Fontello icomoon

newwaveapp123

idKlienta: 402745369476-embb3ijnrv3jqs58pfmm052gd1qps40u.apps.googleusercontent.com

kluczKlienta: z6xh1tBqRPcfZ0\_TXTVFwvHH

margin loga i telefonu

https://warszawa19115.pl/-/rejestracja-osoby-bezrobotnej

Pytania do mentora:

* CORS
* Funkcje callback
* Rozkminić this dlaczego czasem jest bind a funkcja strzałkowa.
* Jak to się dzieje że pod klientem pojawia się baza

Funkcje tablic:

<https://www.youtube.com/watch?v=H4awPsyugS0&list=PLRqwX-V7Uu6aAEUqu96Newc-7qpuh-cxc>

Notatka MongoDB

Wszystkie operatory: <https://docs.mongodb.com/manual/reference/operator/query/>

**CRUD – Create, Read, Update, Delete:**

- **Create** – metody to insertOne lub insertMany 🡪 składnia db.<collection-name>.<method>, gdzie w metodzie jako argument wpisujemy obiekt lub kolekcję obiektów w tablicy. Usunąć kolekcję można za pomocą metody drop() wykonanej na kolekcji

-**Read –** metody to findOne lub find 🡪 składnia db.<collection-name>.<method>, gdzie w metodzie jako argument wpisujemy warunek jako obiekt. Forma może być prosta lub z gt, lt, gte, lte, ne, nin, in, eq. In i Nin zestawiają wyszukiwane atrybuty w tablicach eg. {atrybut: {$in: [‘szukany element’] }}. Jest jeszcze $and i $or z składnią: db.persons.find({ $and: [{ age: { $gt: 18 } }, { age: { $lt: 40 }}] });

-**Update** – metody to updateOne i updateMany – pierwsza znajduje jeden/lub pierwszy element pasujący do warunku, a druga znajduje wszystkie pasujące do warunku 🡪 składnia db.<collection-name>.updateOne(<condition>, <data-to-update>), gdzie data to update ma składnię { $set: { <attrs-to-update>} }. Przykład: db.persons.updateMany({ salary: { $gt: 3000 }}, { $set: { salary: 2500 }});

- **Delete** – metody to deleteOne i deleteMany 🡪 składnia

db.<collection-name>.deleteOne(<condition>)

db.<collection-name>.deleteMany(<condition>)

Całość działa jak find.