**Nama : Dina Ulitia Sinurat**

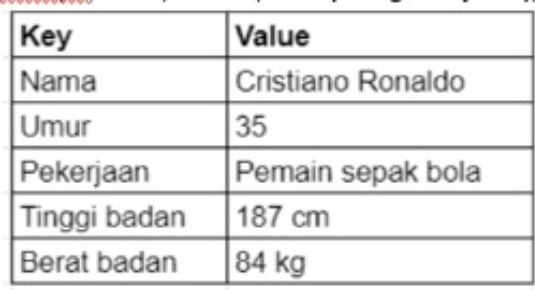
**Nim : 2213008**

**UAS Basdat 2**

1. Dalam NoSQL terdapat beberapa jenis, identifikasi dari jenis – jenis Database NoSQL tersebut kemudian jelaskan disetai dengan contoh “table”-nya

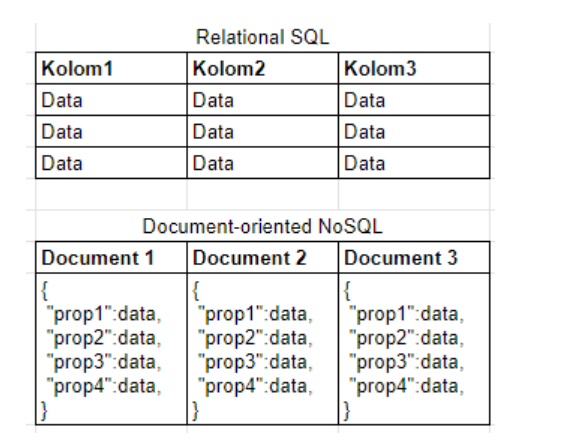
* **Key-value**

Database NoSQL jenis key-value menyimpan data dalam format pasangan kunci-nilai (key-value pair). Kunci adalah nilai unik yang digunakan untuk mengakses nilai. Nilai dapat berupa data apa pun, seperti string, angka, atau objek kompleks.



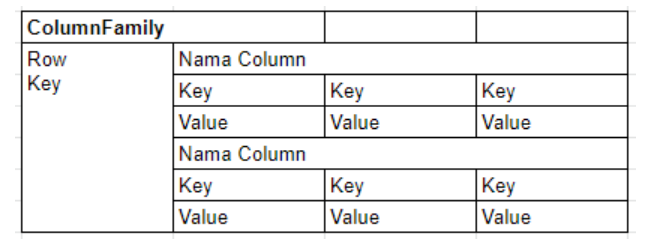
* **Document-oriented**

Database NoSQL jenis document-oriented menyimpan data dalam format dokumen. Dokumen adalah objek yang berisi data, seperti data teks, data numerik, dan data biner. Dokumen dapat memiliki struktur yang berbeda, tergantung pada kebutuhan aplikasi.



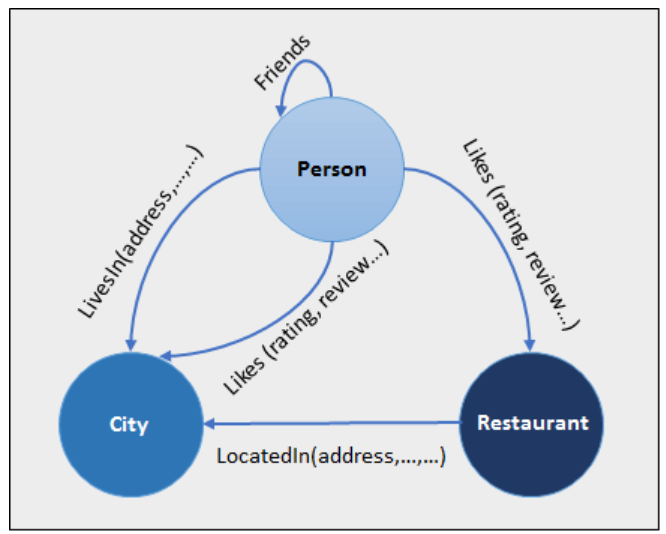
* **Column-oriented**

Database NoSQL jenis column-oriented menyimpan data dalam format kolom. Kolom adalah kumpulan data yang sejenis. Data dalam setiap kolom disimpan secara berdekatan, sehingga dapat diakses dengan cepat.



* **Graph**

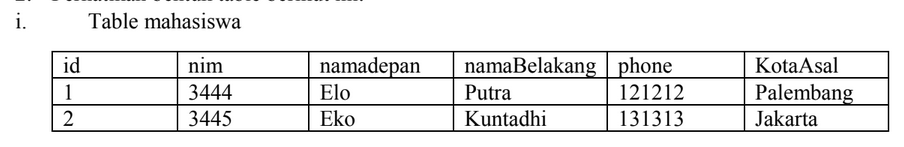
Database NoSQL jenis graph menyimpan data dalam format grafik. Grafik adalah struktur data yang terdiri dari node dan edge. Node mewakili entitas, seperti orang, tempat, atau objek. Edge mewakili hubungan antara node.



1. Perhatikan bentuk table berikut ini:

Silakan buat table dari database relational tersebut menjadi bentuk database NoSQL dengan

MongoDB YANG BENAR/tidak ada normalisasi tabel



**Perintah:**

db.Mahasiswa.insertMany([

{

id: 1,

nim: 3444,

namadepan: "Elo",

namaBelakang: "Putra",

phone: "121212",

kotaAsal: "Palembang"

},

{

id: 2,

nim: 3445,

namadepan: "Eko",

namaBelakang: "Kuntadhi",

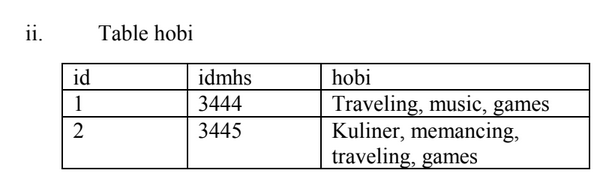
phone: "131313",

kotaAsal: "Jakarta"

}

]);

**Hasil :**



**Perintah :**

Db.Hobi.insertMany([

{

id: 1,

idmhs: 3444,

hobi: ["Traveling", "music", "games"]

},

{

id: 2,

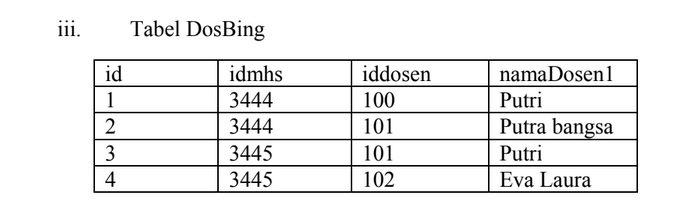
idmhs: 3445,

hobi: ["Kuliner", "memancing", "Traveling", "games"]

}

]);

**Hasil :**

****

**Perintah:**

Db.DosBing.insertMany([

{

id: 1,

idmhs: 3444,

iddosen: 100,

namaDosenl: "Putri"

},

{

id: 2,

idmhs: 3444,

iddosen: 101,

namaDosenl: "Putra bangsa"

},

{

id: 3,

idmhs: 3445,

iddosen: 101,

namaDosenl: "Putri"

},

{

id: 4,

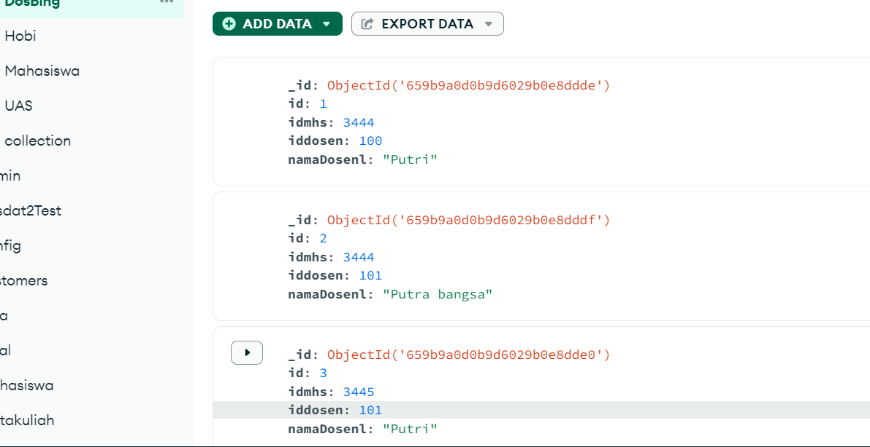
idmhs: 3445,

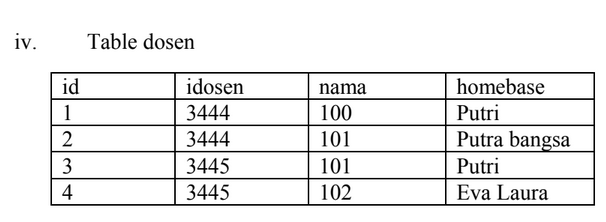
iddosen: 102,

namaDosenl: "Eva Laura"

}

]);

**Hasil:**

****

**Perintah :**

*db.Dosen.insertMany([*

*{*

*id: 1,*

*iddosen: 100,*

*nama: "Putri",*

*Homebase: "Informatika"*

*},*

*{*

*id: 2,*

*iddosen: 101,*

*namaDosenl: "Putri bangsa",*

*Homebase:"T.elektro"*

*},*

*{*

*id: 3,*

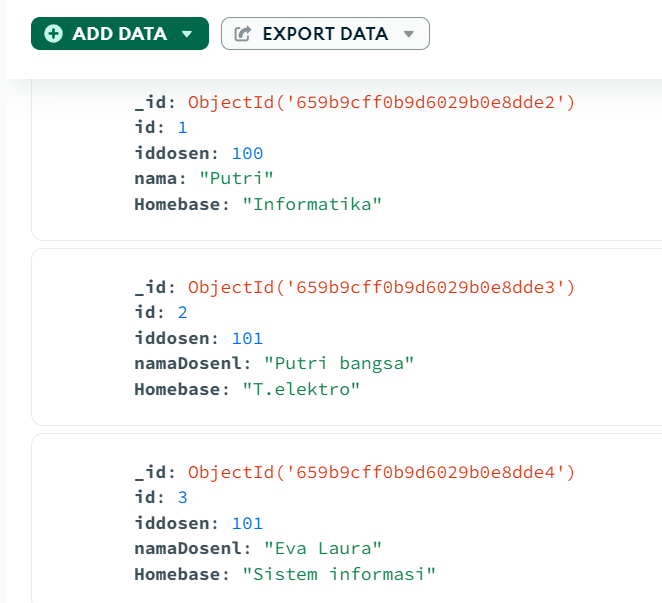
*iddosen: 101,*

*namaDosenl: "Eva Laura",*

*Homebase:"Sistem informasi"*

*}*

*]);*

**Hasil :**

1. Lakukan insert data JSON 1 dan JSON 2 ke dalam database relational (mySQL) dan MongoDB
2. **Json 1 MongoDb**

**Perintah:**

db.Json1.insertOne({

name: "John",

age: 30,

cars: [

{ name: "Ford", models: ["Fiesta", "Focus", "Mustang"] },

{ name: "BMW", models: ["320", "X3", "X5"] },

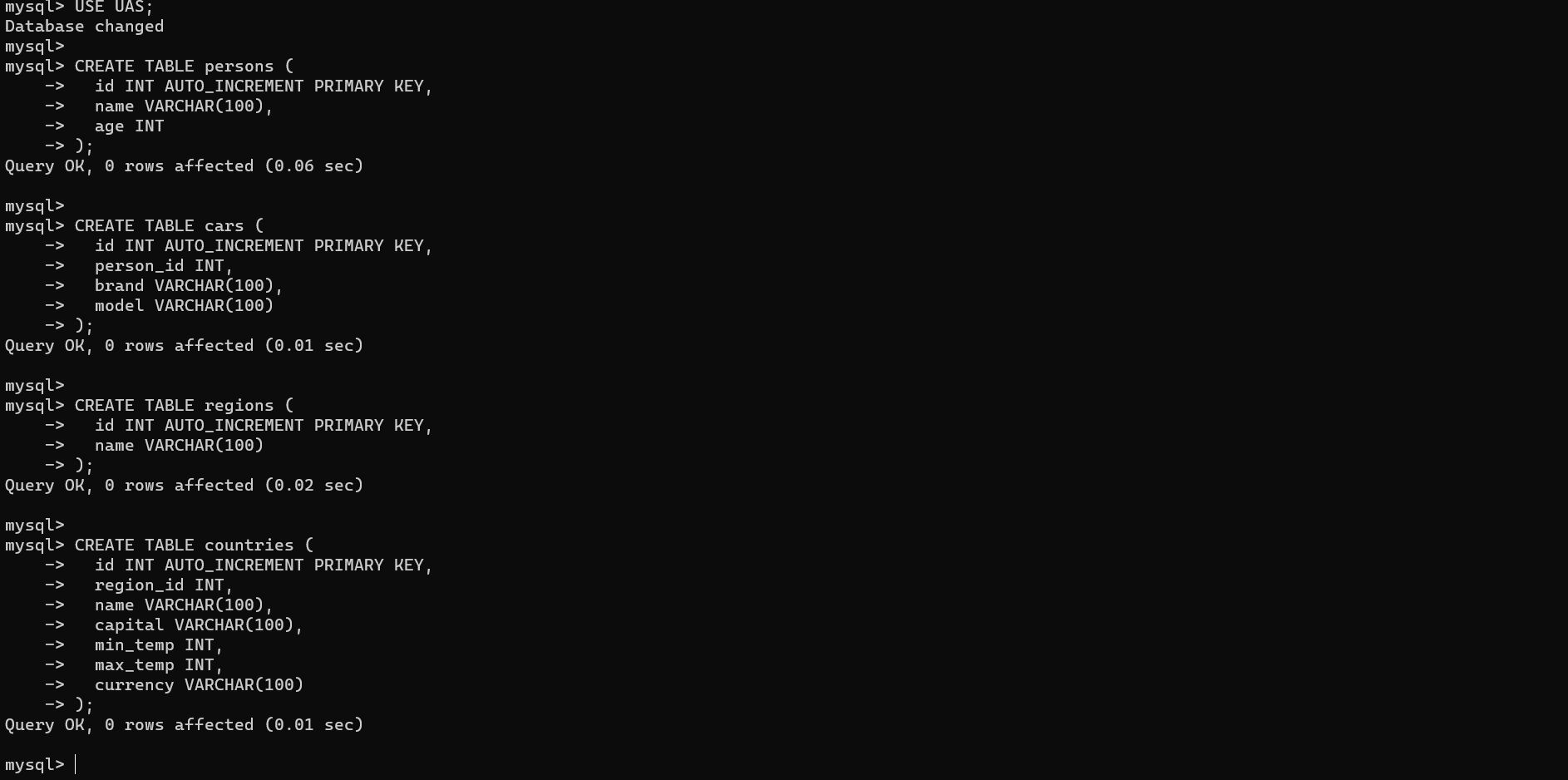
{ name: "Fiat", models: ["500", "Panda"] }

]

});

**Hasil:**

1. **Json 2 MySql**



1. **Json 2 MongoDb**

db.region.insertOne({

description: "Map containing Country, Capital, Currency, and some States of that Country",

name: "Asia",

countries: [

{

name: "India",

capital: "New Delhi",

minTemp: 6,

maxTemp: 45,

currency: "Rupee"

},

{

name: "Nepal",

capital: "Katmandu",

minTemp: 9,

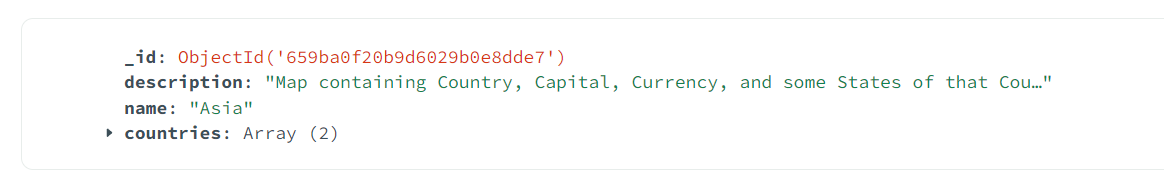
maxTemp: 23,

currency: "Nepalese rupee"

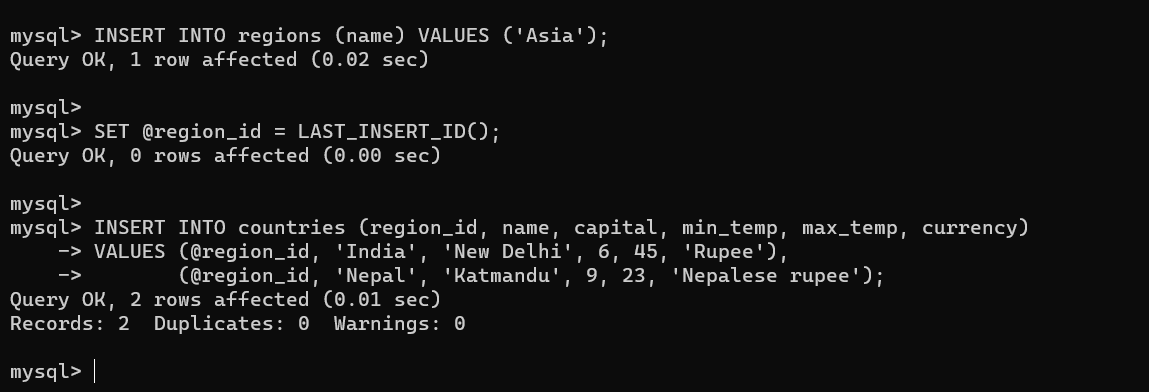
}

]

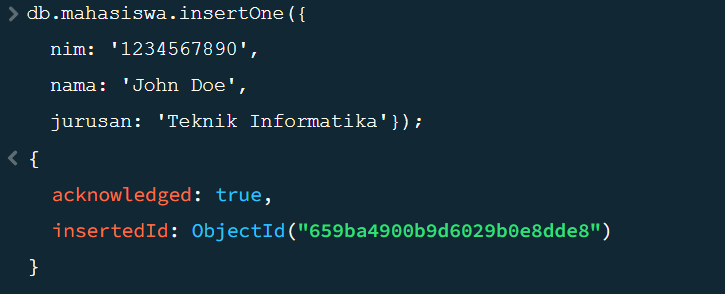
});

**Hasil:**

1. **Json 2 MySql**



1. Lakukaninsert one dan insert many dan update one dan update many(minimal 5 data) untuk table/collection dari soal nomor 2
2. insert one



1. insert many



1. Update one



1. Update many

