**MongoDB: Exam - 28.03.2020**

## Database

We have a **database** with two referential collections **activities** and **users** that you need to **validate**. The **activities** collection requires the next fields:

* ’**userInfo** ’ with **reference** to the users collection;
* ’**balance** ’ with the **balance** of the user **account**;
* ’**isActive** ’ **boolean** value of the user status;
* ‘**tags**’ contains an **array** of **strings**;
* ‘**connections**’ with **array** of **references** to the **users** collection;

The **users** collection requires:

* ‘**name**’ with the user **name**;
* ‘**age**’ number **greater** or **equal** to **18**;
* ‘**gender**’ that could be **male** or **female**;
* ‘**company**’ the company name;
* ‘**contacts**’ that contain **nested** **object** with properties
  + ‘**email**’ – required;
  + ‘**phone**’ – optional;
  + ‘**address**’ – required;

|  |  |
| --- | --- |
| **Documents example:** | |
| > db.activities.findOne()  {  "\_id" : ObjectId("5e7a6dc6ae62e89d8d7b08cd"),  "userInfo" : 15,  "balance" : 3932.42,  "isActive" : false,  "tags" : [  "in",  "tempor",  "excepteur",  "labore",  "esse",  "minim",  "sint"  ],  "connections" : [  17,  16,  19  ]  } | > db.users.findOne()  {  "\_id" : 30,  "name" : "Barbra Reed",  "age" : 32,  "gender" : "female",  "company" : "GINKOGENE",  "contacts" : {  "email" : "barbrareed@ginkogene.com",  "phone" : "+00 (955) 413-3853",  "address" : "460 Estate Road, Cloverdale, South Dakota, 8817"  }  } |

## Solution:

|  |
| --- |
| **Insert your solution here** |
|  |

## Requests

Now we have to provide the next CRUD operations and aggregations:

* **Create** collections **activities** and **users**. Use the documents form the resourse files **activities.json** and **users.json;**

|  |
| --- |
| **Insert your query here** |
|  |

* Find the document that has **“balance”: “3551.18”** into the **activities** collection;

|  |
| --- |
| **Insert your query here** |
|  |

* **Find** the user with email **lulatorres@pharmacon.com** into the **users** collection;

|  |
| --- |
| **Insert your query here** |
|  |

* Find **all users** that have **balance** greater or equalto **1500** and are **activities**;

|  |
| --- |
| **Insert your query here** |
|  |

* We need **email** of one user with tag “**commodo**”.

|  |
| --- |
| **Insert your query here** |
|  |

* Filter the users **over** **30** years old and print only their **name**, **company** and **email**;

|  |
| --- |
| **Insert your query here** |
|  |

* Find user working for “**VIKON**” compony and check if there are any users that have saved him as a spetial connection into the “**connections**” field of **activities** collection;

|  |
| --- |
| **Insert your query here** |
|  |

* **Delete** from the **activities** collection all users with tag “**amet**” into the tags array.

|  |
| --- |
| **Insert your query here** |
|  |

* Find users with **balance** greater then **3000**, take their referential full **information** from **users** collection and add as “**userFullInfo**” field, at the end we need only the information from the fields: “**balance**”, “**isActive**” and “**userFullInfo**” in **descending** order.

|  |
| --- |
| **Insert your query here** |
|  |

* Give us your proposal of **index** to **increase** the efficient **execution** of the above queries. Choose only **one** field.

|  |
| --- |
| **Insert your suggestion here.** |
|  |