**Chapter 1: Introduction**

Name of Project: Arbitrage

Goal of Project: to compare the price of the coins that is common in two Iranian cryptocurrency broker and show the difference between two prices.

**Chapter 2: Prerequisites**

I do this project with Django framework because it based on python.

for beginning I review nobitex API documentation at <https://apidocs.nobitex.ir/> and wallex API documentation at <https://api-docs.wallex.ir/> .

now, I create a database with PostgerSQL. For doing this I create new database that name *crypto\_prices* with enter psql in windows shell and then choosing server, password, username, and db name. after it I I’m going to *pgadmin* and on query tool, create a table that named *prices* with this command:

CREATE TABLE prices (

id SERIAL PRIMARY KEY,

currencies VARCHAR(255),

binance\_last\_trade\_price FLOAT(24),

nobitex\_last\_trade\_price FLOAT(24),

wallex\_last\_trade\_price FLOAT(24)

);

SELECT \* FROM prices;

after that I create a Django project with this command :

django-admin startproject Arbitrage

after that I create an app with this:

python manage.py startapp crypto

then, it’s time for config the settings:

1. in settings.py :

add ‘crypto’ to INSTALLED\_APP

add 'DIRS': [os.path.join(BASE\_DIR, 'templates')] to TEMPLATES

add database info to DATABASES

add STATIC\_URL = 'static/'

1. create crypto/urls.py
2. add path('', include('crypto.urls')) to urlpatterns in arbitrage/urls.py
3. create crypto/templates and create home.html in it.
4. Create crypto/static
5. Set home template to urls.py

Next thing is to create admin by py manage.py createsuperuser

Alright, it’s done and now just to more commands:

py .\manage.py makemigrations

py .\manage.py migrate

and finally run it by: py .\manage.py runserver

**Chapter 3: MVTU**

Now we have everything and now I can go to main job!

**Model:** define database values and table name in a class (name= Prices)

In this section I have a challenge that django wanna to manipulate my DB, for this I set managed= false

**View:** in this section I define function *home* that receives data from DB and returning them. Although I create three url to use it on next version( I want to link every price of broker)

Also I create a function *update\_nobitex\_data* to use another python files(include class and functions) separately.

For doing this I create a directory: *crypto/management\commands* and create a *\_\_init\_\_.py* in both *management* and *commands* directory. Also create *update\_nobitex\_data.py* in *commands* for do my purpose.(explanation later)

**Template:** set “for” loop to show the table

**OTHER:** . Set prices in admin(for managing the table)