Kako je napravljeno?

Po preporuci za WebService uzeo sam Javu, odnosno Spring framework.

Iako nisam nešto iskusan u javi, uz proslijeđeni toutorial nije bilo strašno. Priznajem da ga nisam cijelog pogledao pa vjerojatno ima razlika i isto nije napravljeno po principima frameworka, ali meni je bilo važno pokriti funkcionalnosti zadane u zadatku.

1. WebService

Dakle, servis po sebi ima dvije važne komponente:

1.) API – omogućiti REST API i reagirati na njegove metode
2.) Background task – dohvaćanje podataka s Kraken API-a.

Unutar main funkcije webservisa pokreće se Spring aplikacija(po defaultu), te je ondje dodana još jedna linija koja će pokrenuti komponentu pod 2..

BackgroundScheduledJob - kreiran je kao Service Class, koju čine dvije metode te jedan privatni član.

- periodInMinutes – opisuje koliko često će se callback metoda izvršavati
- Start() – Metoda koja pokreće callback thread
- MainCallback() – Metoda koja se izvršava svakih „periodInMinutes“ minuta

Kraken

Unutar MainCallback() metode koristi se Kraken servis, koji je također definiran jednom Service Classom

- SourceAPIBaseURL – string koja je osnova kraken API URL-a
- PariNames – lista stringova koja sadrži moguće inpute za „PairName“ API key
- GetTickerInformation() – metoda koja za svaki mogući „PariName“ dohvaća podatke pomoću RestAPIWrapper.GET metode sa zadanog URL-a. Kreira novi objekt tipa Ticker, pridružuje vrijendosti za CureationTime, Data i PairName. U slučaju pogreške, ispisuje log.

RestAPIWrapper

Wrapper koji u ovom primjeru sadrži samo GET metodu(samo je ona potrebna).

- GET(string url) – metoda koja kreira GET request na zadani URL, primljeni response parsira u JSON Objekt koji vraća kao povratnu vrijednost. U slučaju pogreške, ispisuje log.

MemoryDataBaseService

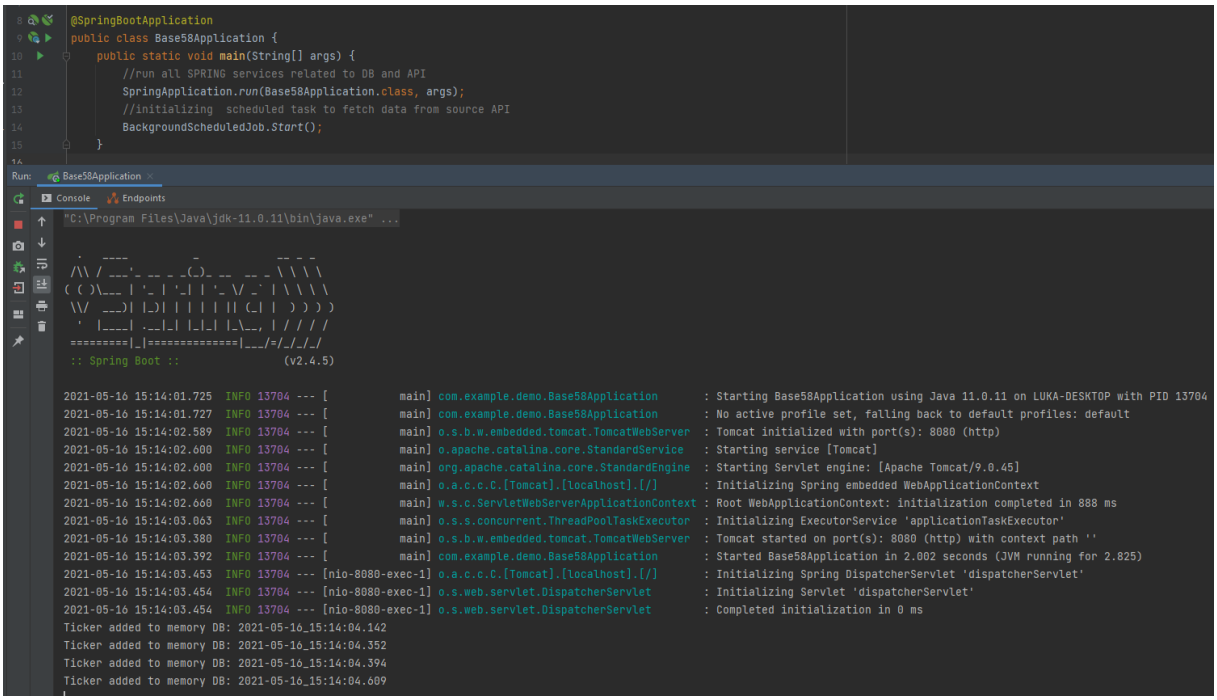
Za početak(kasnije je prebačeno na postgres) DB je definirana u memoriji.

- DataBase – statički objekt tipa DataBase
- Insert(Ticker ticker) – dodaje Ticker u DB
- Select() – vraća listu svih Tickera u bazi
- Select(String pairName) - vraća listu svih Tickera u bazi s konkretnim pariNameom
- Select(String date, String pariName) - vraća listu svih Tickera u bazi s konkretnim pariNameom koji je nastao u zadanom vremenu – date.

Ticker – model podatka koji se provlači kroz cijelu aplikaciju

DataBase – primjer baze podataka u RAM-u, definirana kao lista objekata tipa Ticker.

TickerController – klasa unutar koje su definirani mogući API zahtjevi

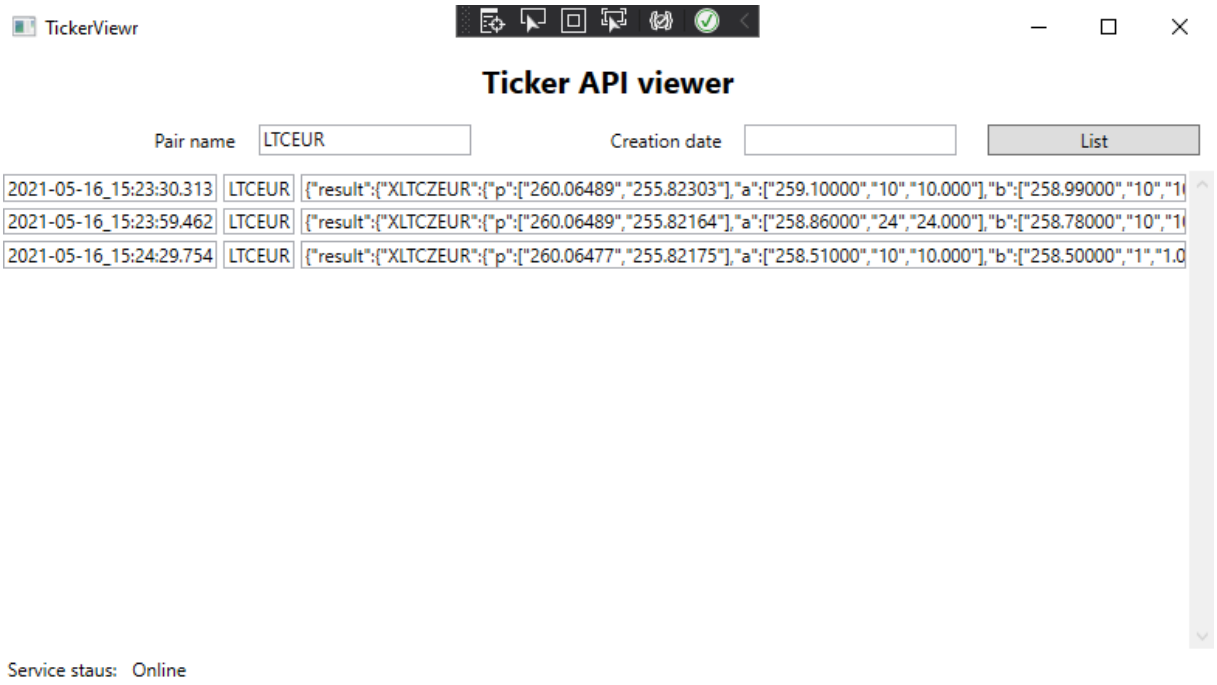


2. Consumer App

C# WPF aplikacija, kreiran po MVVM principu s popratnim Service i Common projektom.

Dizajniran je jednostavan prozor koji po pokretanju u donjem lijevom uglu pokazuje vidi li aplikacija webservise – ova se provjera događa u pozadini konstanto tako.

Dodano je i nekoliko kontrola kojima je moguće testirati API „(localhost:8080)/api/v1/ticker“



Kako pokrenuti?

Koliko sam shvatio spring framework što se tiče web servisa podigne sve sam čim se projekat pokrene unutar IDEA, pa je za sam servis dovoljno pokrenuti projekt.

*Baza podataka je podignuta pomoć dockera localhost:5432

Na strani Consumer App, ukoliko i nije instaliran MS Visaul Studio, taj folder je pushan bez .gitignore filea, pa su i binary datoteke ondje, koje je dovoljno samo pokrenuti.

..\ConsumerApp\ConsumerApp\bin\Debug\ConsumerApp.exe