



بخش اول (ذخیره اطلاعات بدون ساختار / کار با MongoDB

در بخش اول تمرین، برای ذخیره اطلاعات کاربران یک سایت فرضی از مانگو استفاده می‌کنیم و بعد از ذخیره اطلاعات، با انجام چند پرس‌وجوی ساده، نحوه کار با این دیتابیس محبوب را فرا خواهیم گرفت.

توضیح مورد کاربری مساله :

فرض کنید قصد طراحی سایتی برای آموزش مجازی و خرید و فروش دوره‌ها را دارید. در این سایت، افراد می‌توانند در دوره‌های مختلف ثبت نام کرده و با پرداخت هزینه، دوره‌های آموزشی را بگذرانند. مدرسان هم می‌توانند دروس مختلف را تعریف و محتوای آموزشی آنها را بارگذاری کنند. این بخش از سامانه، با دیتابیس‌های رابطه‌ای مدیریت خواهد شد.

اما از آنجا که مدرسان سایت، رزومه‌های مختلفی خواهند داشت (دانشگاه‌های مختلف، مقاطع و رشته‌های مختلف، مدارک غیر دانشگاهی، جوایز، کنفرانس‌ها، مسوولیت‌های اجرای و ...)، می‌توان کل اطلاعات سوابق کاربران را به صورت json و در دیتابیس مانگودی‌بی نگهداری کرد که انعطاف پذیری لازم برای ذخیره داده‌های بدون ساختار و بسیار متغیر را داشته باشد.

برای برقراری ارتباط بین دو دیتابیس رابطه‌ای و مانگودی‌بی، کلید جدول یوزر در دیتابیس رابطه‌ای (یا یک فیلد منحصر بفرد دیگر مانند ایمیل یا کد ملی)، باید در کالکشن متناظر در مانگودی‌بی ذخیره و ایندکس شود.

در این تمرین، فقط به ذخیره اطلاعات صد هزار کاربر در قالب جی‌سان در مانگودی‌بی و کار با این داده‌ها می‌پردازیم.

دریافت اطلاعات

برای ایجاد صد هزار کاربر از امکانات ایجاد کاربر تصادفی سایت randomuser.me استفاده می‌کنیم. این سایت که ملیت ایرانی را هم پشتیبانی می‌کند، از طریق API، یک تا پنج هزار کاربر تصادفی با ملیت و مشخصات داده شده تولید کرده، به ما در قالب جی‌سان برمی‌گرداند. نمونه‌ای از داده‌های تولید شده توسط این سایت را در زیر مشاهده می‌کنید:

```
{
  "results": [
    {
      "gender": "female",
      "name": {
        "title": "Ms",
        "first": "نیایش",
        "last": "نژاد سلطانی"
      },
      "location": {
        "street": {
          "number": 8962,
```



```
    "name": "استقلال میدان",
  },
  "city": "یزد",
  "state": "تهران",
  "country": "Iran",
  "postcode": 59373,
  "coordinates": {
    "latitude": "7.5752",
    "longitude": "162.0874"
  },
  "timezone": {
    "offset": "-5:00",
    "description": "Eastern Time (US & Canada), Bogota, Lima"
  }
},
"email": "nyysh.sltnynjd@example.com",
"login": {
  "uuid": "c5e3eaf3-80e9-4693-8eb6-aa3c64c04480",
  "username": "happyostrich714",
  "password": "qwert1",
  "salt": "b5JyKob5",
  "md5": "a240f2718337bc2cfee83b01b8b492d4",
  "sha1": "0d2a39cb9bbec1893731f529875a9809c7d6478f",
  "sha256": "20a123ea18446e5a515d406df7ad5931d7af4d82a31a30d3dffd096d536c914a"
},
"dob": {
  "date": "1955-12-06T15:34:18.796Z",
  "age": 65
},
"registered": {
  "date": "2015-11-17T00:53:50.137Z",
  "age": 5
},
"phone": "037-02250625",
"cell": "0932-136-2921",
"id": {
  "name": "",
  "value": "None"
},
"picture": {
  "large": "https://randomuser.me/api/portraits/women/41.jpg",
  "medium": "https://randomuser.me/api/portraits/med/women/41.jpg",
  "thumbnail": "https://randomuser.me/api/portraits/thumb/women/41.jpg"
},
"nat": "IR"
```



```
}  
],  
"info": {  
  "seed": "b4fc8f16a480868c",  
  "results": 1,  
  "page": 1,  
  "version": "1.3"  
}  
}
```

کافی است همین جی‌سان دریافت شده را در مانگو دی بی ذخیره کنید (البته اطلاعات موجود در فیلد results و آن هم به ازای هر کاربر در یک سند جداگانه ذخیره خواهد شد).

برای دریافت اطلاعات کاربران (که برای این تمرین حداقل نیاز به صد هزار کاربر خواهیم داشت)، می‌توانید از کد پایتون زیر استفاده کنید (البته کد را باید اصلاح کنید که این تعداد کاربر دریافت و در مانگو ذخیره شود):

```
import requests, json  
  
url = "https://randomuser.me/api/"  
  
querystring = {"nat": "ir", "results": "5000"}  
  
response = requests.request("GET", url, params=querystring)  
data = json.loads(response.text)  
print(data)
```

دانلود مانگودی بی و ساخت کالکشن کاربران

مانگو دی بی را نصب کرده¹ و کالکشن users را درون دیتابیس metadata (این دیتابیس هم باید ایجاد شود) بسازید. می‌توانید از خط فرمان مانگودی بی یا ابزارهای گرافیکی رایج مانند MongoDB Compass یا Robo 3T² برای این منظور استفاده کنید.

علاوه بر کتاب درسی معرفی شده، کتاب کوچک³ The Little MongoDB می‌تواند راهنمای سریع شما برای کار با مانگو در این تمرین باشد.

¹ <https://bit.ly/2XWSqM7>

² <https://robomongo.org/>

³ <https://openmymind.net/mongodb.pdf>



گام اول تمرین

در گام اول، با فراخوانی آدرس <https://randomuser.me/api/?nat=ir> اطلاعات چندین کاربر تصادفی را دریافت و در مانگودی بی به صورت دستی ذخیره کنید و بررسی کنید چه فیلدهایی توسط خود مانگو به صورت خودکار، به داده‌ها افزوده می‌شود.

سپس با استفاده از کتابخانه `pymongo`⁴ کدهای دریافت اطلاعات فوق را به گونه‌ای تغییر دهید که همزمان با دریافت اطلاعات کاربران، آنها را در مانگو هم ذخیره کنید.

با دستور `count`، مطمئن شوید که صد هزار داده، دریافت شده باشند.

خروجی گام اول

نحوه ورود دستی داده‌ها در مانگو، کدهای نوشته شده برای درج اطلاعات و نحوه اطمینان از درج صد هزار سند در گزارش آورده شود.

گام دوم - دستورات اصلی

در گام دوم به دستورات پایه مانگودی بی می‌پردازیم و به سوالات زیر پاسخ می‌دهیم:

1. نام و نام خانوادگی کاربرانی را پیدا کنید که بالای ۵۰ سال سن داشته و ساکن نیشابور باشند.
2. قصد ارسال هدیه به کاربرانی داریم که بیش از بیست سال است که در سایت ما ثبت نام کرده اند. نام خانوادگی، آدرس و موبایل این کاربران را بیابید.
3. تاریخ ثبت شده برای تولد (`dob`) و زمان ثبت نام به میلادی است. می‌خواهیم در کنار این تاریخ‌ها، یک فیلد جدید اضافه کنیم با نام `year_persian` که سال متناظر تقویم فارسی تولد و ثبت نام را هم داشته باشیم. از دستور `update` و یک فرمول ساده تبدیل سال میلادی به شمسی، استفاده کرده، این تغییر را اعمال کنید.
4. قصد داریم در پایان هر روز، به تمام افرادی که امروز، روز تولدشان است، یک ایمیل ارسال کنیم و کد تخفیفی برای آنها بفرستیم. دستوری بنویسید که با اجرا شدن آن در هر روز، نام و نام خانوادگی و ایمیل آن اشخاص به ما برگردانده شود.
5. ذخیره پسورد به شکل خام در اطلاعات کاربر، کاری غیرحرفه‌ای است. می‌خواهیم این مشکل را برطرف کنیم. چه راه حلی برای حل مساله پیشنهاد می‌کنید؟ راه حل را ابتدا روی یک سند خاص، امتحان کنید و مطمئن شوید بعد از اعمال تغییرات، آن کاربر خاص را با داشتن یوزنیم و پسورد، می‌توانید پیدا کنید. سپس تمام کاربران را به روز رسانی کنید.

⁴ <https://pymongo.readthedocs.io/en/stable/tutorial.html>



خروجی گام دوم

دستورات نوشته شده ، خروجی و زمان اجرای هر کوثری

گام سوم - دستورات تجمعی و آماری (Aggregate Functions)

1. می‌خواهیم یک کمپین تبلیغاتی برای سایت ایجاد کنیم. برای این منظور باید کاربران را به سه گروه سنی تقسیم کنیم : نوجوانان / جوانان و افراد میانسال به بالا. دسته اول سنی کمتر از 16، دسته دوم سنی بین 16 تا 30 و دسته سوم، بالاتر از 30 سال خواهند داشت. دستوری بنویسید که تعداد هر گروه را برگرداند.
2. تعداد کاربران هراسان را به تفکیک لازم داریم. چگونه این اطلاعات را تولید می‌کنید ؟
3. برای تمام افرادی که آفست **timezone** آنها برابر +5:00 است، فیلد شماره موبایل (cell) را کلا حذف کنید. حال تعداد افرادی را بیابید که شماره موبایل ندارند.
4. میانگین سن کاربران استان تهران را با میانگین سن کاربران سایر استان‌ها مقایسه کنید.
5. کدام شهر بیشترین کاربر و کدام شهر، کمترین کاربر را دارد.

خروجی گام سوم

دستورات نوشته شده ، خروجی و زمان اجرای هر کوثری