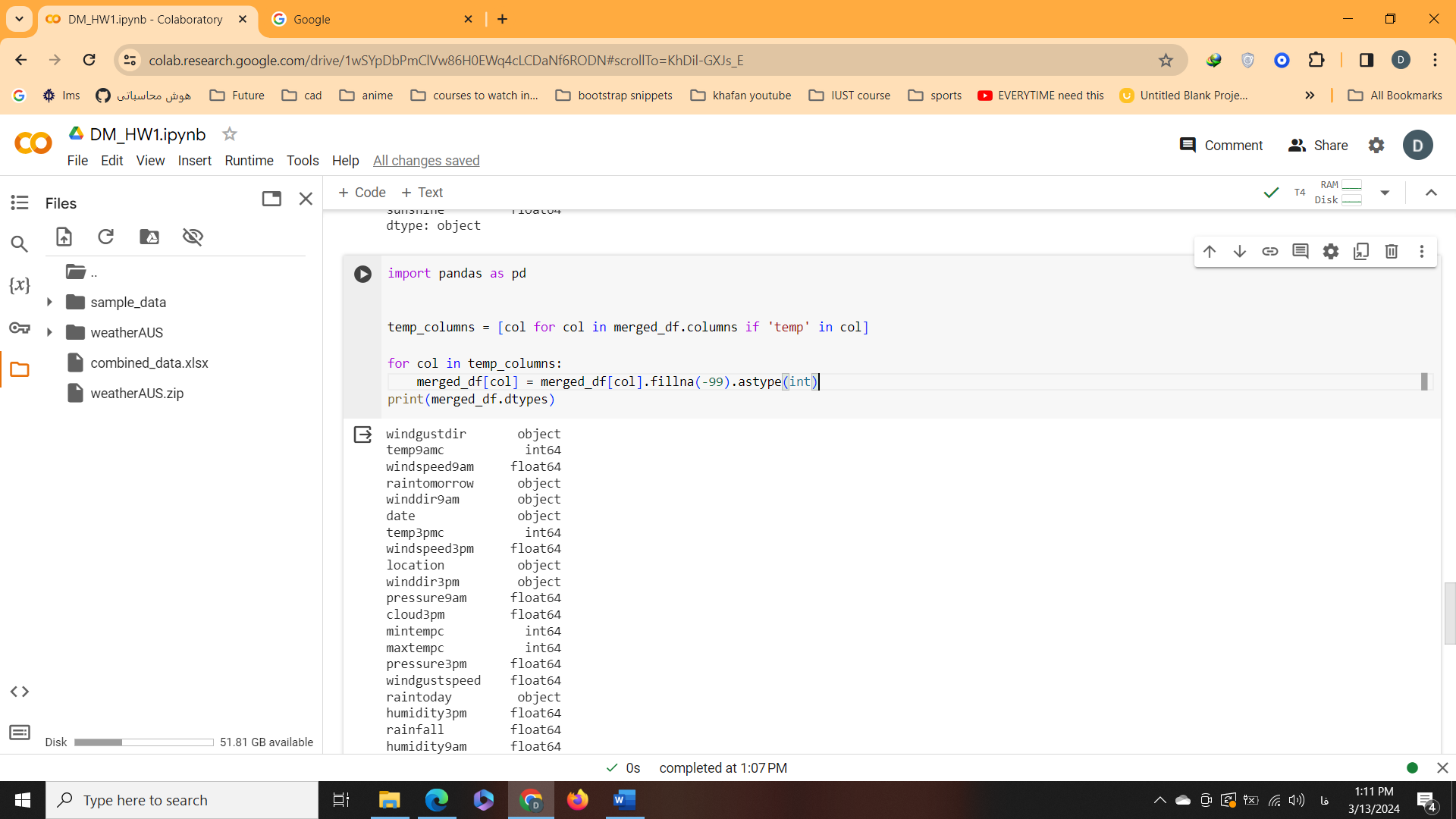


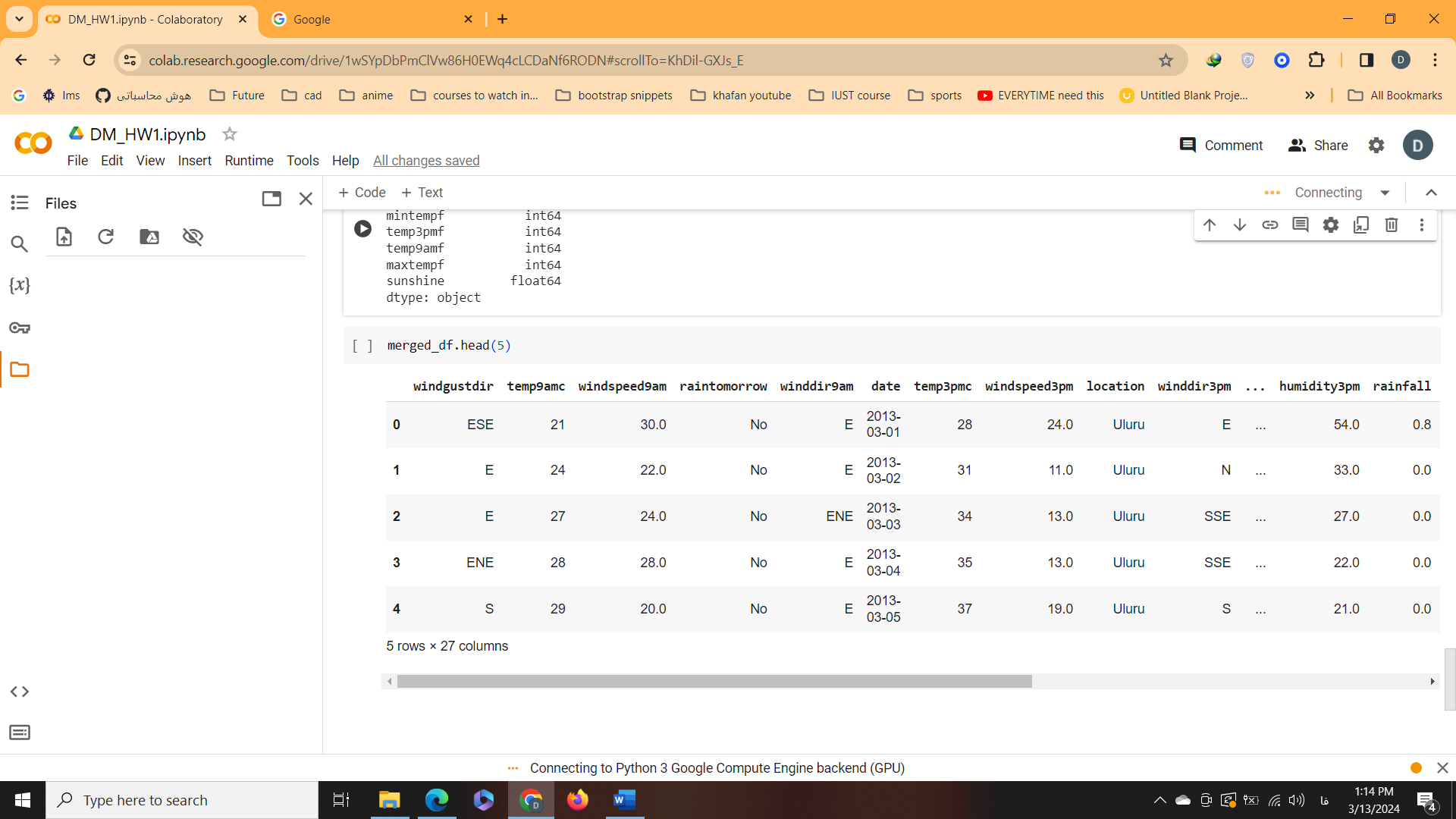
2.پنج سطر ابتدایی به صورت زیر است :



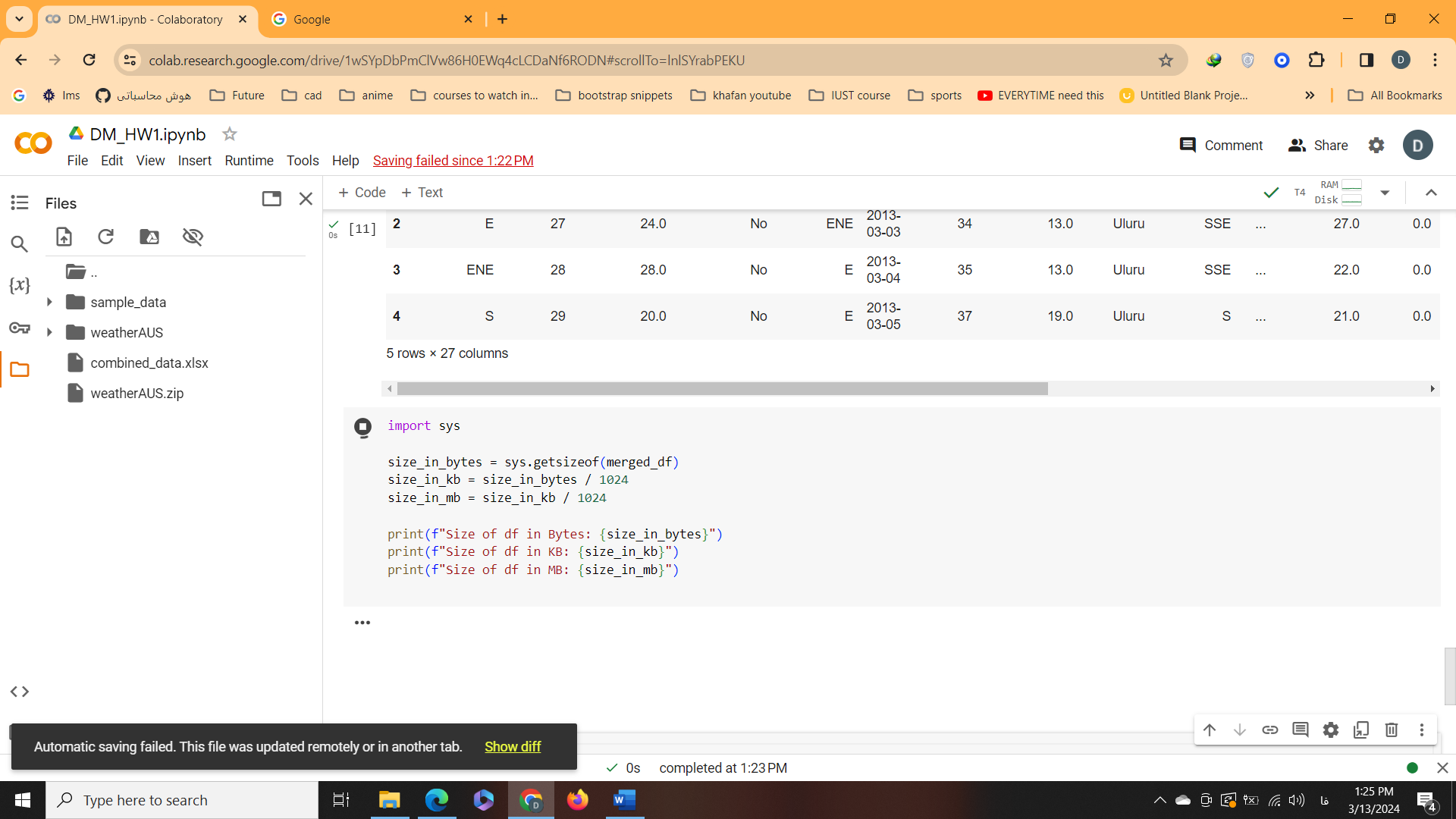
3.به کمک dtypes این کار را انجام میدهیم :

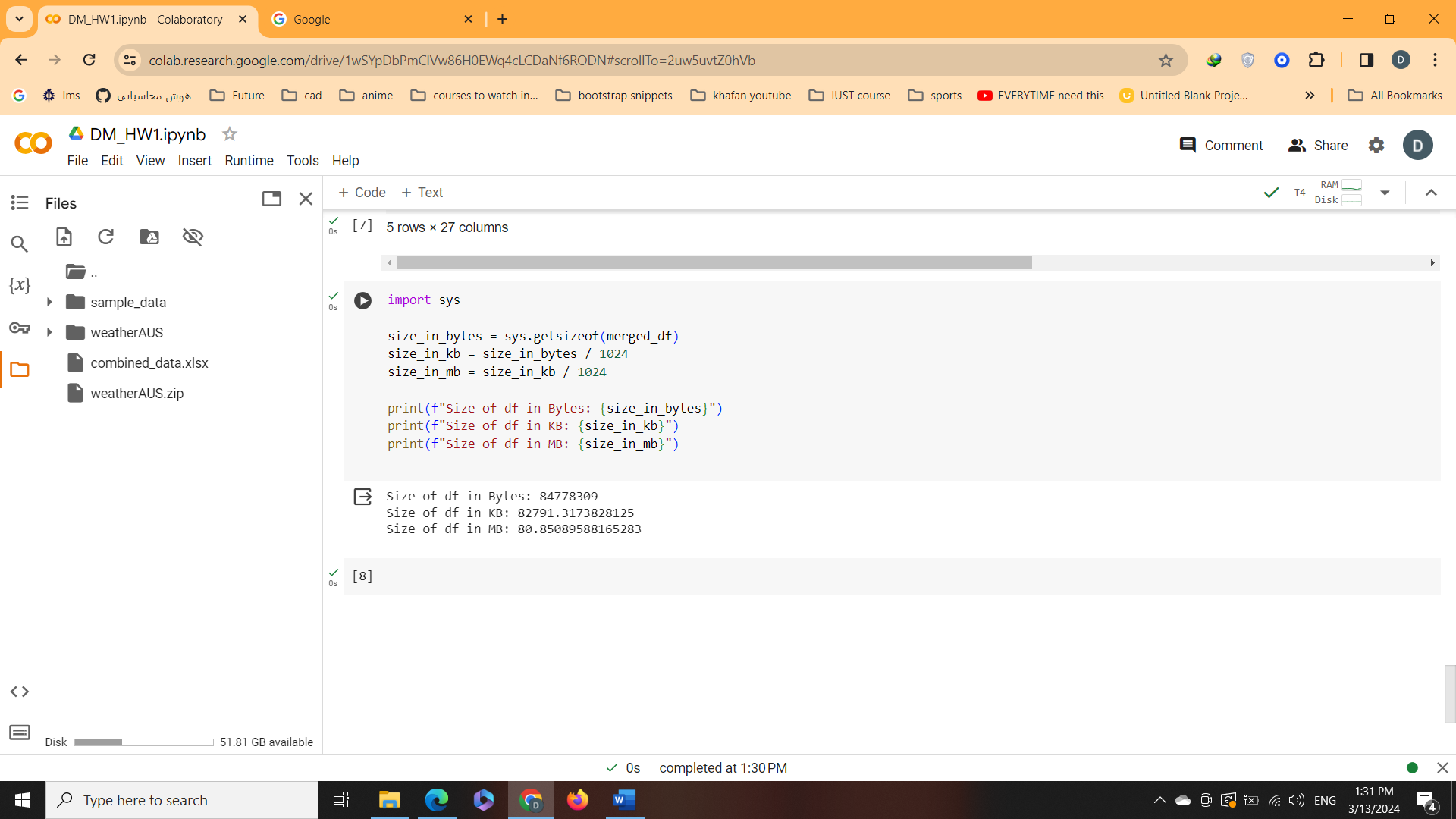
4.برای اینکار ابتدا لازم است که تمامی ستون هایی که temp دارند را شناسایی کنیم سپس تایپ آنها را به int تغییر دهیم، همچنین در اینجا یک چالش داشتیم آن هم اینکه Nan را نمیتوان به Int تغییر داد برای همین این مقادیر را در این ستون ها به مقدار 99- تغییر دادیم که در داده های خودمان غیرممکن است ( اگر با صفر پر میکردیم ممکن بود داده واقعا در آن لحظه صفر باشد و در ادامه به مشکل میخوردیم). خروجی و کد را در ادامه میبینیم :



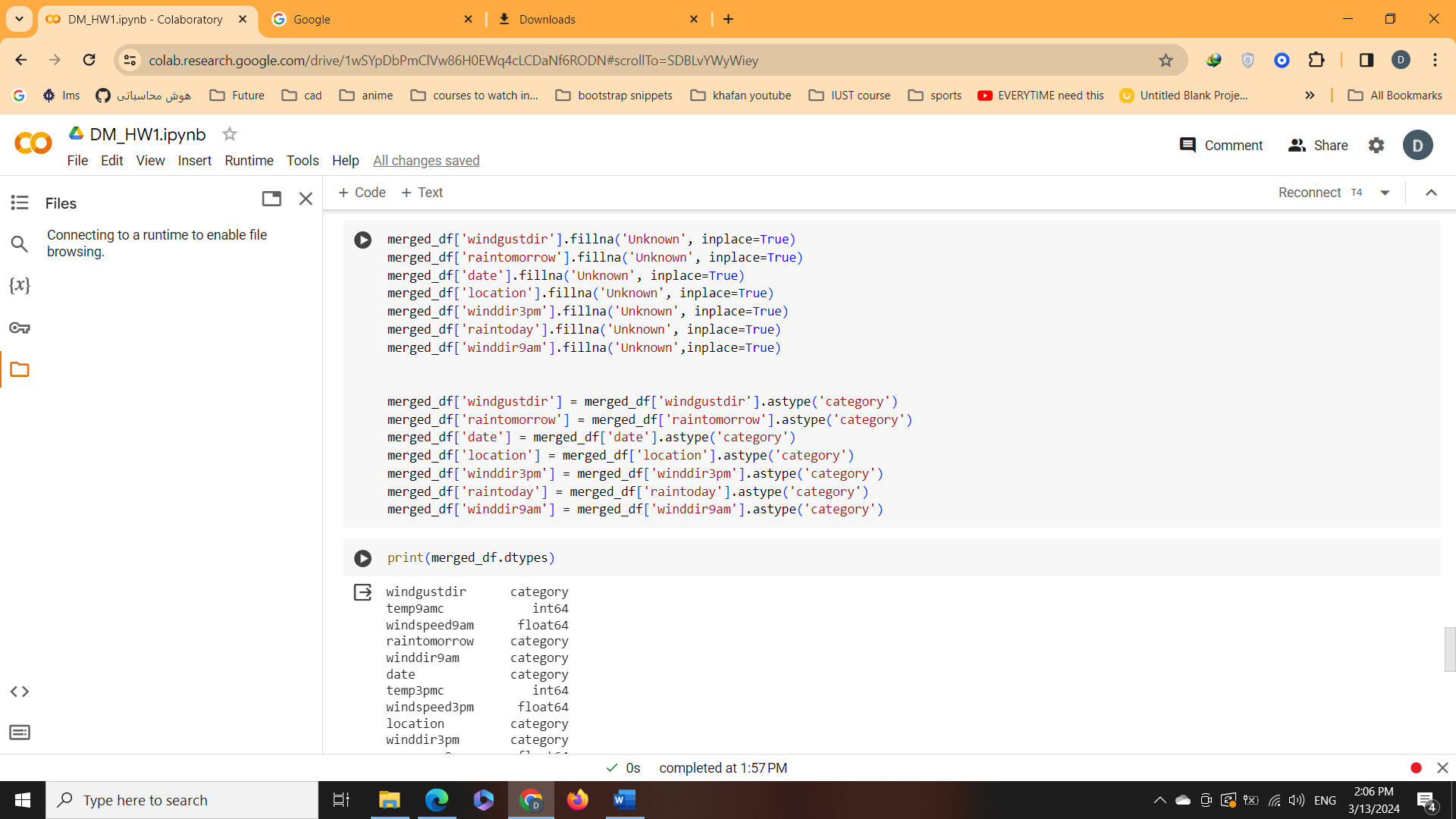


5.برای بدست آوردن سایز دیتافریم در RAM از کتابخانه sys استفاده میکنیم :

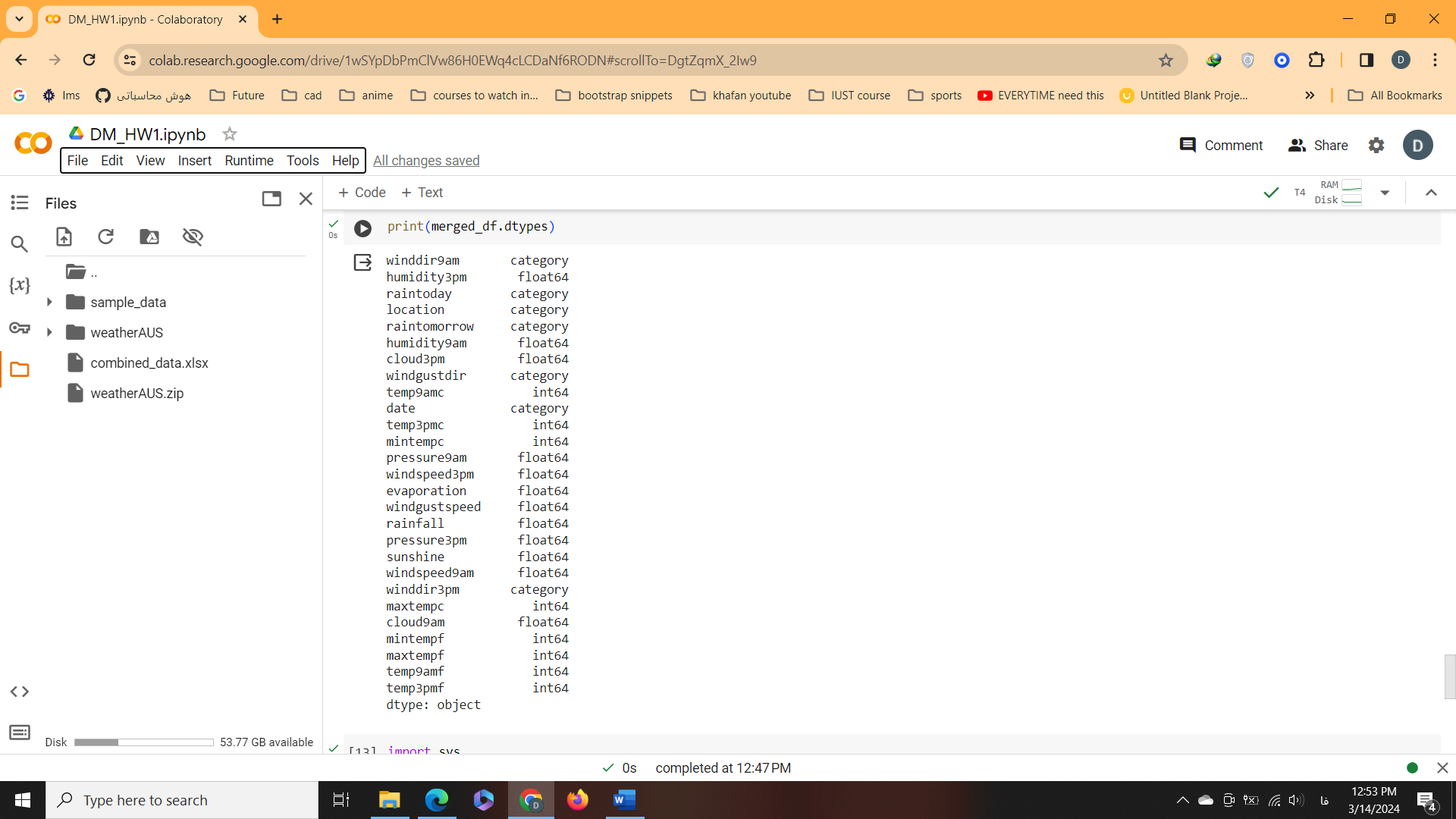
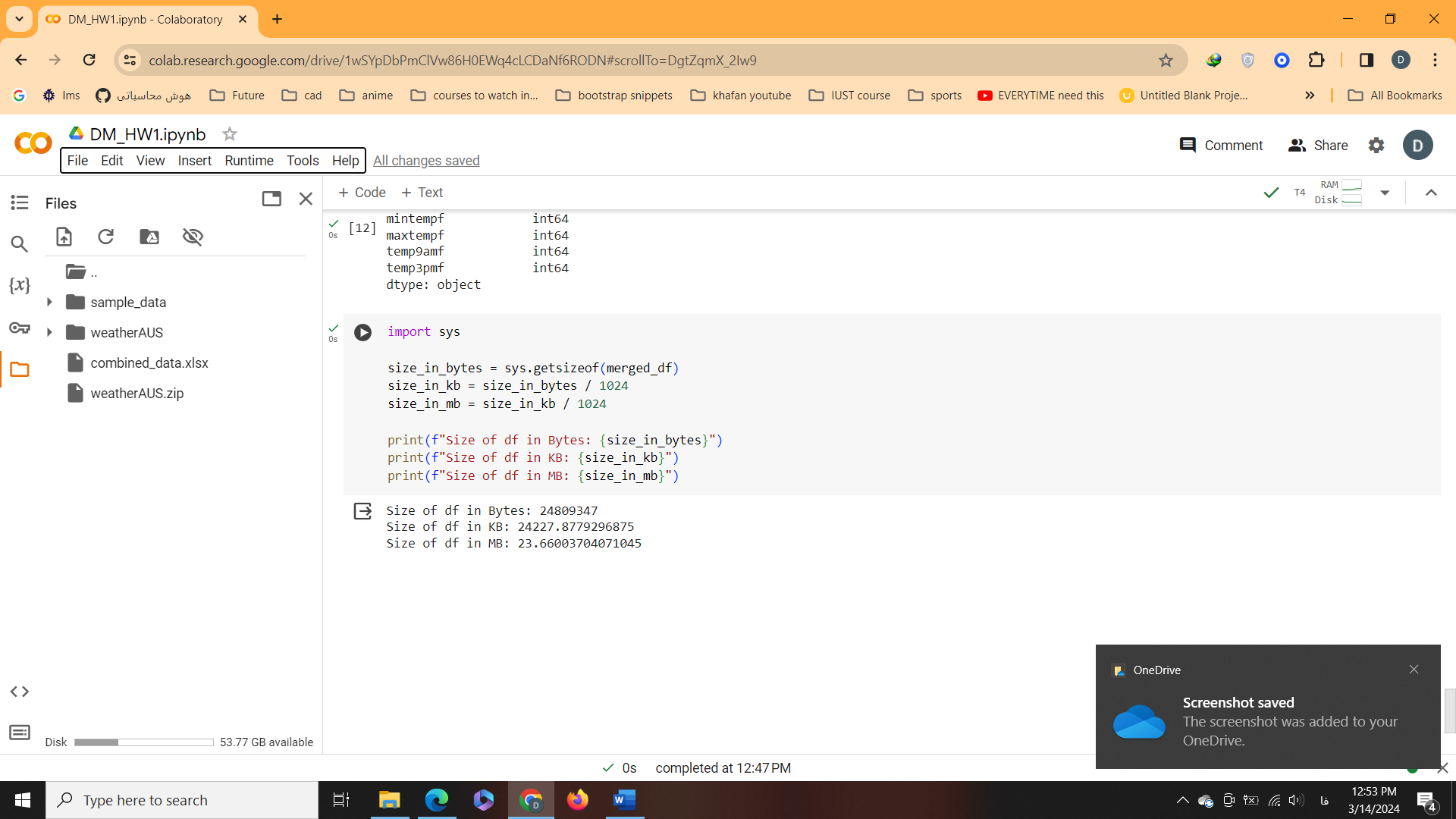




6. بدین صورت تمامی متغیرهای باینری و اسمی را به category تغییر میدهیم، ابتدا مقادیر را از NULL به Unknown تغییر میدهیم سپس به کمک astype به category تبدیل میکنیم :

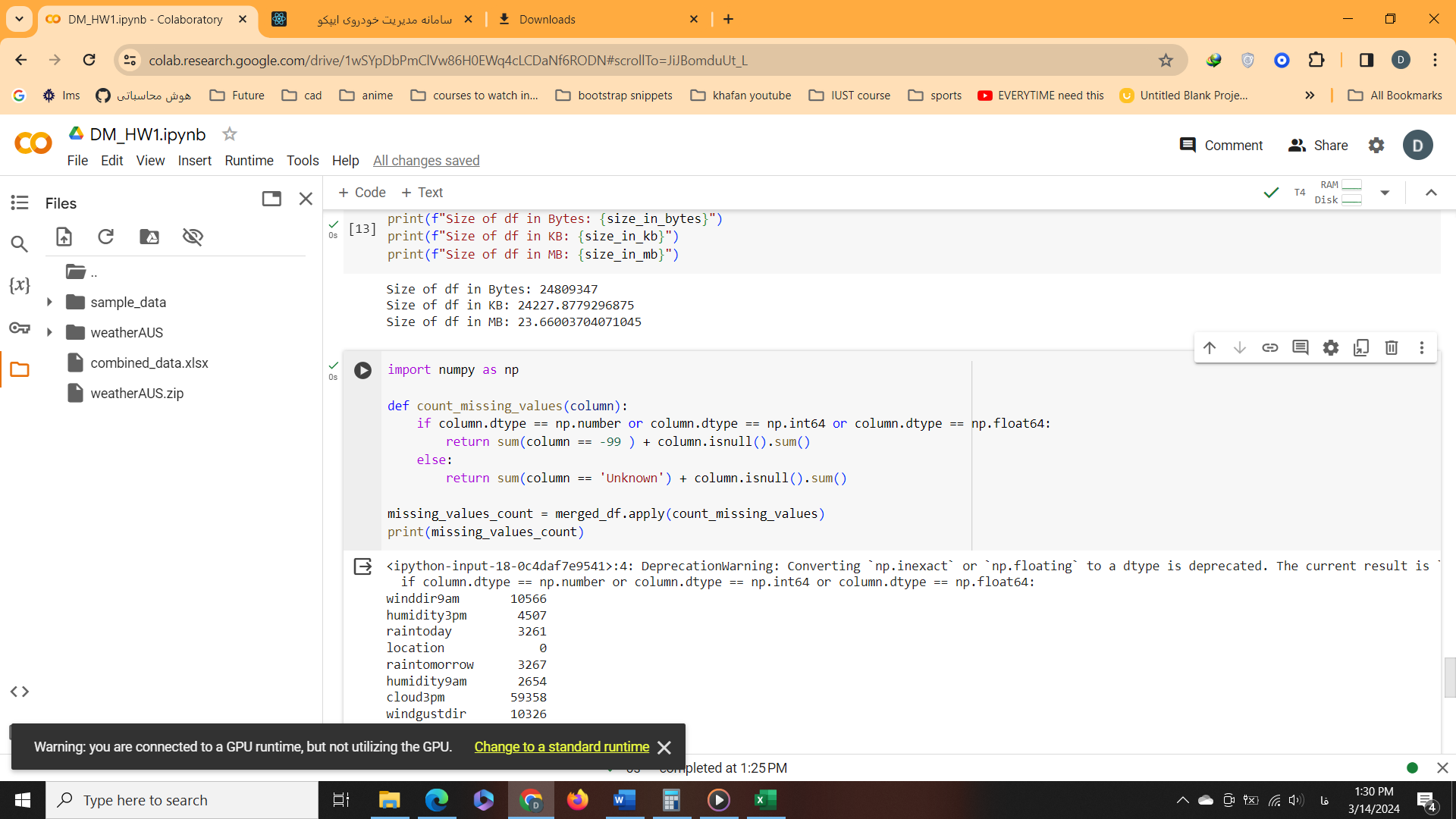


7.با توجه به تغییرات داده شده و تغییرات که در ادامه میبینیم حجم به 23 مگابایت کاهش یافت:

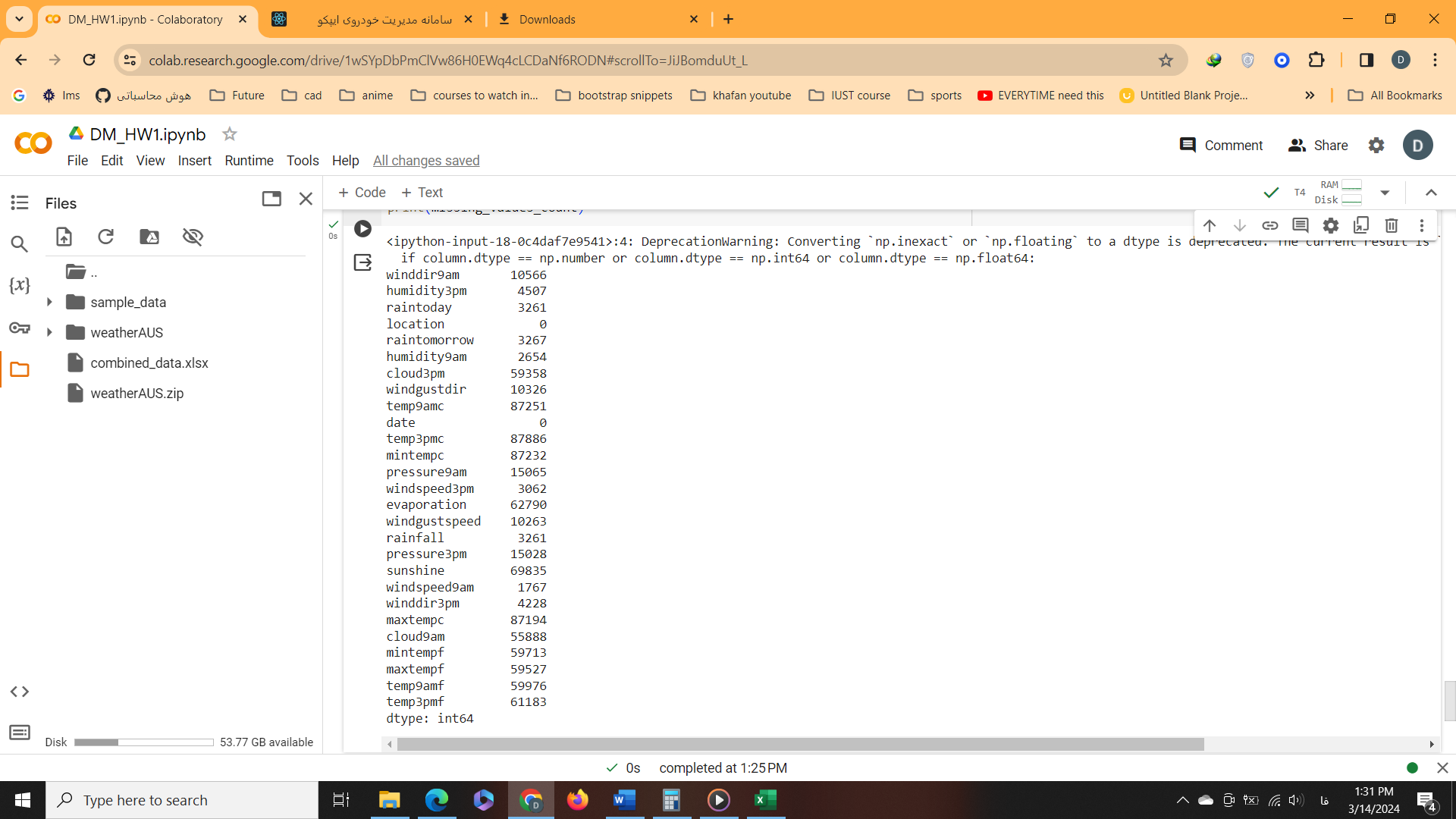


که اگر بخواهیم تغییرات و درصد آن‌ها را نمایش دهیم به صورت مقابل میشود : 23 – 80 که 57 مگ کاهش یافته است که درصد آن هم به صورت حدودا 71% کاهش داشتیم.

8.برای هر کدام از ستون ها چون یک سری مقادیر را برای محاسبات تغییر دادیم لازم است تا اندکی تابع محاسبه گر تعداد را تغییر دهیم:



که نتایج آن به صورت زیر است :

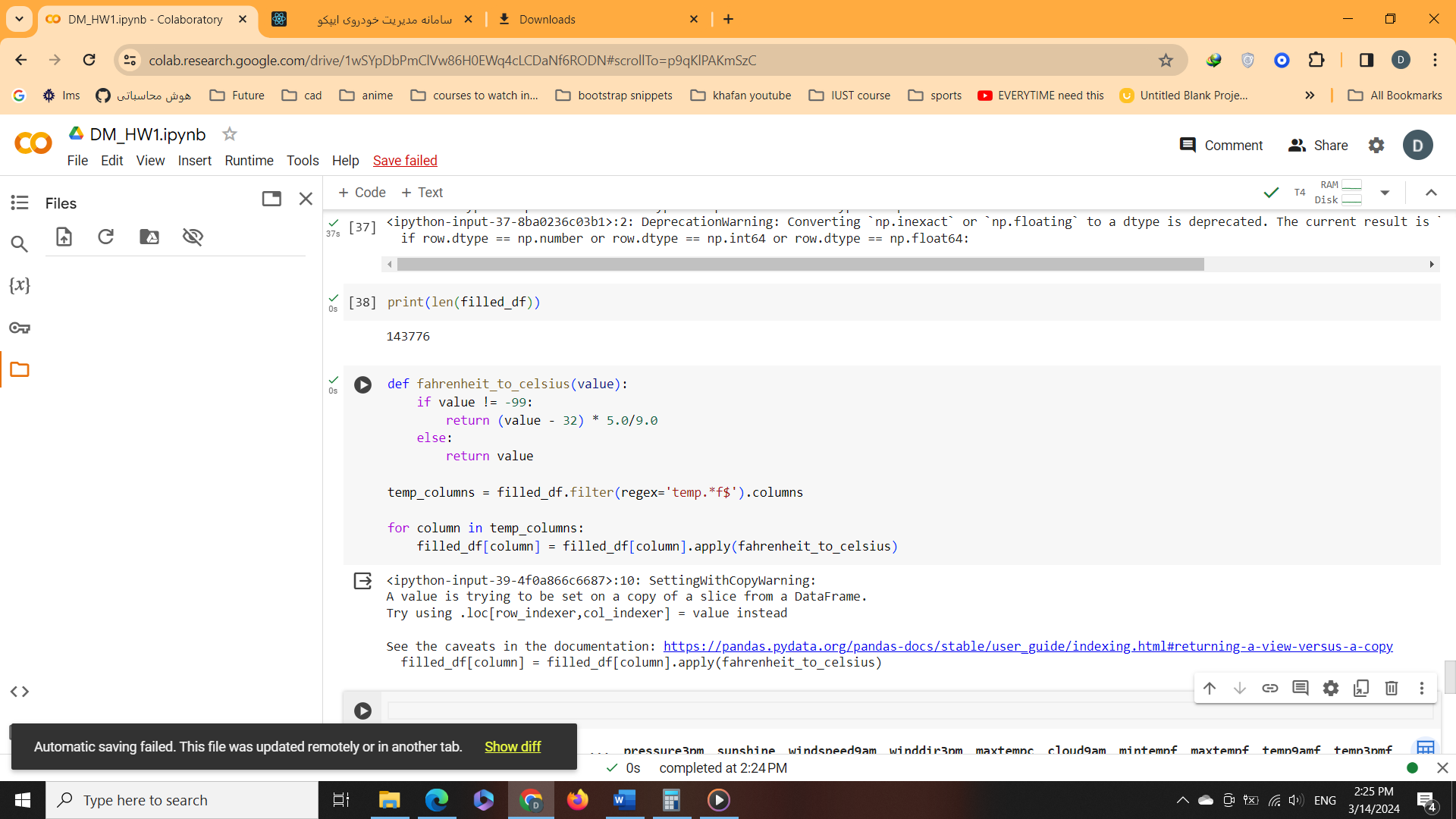


9.برای ستون هایی که خودمان تغییر دادیم مثلا temp9amc که ممکن است در یک ایستگاه کلا بر مبنای فارنهایت ارائه شده باشد، حذف کردن سطر کاملا کار اشتباهی است، چون دیتا کامل موجود است ولی در سطرهایی که مثلا جهت باد وجود ندارد و ما هم نمیتوانیم پیش بینی داشته باشیم حذف سطر منطقی است همچنین در سطرهایی که مثلا داده قبلی و بعدی را داریم، با یک تقریبی میتوان گفت که داده میانی هم مثل قبلی و بعدی بوده است، یا مثلا بین دو روز ابری، احتمال زیاد روز بین هم ابری بوده است، همچنین در مورد داده های عددی مانند دما میتوان با یک احتمال خوب میانگین چند داده حول آن را در نظر گرفت، همچنین این مورد پر کردن داده های گم شده کاملا به کاربرد مورد استفاده ما بستگی دارد شاید مثلا تنها برای ما مفید باشد و از آن استفاده کنیم پس اگر رطوبت یا ابری بودن را نداشتیم، مشکلی بوجود نمیاید. همچنین یک روش دیگری که میتوان استفاده کرد و بنده هم سعی میکنم این روش را در نظر بگیرم این است که چندین ستون با هم چک شوند اگر حجم زیادی از دیتا ( تعداد زیادی ستون) از دست رفته بود آن سطر را حذف میکنیم اینطوری هم مطمئن میشیم اشتباه حذف نمیکنیم(مثال توضیح داده شده در سطر دوم همین صفحه) هم اگر دیتای باقی مانده مفید نبود حذف میشود.

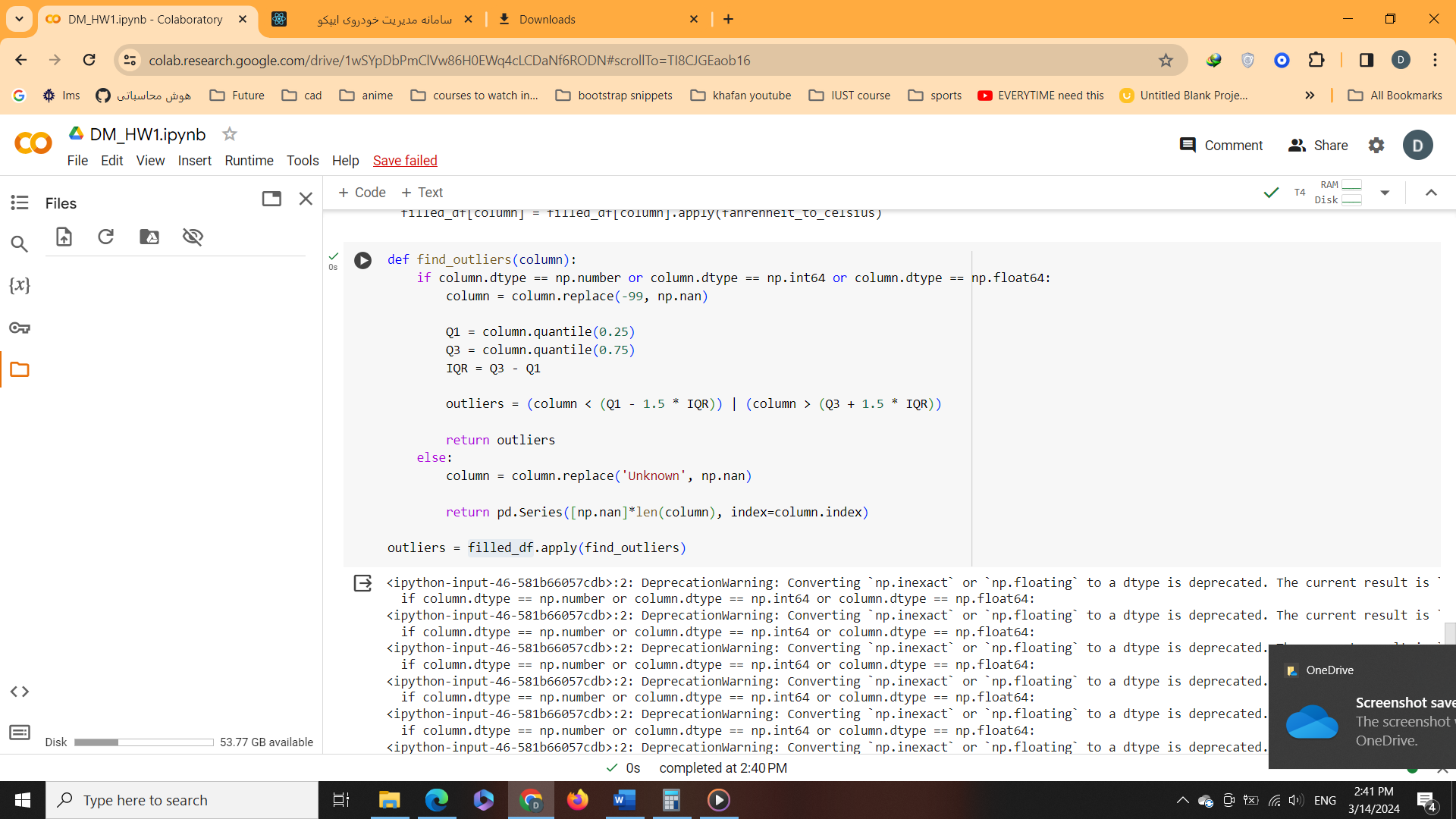
10. با توجه به توضیحات بالا کد را پیاده سازی میکنیم ، بدین صورت که در هر row مقدار missing value ها را پیدا میکنیم و بر مبنای آن عمل میکنیم :

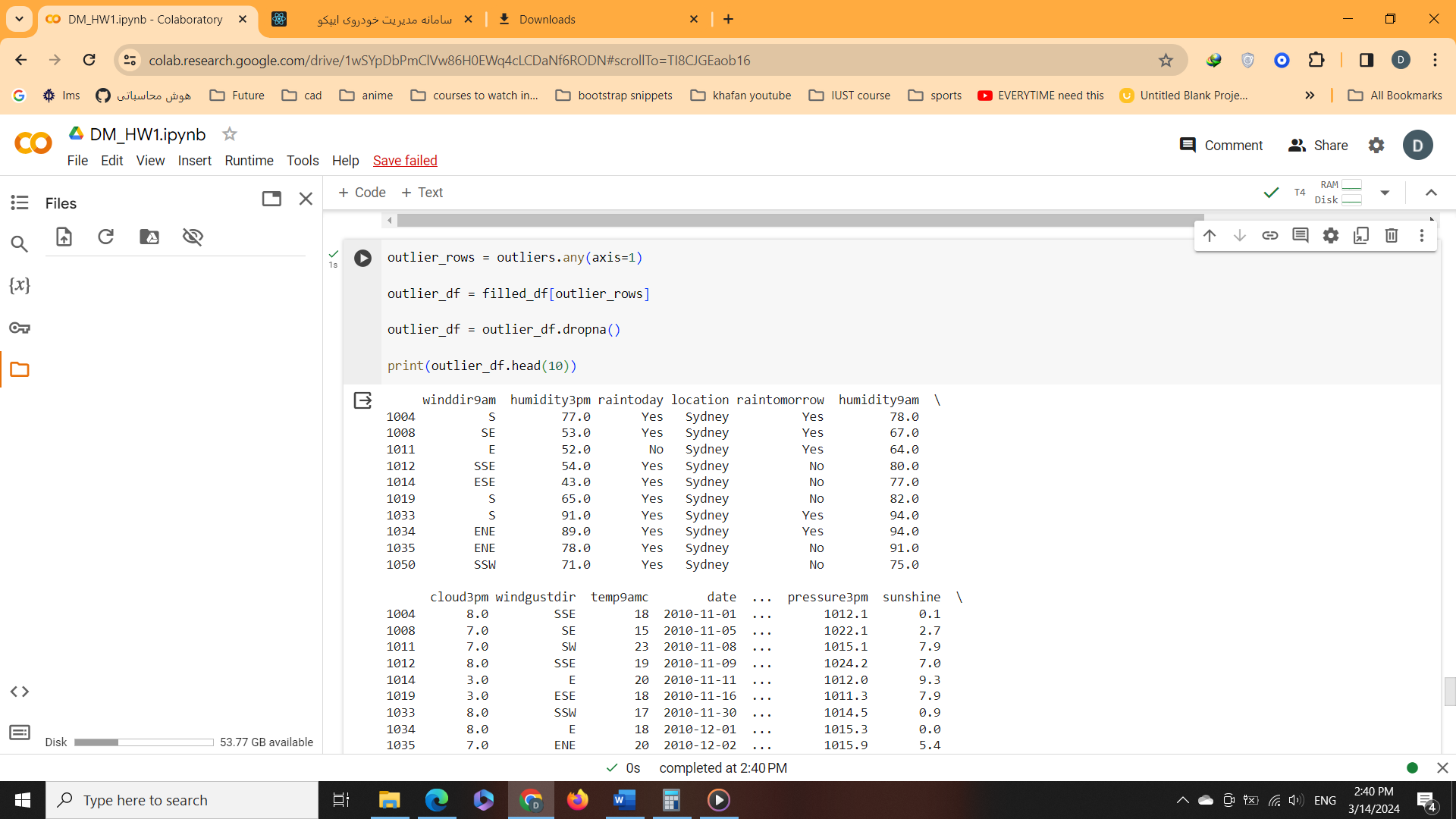


11. برای اینکار ابتدا ستون های را پیدا میکنیم که در اسم ستون temp وجود داشته باشد و با f تمام شود سپس با کمک فرمول تبدیل فارنهایت به سلیسیوس این تبدیل را انجام میدهیم :



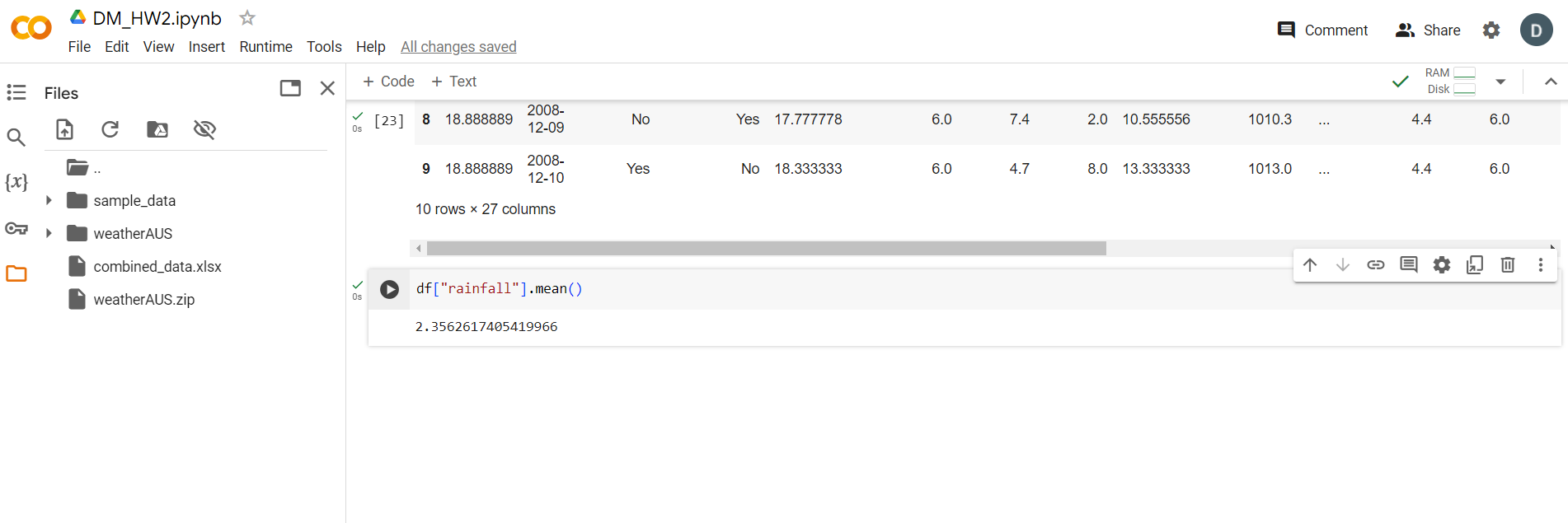
12. از روش چارک برای این کار استفاده کردیم، برای ستون های عددی چارک اول و سوم را حساب کردیم سپس به کمک این دو مقدار IQR را حساب میکنیم، سپس داده هایی که که بیش از 1.5 برابر IQR، کوچکتر از چارک اول هستند یا همین مقدار بزرگ از چارک سوم هستند را به عنوان outlier تشخیص میدهد که برای درک این مقادیر تعدادی را نیز چاپ کردیم:





**سوال دوم)**

1)در قسمت اول خواسته شده میانگین بارش در استرالیا را بدست بیاوریم، با توجه به پیش پردازش مناسب تنها لازم است که از فانکشن built-in میانگین استفاده کنیم کد یک خطی این قسمت به صورت زیر است :



2) در این قسمت ابتدا برای تغییر نکردن دیتافریم اصلی، یک کپی از آن میگیریم در مرحله بعد نوع ستون زمان و را به datetime تغییر میدهیم که کار کردن و فیلتر کردن سال راحتتر باشد، در ادامه ابتدا لوکیشن را به watsonia فیلتر میکنیم و همزمان میگوییم سطرهایی را انتخاب کن که در datetime آن ها بخش مربوط به سال برابر 2015 باشد در ادامه در خط بعد تنها تعداد روزهایی که بارانی بودند را به کمک raintoday بدست میاوریم و تعداد Yes ها را پرینت میکنیم.( که در اینجا 78 جواب شد در صورتی که خودم که در اکسل فیلتر کردم دیدم این مقدار برابر 79 شد با جست و جو و بررسی مورد فهمیدم که یک سطر آن تقریبا خالی بوده است و دیتای خاصی نداشته (طبق شرطی که گذاشته بودم در پیش پردازش اگر از یک حدی کمتر دیتا داشت کلا حذف میکند) )

در ادامه کد این قسمت را مشاهده میکنیم :



3) برای این قسمت ابتدا لوکیشن را فیلتر میکنیم به طوری که تنها سطرهایی داشته باشیم که مربوط به شهر Townsville باشند، سپس با توجه به اینکه در روز دو بار رطوب سنجیده میشود یک ستون جدید تعریف میکنیم که به ازای هر سطر این مقدار برابر با مقدار بزرگتر بین این دو رطوبت است، سپس برای بیان ماکسیمم کل از این ستون max میگیریم، همچنین برای بهتر شدن مفهوم آن روز و تاریخ متناسب با آن را هم پرینت میکنیم که در ادامه مشاهده میکنیم برابر 100 است و در تاریخ 13-04-2014 رخ داده است.

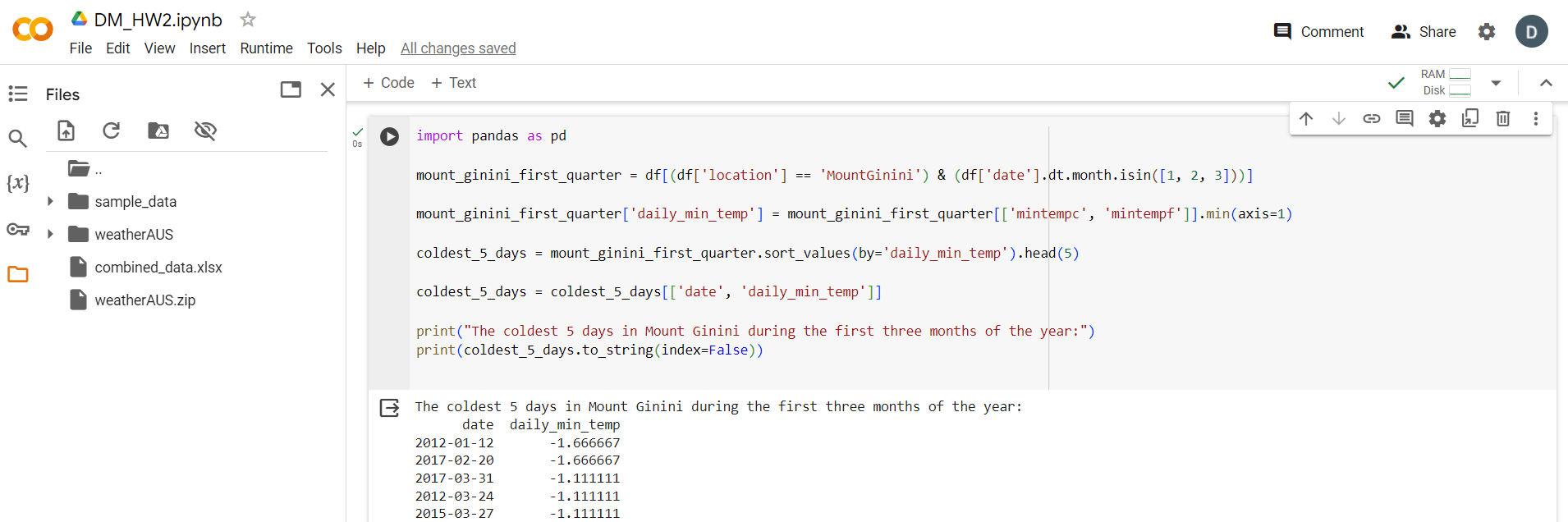


4)در این قسمت هدف محاسبه حداکثر اختلاف دما در ماه اول میلادی است، چیزی که مقداری این مسئله را سخت میکند بحث فارنهایت و سانتیگراد است که با توجه به پیش پردازش کامل این مشکل حل شده است و تمامی دماها به سانتیگراد تبدیل شده است پس دیگر کاری ندارد، ابتدا دوباره برای راحتی کار با تاریخ آن را به datetime تبدیل میکنیم و فقط آنهایی انتخاب میکنیم که ماه آنها برابر 1 باشد، در ادامه یک ستون جدید از اختلافها تشکیل میدهیم البته چون در دو ستون هستند لازم است در هر کدام مقدار بزرگتر یا کوچکتر را انتخاب کنیم (دقت کنیم که جفت آن ها به سانتیگراد است)، در نهایت در ستون جدید مقدار ماکسیمم این تفاوت را پرینت میکنیم. در ادامه کد این قسمت را مشاهده میکنیم:



5)در این قسمت از این قابلیت که در مرحله قبل date را تبدیل به آبجکت تایپ datetime کردیم استفاده میکنیم، بدین صورت که ابتدا شهر را فیلتر میکنیم و همزمان چک میکنیم آیا ماه در 1و2و3 هست یا خیر که این یعنی سه ماهه اول هر سال.

حال که دیتایی که میخواستیم را بدست آوردیم به سراغ 5 روز سرد میرویم، که در این رابطه ابتدا داده ها را سورت میکنیم(بر مبنای ستون جدیدی که تعریف کردیم و مینیمم هر روز را دارد) سپس 5 تای اول آن که کمترین ها هستند را برمیداریم ( به همراه تاریخ ) کد و خروجی در ادامه دیده میشود :

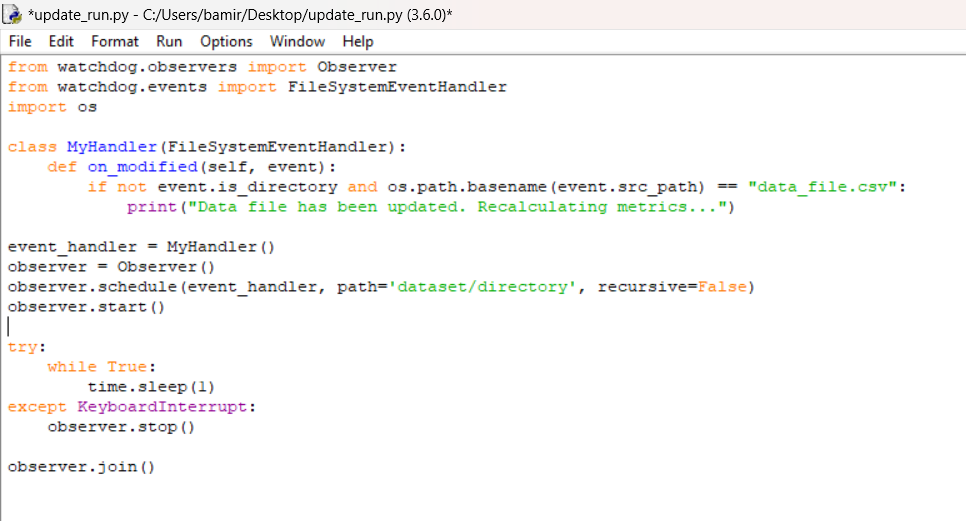


3) در این بخش ابتدا لازم بود یک مقدار تحقیقات انجام دهیم، با توجه به نتایج یافت شده چندین حالت داریم، 1. به صورت مرتب دیتا اضافه شود و لازم باشد بعد هر آپدیت این متریک ها هم آپدیت شوند

2.هر شب آپدیت کنیم

3.هر وقت فایل ها دیتاست تغییر کردند آپدیت کنیم

که با توجه به مثال ما گزینه سوم مناسب تر بود، در این زمینه از لایبرری watchdog میتوان استفاده کرد و ران کردن آن هم بدین صورت است که یک Observer تعریف میکنیم و روی دایرکتوری دیتاست قرار میدهیم سپس به ازای on\_modified لازم است تا چک کنیم اگر در دایرکتوری بود و مربوط به دیتاست ما بود در ادامه فانکشنهایی که جدا کردیم را فراخوانی مشکلی که در این قسمت داریم با توجه به ران کردن فایلمان در کولب امکان تست کردن این ویژگی وجود ندارد چون کولب این قابلیت را ساپورت نمیکند ولی در ادامه کد این قسمت را مشاهده میکنیم که یک سری جاها خالی هستند و میتوانند با آدرس دیتاست یا اسم فایل آن کامل شوند و فایل آن هم در کنار نوتبوک قرار داده شده است :



در ادامه پرینت هم فانکشن هایی که در زیر تعریف میکنیم قرار داده میشوند :

