Project deliverable description.

We have 2 non trivial insert statements. Each for Scheduling and Patient account creating. Then we have one delete statement for delete schedule and one update statement for patient.
For joining we have several statements, but characteristic statements are for - view the appointment of doctor whom the nurse work with.
Since nurses themselves, don't have any appointment and only have working with doctor, it needs join the tables.
-view the appointment details and doctor's name, then we need to join patient, doctor, appointment.
-to view the scheduled time we need to join staff, scheduled time, weekly schedule.
And For patient and Doctor, when we see the prescription detail we join 4 tables.

For Aggregation, we make it for showing the statistical info about medicine doctor prescribe to patient. Medicine have ingredient and brand information, so we order the number of medicine in prescription by brand and ingredients.

For Create view we make view for emergeny available doctor info for patient. We use create or replace view since, we will update every time patient click the button, but if we use create view, that name of view might be already exist.

For other selection query, we use it for view the informations of each users.

URL of page is
127.0.0.1:8000/USERTYPE/ID/detail/
And in detail page, it redirects to our several functionality pages.
For schedule
127.0.0.1:8000/schedule/ScheduleID/list/

We change our formal specification, include few things below.
- **All insert,update,delete about appointment is not included even if we could view appointment**
- **All insert,update,delte about treats(prescription) is not included even if we have view of prescription details.**
- **Staff will not update/insert/delete themselves, it will be only done by admin**

We change our create table statement slightly, since we missed some restrictions.
- **Add on delete cascade for contains, books, treats (relational) tables.**
- **Appointment's primary key is now single primary key.**

We change our ER diagram one thing. Delete attribute of appointment-make, "symptoms".

Our sql files are included in
dashospital-create-table
dashospital-insert-row
Based on two above files, the final status of localhost db of ourproject is in **final.sql**
(It includes status after we do insert, delete, update in our project, so data is little bit different than our initial data by insert-row)
sql-dml (include query inside of our source code)

Source code basically focus on the 3 apps Staff, WeeklySchedule, Patient
views.py in three folders contains queries.
template folder contains all html file we use.