

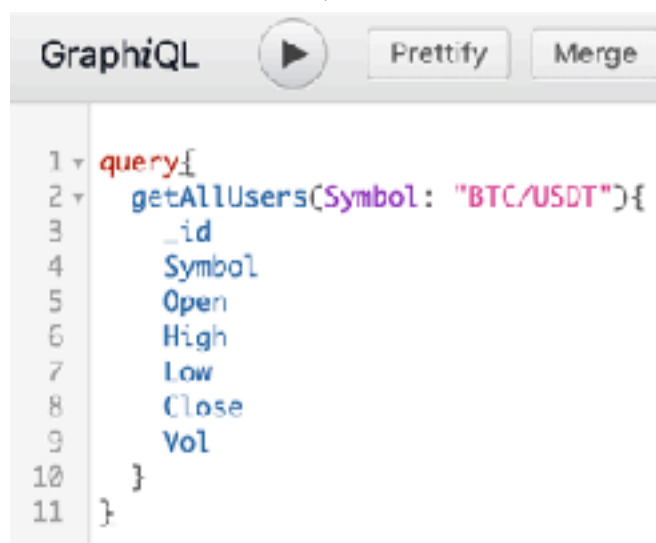
## پروژک اول

کدهای 1, 2, 3, 4 مربوط به این قسمت می‌باشند. داده‌ها با استفاده از پکیج cctx دریافت شده (کد 1) و در یک دیتابیس mongodb atlas (کد 3) ذخیره شده‌اند. اکانت ساخته شده در mongodb atlas دارای 512 مگ حجم رایگان برای ذخیره اطلاعات است. نحوه‌ی ذخیره‌ی داده‌های هر روز همانند شکل الف می‌باشد که “\_id” معادل همان تاریخ روز مورد نظر به صورت Unix timestamp است. همچنین شکل ب، query استفاده شده در graphql برای دریافت داده‌های “BTC/USDT” را نشان می‌دهد (کد 4).

شکل الف

```
_id: 1653782400000
Symbol: "BTC/USDT"
Open: 29031.33
High: 29587.78
Low: 28839.21
Close: 29468.1
Vol: 27567.34764
```

شکل ب



```
1 query{
2   getAllUsers(Symbol: "BTC/USDT"){
3     _id
4     Symbol
5     Open
6     High
7     Low
8     Close
9     Vol
10  }
11 }
```

## پروژک دوم

کد شماره‌ی ۵ مربوط به این پروژه است. در پوشه‌ی تست فایل‌های signals.json و decisions.json قرار دارند که برای امتحان تابع تشکیل شده‌اند. خروجی تابع برای رده بندی مثال، به صورت زیر می‌باشد.

Decisions:

```
[
  { analyst: 'ali', signal: 1, analyst_decision: 'confirm' },
  { analyst: 'ali', signal: 2, analyst_decision: 'reject' },
  { analyst: 'ali', signal: 3, analyst_decision: 'confirm' },
  { analyst: 'erfan', signal: 1, analyst_decision: 'reject' },
  { analyst: 'erfan', signal: 2, analyst_decision: 'confirm' },
  { analyst: 'erfan', signal: 3, analyst_decision: 'reject' },
  { analyst: 'shay', signal: 1, analyst_decision: 'confirm' },
  { analyst: 'shay', signal: 2, analyst_decision: 'confirm' }
]
```

Signals:

```
[
  { signal_number: 1, status: 'stop' },
  { signal_number: 2, status: 'target' },
  { signal_number: 3, status: 'open' }
]
```

Analysts List:

```
[
  { analyst: 'erfan', correct_decisions: 2, total_decisions: 3 },
  { analyst: 'shay', correct_decisions: 1, total_decisions: 2 },
  { analyst: 'ali', correct_decisions: 0, total_decisions: 3 }
]
```

## پروژک سوم

۱-

الف) برای رتبه بندی آنالیزها ابتدا یک کالکشن خالی با نام list تشکیل می‌شود که آنالیزها یک به یک در آن قرار خواهند گرفت. این تابع به ترتیب هر یک از decision ها (در کالکشن decisions) را بررسی می‌کند، به این صورت که ابتدا با استفاده از داده‌ی signal به دنبال یک داکيومنت با signal\_number مشابه در کالکشن signals می‌گردد. پس از پیدا کردن این داکيومنت با استفاده از داده‌های status و analyst\_decision بررسی می‌شود که تصمیم گرفته شده توسط آنالیز درست بوده یا نه (که یک عدد صفر (غلط) یا یک (درست) خواهد بود). حال بررسی می‌شود که اسم آنالیز تصمیم گیرنده در list قرار دارد یا خیر. اگر قرار نداشته باشد، یک داکيومنت جدید با این نام و نتیجه‌ی تصمیم ساخته می‌شود. در صورتی که این نام از قبل در لیست موجود باشد، تعداد تصمیم‌ها یک عدد افزایش یافته و با توجه به درست یا غلط بودن تصمیم، تعداد تصمیم‌های درست این آنالیز، بروز می‌شود. در نهایت لیست نهایی از رتبه‌بندی آنالیزها با توجه به کل تصمیم‌های درست تشکیل می‌شود.

ب) همچنین شرایط زیر برای عمل کردن این تابع مورد نیاز است: داکيومنت‌های ناقص در کالکشن‌ها نباشند. داکيومنت‌های تکراری وجود نداشته باشند. سیگنال‌های تمام داکيومنت‌های کالکشن decisions، در کالکشن signals موجود باشند. Status سیگنال‌ها خارج از سه حالت open target و stop نباشد. analyst\_decision تصمیم‌ها خارج از دو حالت reject و confirm نباشد.

پ) برای تست کردن تابع باید از کالکشن‌هایی استفاده شود که هم‌ه‌ی انواع متغیرها با تکرارهای غیریکسان در آن حضور داشته باشند. (مثال بررسی شده در پروژک دوم)

۲- یک تست در بخش پروژک دوم توضیح داده شده که داده‌های آن در پوشه‌ی test(5-analyst-func) قرار داد.