

با عرض سلام و خسته نباشید خدمت استاد گرامی و دستیاران آموزشی خدمت شما عارضیم که گروه شماره 13 متشکل از 4 عضو است ؛ مائده محمدی ، هلیا عطارد، مرضیه قنبری و علی فهیمی برای درک بهتر پروژه، این گزارش را درست کردیم که شامل توضیح مختصری از شرح توابع ، منابع و وظایف اعضا گروه و نمونه خروجی بصری می باشد.

### شرح وظایف و نحوه انجام پروژه:

حجم اعظم کار این پروژه در یکی دو هفته اخیر انجام شد . به این صورت که در ساعات معینی در Skype جمع می شدیم و یا با anydesk و ... به هم دیگر در ایده و دیباگ کردن کد کمک می کردیم و این کار رو تقریبا هر چهار پنج روز به طور پیوسته انجام می دادیم ... ناگفته نماند که گاهی هم که همه اعضای گروه بن بست میخوردیم ، سعی میکردیم از آشنایان مجرب تر راهنمایی بگیریم.

قسمت اعظم کد را به تعدادی تابع مجزا تقسیم کردیم و در انتها توابع و دیگر بخش های کد را بین اعضا تقسیم کردیم که جزئیات آن آورده شده است.

اعضا	توابع
مرضیه قنبری	Record,add,apply,div
هلیا عطارد	Plot, time-obj,invoice
مائده محمدی	Inv,fb,main loop for running
علی فهیمی	Plot,analys

و همچنین نوشتن گزارش و نوشتن کد های مربوط به دریافت داده ها به عهده هلیا عطارد بوده است. و دیگر بخش های کد مانند ایجاد فایل JSON و پیوند زدن بخش های مختلف و دیباگ کلی ، با نرم افزار anydesk و ... با شرکت کل اعضا انجام شده اند.

### برای اجرای برنامه باید دو مرحله انجام بشه:

`python prepare_input.py > input.txt`

در بخش اول داده های ارسالی توسط TA در پوشه مربوطه قرار می گیره) با فرمت (csv ، این کد اطلاعات لازم را به فرمت مورد نظر برنامه درمیاره تا قابلیت خواندن توسط برنامه ی اصلی را داشته باشد. در نتیجه تنها با تغییر فایل ها می توان بسادگی ورودی مورد نظر برنامه را تولید کرد.

python manager.py < input.txt

با این دستور ورودی برنامه بصورت اتوماتیک از فایل input.txt خوانده می‌شود. اطلاعات بصورت در این فایل ذخیره شده که برنامه ابتدا اطلاعات قبلی خود را ریست می‌کند سپس اطلاعات جدید را دریافت می‌کند و در نهایت خروجی‌های مورد نظر را تولید و به نمایش در می‌آورد.

### این برنامه شامل:

یک فایل پایتون 'manager.py' که برنامه اصلی قابل اجرا است.  
یک فایل csv با نام 'accounts.csv' که حاوی تراز مالی هر واحد و صندوق ساختمان است.  
یک فایل csv با نام 'check\_list.csv' که برای نگداری داده های تراکنش ها استفاده میشود.  
یک فایل 'building.json' که حاوی اطلاعات اولیه ساختمان شامل شماره واحد ها و متراژ و نام ساکن و... است.

\*با اجرا برنامه در هر مرحله برای گرفتن ورودی متن راهنما چاپ میشود.

### دستورات:

**Record .1 :** گرفتن ورودی برای ثبت تراکنش در فایل check\_list (فرمت گرفتن دستور بعد از تایپ کلمه record در برنامه چاپ میشود)

فرمت دستور:

Time category subcategory price related \_units responsible\_unit div describe

نمونه دستور:

2020-2-6 repair installation 300 all d "motor\_repair"

\*دقت شود که دستورات حتما فقط با یک فاصله وارد شوند و اطلاعات اضافه (یا فاصله اضافی) وارد نشود

\*لازم نیست عدد قیمت را منفی وارد کنید برنامه به صورت خودکار شارژ را از هزینه ها جدا میکند

\*دقت کنید که فرمت تاریخ وارد شده حتما به صورت "سال-ماه-روز" میلادی از چپ به راست باشد و با علامت - جدا شده باشند؛ اگر کلمه 'now' را وارد کنید تاریخ جاری وارد میشود

\*به املای کلمات وارد شده دقت کنید زیرا اگر اشتباه وارد شوند برنامه تراکنش را ثبت نمی کند

\*اطلاعات دسته بندی ها و زیر دسته ها در تصویر پایین آمده(حرف کوچک جلوی هر نام مربوط نحوه تقسیم ان تراکنش بین واحد ها به صورت پیش فرض است)

دسته بندی ها شامل: شارژ(شارژ)، اسانسور(اسانسور)، قبض(برق، اب، گاز)، خدمات(نظافت، باغبانی، امنیت)، تعمیرات(تاسیسات، ساختمان، سیستم برق)، متفرقه

```
{
  #catrgories informations
  "charge":{"charge":"c"},
  "elevator":{"elevator":"e"},
  "bill":{"electricity":"e","water":"r","gas":"a"},
  "service":{"cleaning":"e","gardening":"e","security":"p"},
  "repairing":{"installation":"a","building":"p","electical":"e"},
  "other":{"other":None}
```

\*شارژ به صورت مجزا از هر واحد دریافت میشود

**1.1 apply:** یک تابع داخلی برای record است برای تقسیم پول برای واحد های خاص یا تمامی واحد ها میباشد.

**1.2 add:** یک تابع داخلی برای record برای اضافه کردن تراکنش هاست که تراکنش های جدید را اضافه میکند.

**2. report:** مربوط به قسمت گرفتن گزارش است و پس از وارد کردن راهنمای دستورات بعدی چاپ میشود که به صورت زیر است:

**2.1 inv:** این دستور دو تاریخ دریافت میکند و یک فایل csv صورت حساب شامل تمام تراکنش های بین ان دو تاریخ در مسیر جاری برنامه میسازد که نام ان همان دو تاریخ وارد شده اند

**2.2 fb:** این دستور شماره واحد را از شما میگیرد و تراز مالی ان را چاپ می کند اگر کلمه 'box' وارد شد موجوی صندوق ساختمان را نشان می دهد(دقت شود که عدد منفی هم برای واحد ها و هم برای صندوق به معنی بدهی است)

**Analys 2.3:** این دستور نام زیر دسته را از شما میپرسد و هزینه و سهم آن در کل هزینه ها و سهم هر زیر دسته اش را نشان می دهد.

**Plot 2.4:** این دستور نموداری برای زیر دسته بندی وارد شده رسم می کند

**Reset 3:** این دستور تمام اطلاعات ذخیره شده (اطلاعات فایل های `check_list accounts`) را پاک میکند و برنامه به حالت اولیه بر میگردد. (برای اطمینان از دستور برنامه از شما میخواهد اگر مطمئن هستید `yes` را وارد کنید)

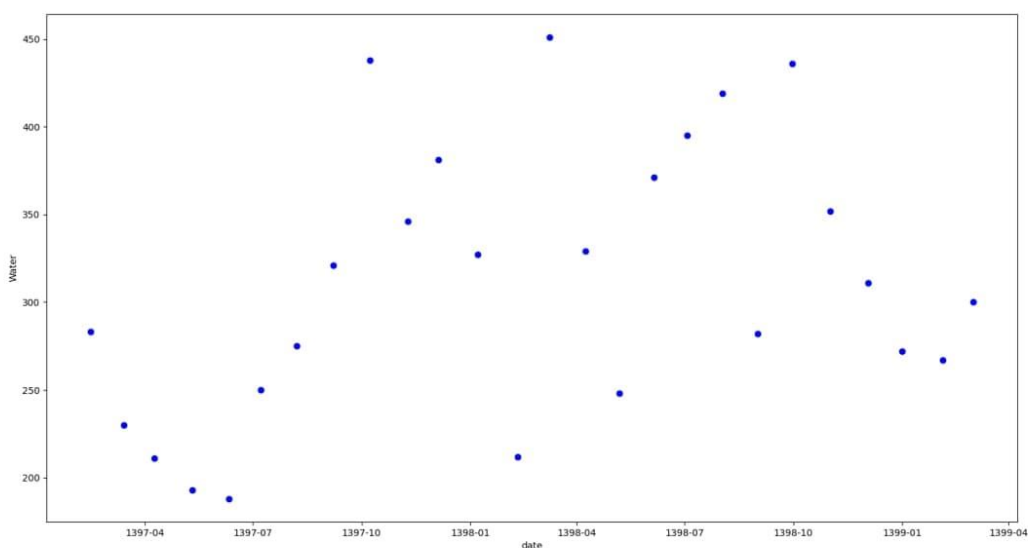
**مزایا:**

\* در هر مرحله با وارد کردن `back` به مرحله قبلا باز میگردید و در اخر با وارد کردن `exit` برنامه خارج میشود.

\* برنامه در صورت وارد کردن اطلاعات اشتباه در هر مرحله دچار خطا نمی شود فقط دستور اجرا نمی شود برنامه به مرحله جاری بر میگردد تا دستور را دوباره وارد کنید.

\* تاریخ در فایل `check_list` و فایل های صورت حساب به فرمت `date` ذخیره میشوند تا در صورت نیاز به کار با اکسل نیاز به تبدیل تاریخ نباشد

نمونه بصری خروجی :



و همچنین در فایلی به نام 1399-01-01 to 1399-10-20.csv نمونه خروجی invoice را داریم.