**توضیحات فایل main.py :**

**کتابخانه ها :**

re : کتابخانه ی ریجکس برای پیدا کردن یک الگو در یک متن

Sys : کتابخانه ی سیستم عامل (برای خارج شدن از برنامه به کار میره)

Json : برای باز کردن و ذخیره کردن فایل های جیسون و txt

Requests : برای دریافت فایل html یک سایت

Bs4 : برای پیمایش و گرفتن اطلاعات مورد نیاز از یک فایل html

Dicttoxml : برای ذخیره کردن یک دیکشنری در یک فایل xml

Xml : کتابخانه ی پیش فرض پایتون (برای زیبا سازی فایل xml استفاده کردم)

PyQt5 : کل برنامه روی این کتابخانه هست

MainPage : فایل gui پنجره ی اصلی هست که از designer ساخته شده .

**برنامه دو بخش داره ، کلاس ContactsManager که کل برنامه باید روش اجرا بشه و از پنجره ی اصلی برنامه ارث بری میکند ، و قسمت if \_\_name\_\_ که برای این هست که اگر برنامه به صورت مستقیم اجرا شد ، کد های داخل آن اجرا شود**

**تابع های کلاس :**

دو تابع اول ، پیش فرض خودش هستند و نیازی به توضیح نیست

تابع Defaults برای اولین اجرا هست و وقتی کلاس صدا زد میشود ، این تابع هم صدا زده میشود تا لیست اصلی را بسازد و یه سری پیش فرض ها را انجام دهد ، مثل اینکه لیبل های ارور را مخفی کند ، قسمت هایی که هنوز نیاز نیستند و صدا زده نشدند را مخفی کند و دو تابع DatabaseToCombo و DatabaseToTable را صدا میزند (توضیحاتشون رو زیر نوشتم)

تابع BtnMainAdd برای استفاده از قابلیت اضافه کردن توی برنامه هست و اگه دکمه ی Add روی سربرگ Contacts بزنی ، این تابع اجرا میشه و فقط یه سری lineEdit و label رو نمایش میده و حالت برنامه رو میذاره روی اضافه کردن (یعنی اگه دکمه ی Apply رو بزنی ، به مخاطبین اضافه میشه نه اینکه حذف بشه یا تغییر کنه مخاطبی)

تابع BtnMainRemove برای نمایش یا مخفی کردن lineEdit ها و label و combobox مربوط به قسمت remove هستند و اگه زده بشه ، حالت برنامه رو هم روی remove میذاره و با زدن دکمه ی apply ، مخاطب انتخاب شده ، حذف میشه

تابع BtnMainUpdate برای نمایش یا مخفی کردن lineEdit ها و label و combobox مربوط به قسمت update هستند و اگه زده بشه ، حالت برنامه رو هم روی update میذاره و با زدن دکمه ی apply ، مخاطب انتخاب شده ، با تغییراتی که داده شده ، تغییر میکنه

تابع BtnApply تابع اصلی در سربرگ Contacts هست ، و با زده شدن دکمه ی Apply فعال میشه ، کارش به این صورت هست که اول میبینه حالت برنامه روی چه چیزی هست (روی remove – add- update هست ؟) بعد همون کار رو با مقادیری که به برنامه دادی ، روی دیتابیس برنامه انجام میده و مخاطبین رو حذف ، اضافه یا آپدیت میکنه

تابع BtnSortFirstName برای این هست که با زده شدن دکمه ی First Name در سربرگ List (بالای جدول) ، مقادیر داخل جدول رو بر اساس First name مرتب میکنه

تابع BtnSortLastName برای این هست که با زده شدن دکمه ی Last Name در سربرگ List (بالای جدول) ، مقادیر داخل جدول رو بر اساس Last name مرتب میکنه

تابع BtnExtract تابعی برای این هست که وقتی دکمه ی extract در سربرگ سوم زده میشه ، متن را از input گرفته و مخاطبین رو ازش جدا کنه (شماره ها و اسم هاشون) و در آخر هم توی دو قسمت output نشون بده

تابع DatabaseToCombo کل دیتابیس (مخاطبینی که توی برنامه وجود دارن) رو توی کمبوباکسی که توی سربرگ اول هست میریزه (وقتی دکمه ی remove یا update رو بزنی ، یه قسمتی اضافه میشه که نوشته select contact ، و روبروش یه لیست هست ، همون لیست هست)

تابع EmptyLIneEdits1 تمام lineEdit ها داخل سربرگ اول رو خالی میکنه

تابع EmptyLIneEdits2 تمام lineEdit ها داخل سربرگ web رو خالی میکنه

تابع ComboEditLine برای وقتی هست که یه مخاطبی از combobox انتخاب میشه ، مقادیر اون مخاطب (مثل ایمیل و اسم و اسم خانوادگی و شماره تلفنش) توی lineEdit ها نمایش داده بشه

تابع DatabaseToTable برای وقتی هست که میخوای دیتابیس رو در جدول وارد کنی (جاهای مختلف کد از این تابع استفاده شده که به محضی که یه مخاطب حذف ، اضافه یا تغییر میکنه ، کل جدول رو آپدیت کنه)

تابع ContactExtract تابع کمکی برای تابع BtnExtract هست و وظیفه ش جدا کردن اسم و شماره از متنه و این مقادیر رو به تابع اصلی یعنی BtnExtract بر میگردونه که روش کار بشه (نمایش داده بشه و ...)

تابع WebScrap برای موقعی هست که دکمه ی Get Data در سربرگ Web زده میشه ، میره و اطلاعات رو از سایت میگیره و اطلاعات مورد نیازش رو ازش جدا میکنه با کتابخونه های requests و bs4 و در آخر اونا رو نمایش میده

تابع WebAdd برای موقعی هست که در سربرگ Web روی دکمه ی Add میزنی ، با اطلاعات داخل lineEdit ها یک مخاطب میسازه و در دیتابیس برنامه ذخیره میکنه (مثل همون Apply در قسمت اول وقتی که حالت برنامه روی Add بود)

تابع EmailVarify برای فهمیدن این هست که آیا آدرس ایمیل معتبر هست یا نه

تابع SortContact برای مرتب کردن مخاطبین در دیتابیس برنامه ست ، که بر اساس مقداری که میگیره (اسم باشه یا اسم خانوادگی) اونا رو مرتب میکنه و بر میگردونه که در تابع BtnSortFirstName و BtnSortLastName ازش استفاده شده بود

تابع SaveToFile برای ذخیره کردن دیتابیس به فایل txt یا json یا xml هست ، وقتی که دکمه ی save در سربرگ اول زده میشه ، یک پنجره باز میشه و میپرسه که با چه فرمتی ذخیره بشه ، و انتخاب فرمت ، فایل با همون فرمت ذخیره میشه (کتابخونه های اول برنامه اینجا خیلی استفاده شدند مثل xml و json و ...)

تابع LoadToProgram برای باز کردن دیتابیس و وارد کردن آن به برنامه هست

**قسمت دوم برنامه که در if قرار داره :**

خط اول تا سوم و دو خط آخرش یعنی :

app = QtWidgets.QApplication*(*sys.argv*)*app.setStyle*(*QtWidgets.QStyleFactory.create*(*"Fusion"*))*MainWindow = QtWidgets.QMainWindow*()*

و

MainWindow.show*()*sys.exit*(*app.exec\_*())*

به صورت پیش فرض هستند و ماله خود pyqt5 هستند

سه خط دوم ، برای ساختن program هست و گرفتن یک آبجکت از کلاس **ContactsManager** هست ، و اجرای دو تابع از آن است

ui = ContactsManager*()*ui.setupUi*(*MainWindow*)*ui.Defaults*()*

10 خط بعدی ، برای وصل کردن pushbutton ها fه تابع های داخل کلاس هست که بالا توضیح دادم.

**فایل های داخل پوشه :**

فایل MainPage.py : همان کد های گرافیکی برای پنجره ی اصلی هست

فایل MainPage.ui : همان کد های گرافیکی برای پنجره هست که برای استفاده در نرم افزار designer هست

**فایل keep.txt :**

نیازی به فرستادن این فایل برای استاد نیست ، و برای ذخیره کردن کد هایی هست که برای تبدیل ui به py هست نوشته شده و یه سری توضیحات اضافی

**فایل mainCMS.exe :**

نیازی به فرستادن این فایل برای استاد نیست