

Stacja paliw

Opis

Sieci stacji paliw potrzebna jest baza danych do przechowywania informacji o sprzedaży, magazynowaniu i logistyce paliw naftowych, pracownikach, produktach sprzedaży detalicznej oraz nieruchomościach.

Nieruchomości muszą mieć określony typ: stacja lub magazyn, adres oraz kierownika. Pracownicy muszą mieć przypisane miejsce zatrudnienia oraz stanowisko. Każdy pracownik ma służbowy e-mail, który służy również jako identyfikator przy połączeniu z zewnętrznymi systemami. Pracownicy magazynów mogą tworzyć i wysyłać transporty. Pracownicy stacji paliw mogą zamawiać paliwa, zarządzać sprzedażą. Zamówienia muszą mieć określony towar oraz ilość, magazyn oraz stację dostawy oraz pracowników zamawiających i realizujących zamówienie. Nieruchomości muszą mieć określone stany magazynowe, zawierające towar oraz jego ilość. Na danych stacjach pracownicy muszą mieć możliwość określenia cen surowców.

Założenia bazy

Pracownicy

- muszą mieć określone imię i nazwisko
- muszą mieć określone stanowisko
- muszą mieć określony email służbowy
- muszą mieć określoną stację lub magazyn zatrudnienia

Pracownicy magazynów

- muszą mieć możliwość podglądu zamówień do realizacji
- muszą mieć możliwość zmiany statusu zamówienia na “w realizacji”

Kierowcy

- muszą mieć możliwość zmiany statusu zamówienia na “w dostawie”

Pracownicy stacji

- muszą mieć możliwość składania zamówień na konkretne towary
- muszą mieć możliwość zmiany statusu zamówienia na “zakończone”

Kierownicy stacji

- muszą mieć możliwość zmieniania cen towarów

Stacje

- muszą mieć określony typ: stacja lub magazyn

- muszą mieć określony adres
- muszą mieć określonego kierownika
- muszą mieć określone stany magazynowe konkretnych towarów
- muszą mieć określone ceny konkretnych towarów

Magazyny

- muszą mieć określony typ: stacja lub magazyn
- muszą mieć określony adres
- muszą mieć określonego kierownika
- muszą mieć określone stany magazynowe konkretnych towarów

Towary

- muszą mieć określone typy paliwa
- muszą mieć określone nazwy marketingowe

Zamówienia

- muszą mieć określony status
- muszą mieć określoną listę towarów
- muszą mieć określony magazyn realizujący zamówienie
- muszą mieć określoną stację docelową
- muszą mieć określonego pracownika realizującego zamówienie
- muszą mieć określonego pracownika składającego zamówienie
- muszą mieć określonego kierowcę przewożącego zamówienie

Kolekcje bazy danych

Nazwa kolekcji	gas_stations
Pola	
<ul style="list-style-type: none">● gas_stationID● products[]<ul style="list-style-type: none">○ type○ name○ stock○ price● address<ul style="list-style-type: none">○ region○ city○ street○ number○ zip_code○ country	

Nazwa kolekcji	warehouses
Pola	
<ul style="list-style-type: none">● warehousesID● products[]<ul style="list-style-type: none">○ type○ name○ stock● address<ul style="list-style-type: none">○ region○ city○ street○ number○ zip_code○ country	

Nazwa kolekcji	workers
Pola	
<ul style="list-style-type: none">● workesrsID● name● surname● email● place_of_employment<ul style="list-style-type: none">○ type○ id● job_position	

Nazwa kolekcji	orders
Pola	
<ul style="list-style-type: none"> ● ordersID ● products[] <ul style="list-style-type: none"> ○ type ○ name ○ stock ● origin <ul style="list-style-type: none"> ○ storagesID ○ address <ul style="list-style-type: none"> ■ region ■ city ■ street ■ number ■ zip_code ■ country ● target <ul style="list-style-type: none"> ○ gas_stationsID ○ address <ul style="list-style-type: none"> ■ region ■ city ■ street ■ number ■ zip_code ■ country ● order_owner <ul style="list-style-type: none"> ○ workersID ○ names ○ surname ○ email 	

Nazwa kolekcji	orders_details
Pola	
<ul style="list-style-type: none"> ● orders_detailsID ● order_id ● order_status ● drivers[] <ul style="list-style-type: none"> ○ workesrsID ○ name ○ surname ○ email ● storekeeper <ul style="list-style-type: none"> ○ workersID ○ name ○ surname ○ email 	

Zapytania do bazy danych

Stacje z niskim stanem magazynowym któregoś paliwa

`db.gas_stations.find({ "products": { $elemMatch: { stock:{$lte:3000}}}})`

```
> db.gas_stations.find({ "products": { $elemMatch: { stock:{$lte:3000}}}})
< { _id: ObjectId("624888f1f4fb08e41ea6e3a5"),
  products:
    [ { name: 'Super Diesel',
        type: 'Diesel',
        stock: Decimal128("3000"),
        price: Decimal128("6.20") },
      { name: 'Diesel+',
        type: 'Diesel',
        stock: Decimal128("5600"),
        price: Decimal128("7.70") },
      { name: 'Super 95',
        type: '95',
        stock: Decimal128("3000"),
        price: Decimal128("6.70") } ],
  address:
    { country: 'Poland',
      region: 'Wielkopolskie',
      city: 'Poznań',
      street: 'Warszawska',
      number: '34',
      zip_code: '24-123' } }
```

Stacje na ulicy Nadbystrzyckiej w Lublinie

`db.gas_stations.find({"address.city":'Lublin', "address.street":'Nadbystrzycka'})`

```
> db.gas_stations.find({"address.city":'Lublin', "address.street":'Nadbystrzycka'})
< { _id: ObjectId("624888f1f4fb08e41ea6e3a4"),
  products:
    [ { name: 'Super 95',
        type: '95',
        stock: Decimal128("4000"),
        price: Decimal128("5.70") },
      { name: 'Diesel+',
        type: 'Diesel',
        stock: Decimal128("6000"),
        price: Decimal128("6.62") } ],
  address:
    { country: 'Poland',
      region: 'Lubelskie',
      city: 'Lublin',
      street: 'Nadbystrzycka',
      number: '24',
      zip_code: '01-123' } }
```

pracownicy o nazwisku Kowalski

```
db.workers.find({surname:'Kowalski'})
```

```
> db.workers.find({surname:'Kowalski'})
< { _id: ObjectId("6248a4105307713668a0fa7f"),
  name: 'Jan',
  surname: 'Kowalski',
  email: 'jk@gmail.com',
  place_of_employment: { type: 'gas_station', id: '624888f1f4fb08e41ea6e3a3' },
  job_position: 'gas_station_worker' }
```

pracownik o najniższym ID

```
db.workers.find().sort({"_id":1}).limit(1)
```

```
> db.workers.find().sort({"_id":1}).limit(1)
< { _id: ObjectId("6248a4105307713668a0fa7f"),
  name: 'Jan',
  surname: 'Kowalski',
  email: 'jk@gmail.com',
  place_of_employment: { type: 'gas_station', id: '624888f1f4fb08e41ea6e3a3' },
  job_position: 'gas_station_worker' }
```

zamówienia w transporcie

```
db.orders_details.find({"order_status":'in_progress'})
```

```
> db.orders_details.find({"order_status":'in_progress'})
< { _id: ObjectId("6248abe75307713668a0fa9b"),
  order_id: '6248a8f35307713668a0fa8c',
  order_status: 'in_progress',
  drivers:
    [ { id: '6248a4105307713668a0fa80',
      name: 'Adam',
      surname: 'Nowak',
      email: 'an@gmail.com' } ],
  storekeeper:
    { id: '6248a4105307713668a0fa81',
      name: 'Michał',
      surname: 'Małysz',
      email: 'mm@gmail.com' } }
```

zamówienia pracownika Jan kowalski

`db.orders.find({"order_owner.name":'Jan',"order_owner.surname":'Kowalski'})`

```
> db.orders.find({"order_owner.name":'Jan',"order_owner.surname":'Kowalski'})
< { _id: ObjectId("6248a8f35307713668a0fa8c"),
  products:
    [ { name: 'Super Diesel',
        type: 'Diesel',
        stock: Decimal128("5000") },
      { name: 'Diesel+', type: 'Diesel', stock: Decimal128("4000") } ],
  origin:
    { id: '6248896ef4fb08e41ea6e3a8',
      address:
        { country: 'Poland',
          region: 'Mazowieckie',
          city: 'Warsaw',
          street: 'Marszałkowska',
          number: '51',
          zip_code: '00-123' } },
  target:
    { id: '624888f1f4fb08e41ea6e3a4',
      address:
        { country: 'Poland',
          region: 'Lubelskie',
          city: 'Lublin',
          street: 'Nadbystrzycka',
          number: '24',
          zip_code: '01-123' } },
  order_owner:
    { id: '6248a4105307713668a0fa7f',
      name: 'Jan',
      surname: 'Kowalski',
      email: 'jk@gmail.com' } }
{ _id: ObjectId("6248a8f35307713668a0fa8d"),
  products:
    [ { name: 'Super Diesel',
        type: 'Diesel',
        stock: Decimal128("3000") },
```

Stacje w województwie Mazowieckim

`db.gas_stations.find({"address.region":'Mazowieckie',"address.country":'Poland'})`

```

> db.gas_stations.find({"address.region":"Mazowieckie", "address.country":"Poland"})
< { _id: ObjectId("624888f1f4fb08e41ea6e3a3"),
  products:
    [ { name: 'Super Diesel',
        type: 'Diesel',
        stock: Decimal128("5000"),
        price: Decimal128("6.7000000000000000") },
      { name: 'Diesel+',
        type: 'Diesel',
        stock: Decimal128("4000"),
        price: Decimal128("6.70") } ],
  address:
    { country: 'Poland',
      region: 'Mazowieckie',
      city: 'Warsaw',
      street: 'Marszałkowska',
      number: '51',
      zip_code: '00-123' } }
stacja_paliw> db.gas_stations.find({"address.region":"Mazowieckie", "address.country":"Poland"})

```

suma paliw w zamówieniach

```

db.orders.aggregate([ { $group: { _id: "$_id", total: { "$sum": { "$sum": "$products.stock" } } } }, {
  $sort:{total: -1} } ])

```

```

> db.orders.aggregate([ { $group: { _id: "$_id", total: { "$sum": { "$sum": "$products.stock" } } } }, { $sort:{total: -1} } ])
< { _id: ObjectId("6248a8f35307713668a0fa8c"),
  total: Decimal128("9300") },
  { _id: ObjectId("6248a8f35307713668a0fa8d"),
  total: Decimal128("9000") },
  { _id: ObjectId("6248a8f35307713668a0fa8e"),
  total: Decimal128("8000") }
stacja_paliw>

```

łączna ilość paliw na stacjach

```

db.gas_stations.aggregate([ { $group: { _id: "$_id", total: { "$sum": { "$sum": "$products.stock" } } } }, {
  $sort:{total: -1} } ])

```

```

> db.gas_stations.aggregate([ { $group: { _id: "$_id", total: { "$sum": { "$sum": "$products.stock" } } } }, { $sort:{total: -1} } ])
< { _id: ObjectId("624888f1f4fb08e41ea6e3a5"),
  total: Decimal128("11600") },
  { _id: ObjectId("624888f1f4fb08e41ea6e3a4"),
  total: Decimal128("10000") },
  { _id: ObjectId("624888f1f4fb08e41ea6e3a3"),
  total: Decimal128("9200") }

```


liczba pracowników o danych nazwiskach

```
db.workers.aggregate([{$group : {_id: "$surname", liczba: {$sum : 1}}}]
```

```
> db.workers.aggregate([{$group : {_id: "$surname", liczba: {$sum : 1}}]])  
< { _id: 'Kowalski', liczba: 1 }  
  { _id: 'Nowak', liczba: 1 }  
  { _id: 'Małysz', liczba: 1 }
```

Uaktualnienia danych

Zmiana nazwiska pracownika

```
db.workers.update(  
  {"name": "Michał", "surname": "Małysz"},  
  {$set: {"surname": "Domański"}}  
)
```

Przed:

```
_id: ObjectId('6248a4105307713668a0fa81')  
name: "Michał"  
surname: "Małysz"  
email: "mm@gmail.com"  
place_of_employment: Object  
  type: "warehouse"  
  id: "6248896ef4fb08e41ea6e3a8"  
job_position: "storekeeper"
```

Po:

```
_id: ObjectId('6248a4105307713668a0fa81')  
name: "Michał"  
surname: "Domański"  
email: "mm@gmail.com"  
place_of_employment: Object  
  type: "warehouse"  
  id: "6248896ef4fb08e41ea6e3a8"  
job_position: "storekeeper"
```

Zmiana statusu zamówienia na closed

```
db.orders_details.update(  
  { "order_id": "6248a8f35307713668a0fa8c"},  
  {$set: { "order_status": "closed" }}  
)
```

Przed:

```
  _id: ObjectId('6248abe75307713668a0fa9b')  
  order_id: "6248a8f35307713668a0fa8c"  
  order_status: "in_progress"  
  ✓ drivers: Array  
    > 0: Object  
  ✓ storekeeper: Object  
    id: "6248a4105307713668a0fa81"  
    name: "Michał"  
    surname: "Małysz"  
    email: "mm@gmail.com"
```

Po:

```
  _id: ObjectId('6248abe75307713668a0fa9b')  
  order_id: "6248a8f35307713668a0fa8c"  
  order_status: "closed"  
  ✓ drivers: Array  
    > 0: Object  
  ✓ storekeeper: Object  
    id: "6248a4105307713668a0fa81"  
    name: "Michał"  
    surname: "Małysz"  
    email: "mm@gmail.com"
```

Zmiana maila pracownika

```
db.workers.update(  
  { "name": "Adam", "surname": "Nowak"},  
  {$set: { "email": "adamNowak@gmail.com" }}  
)
```

Przed:

```
  _id: ObjectId('6248a4105307713668a0fa80')  
  name: "Adam"  
  surname: "Nowak"  
  email: "an@gmail.com"  
  ✓ place_of_employment: Object  
    type: "warehouse"  
    id: "6248896ef4fb08e41ea6e3a8"  
  job_position: "driver"
```

Po:

```
  _id: ObjectId('6248a4105307713668a0fa80')  
  name: "Adam"  
  surname: "Nowak"  
  email: "adamNowak@gmail.com"  
  ✓ place_of_employment: Object  
    type: "warehouse"  
    id: "6248896ef4fb08e41ea6e3a8"  
  job_position: "driver"
```

Zmiana magazyniera w szczegółach zamówienia

```
db.orders_details.update(
  {"order_id": "6248a8f35307713668a0fa8c"},
  {$set: {"storekeeper": {"id": "6248a4105307713668a0fa7f", "name": "Jan", "surname": "Kowalski",
    "email": "jk@gmail.com"} }}
)
```

Przed:

```
_id: ObjectId('6248abe75307713668a0fa9b')
order_id: "6248a8f35307713668a0fa8c"
order_status: "closed"
▼ drivers: Array
  ▼ 0: Object
    id: "6248a4105307713668a0fa80"
    name: "Adam"
    surname: "Nowak"
    email: "an@gmail.com"
  ▼ storekeeper: Object
    id: "6248a4105307713668a0fa81"
    name: "Michał"
    surname: "Małysz"
    email: "mm@gmail.com"
```

Po:

```
_id: ObjectId('6248abe75307713668a0fa9b')
order_id: "6248a8f35307713668a0fa8c"
order_status: "closed"
▼ drivers: Array
  ▼ 0: Object
    id: "6248a4105307713668a0fa80"
    name: "Adam"
    surname: "Nowak"
    email: "an@gmail.com"
  ▼ storekeeper: Object
    id: "6248a4105307713668a0fa7f"
    name: "Jan"
    surname: "Kowalski"
    email: "jk@gmail.com"
```

Zmiana stanu magazynu

```
db.warehouses.update(
{"address.city":"Lublin"},
{$set: {"products.0.stock":0}}
)
```

Przed:

```
  _id: ObjectId('6248896ef4fb08e41ea6e3a9')
  products: Array
    0: Object
      name: "Super 95"
      type: "95"
      stock: 4000
    1: Object
  address: Object
    country: "Poland"
    region: "Lubelskie"
    city: "Lublin"
    street: "Nadbystrzycka"
    number: "24"
    zip_code: "01-123"
```

Po:

```
  _id: ObjectId('6248896ef4fb08e41ea6e3a9')
  products: Array
    0: Object
      name: "Super 95"
      type: "95"
      stock: 0
    1: Object
  address: Object
    country: "Poland"
    region: "Lubelskie"
    city: "Lublin"
    street: "Nadbystrzycka"
    number: "24"
    zip_code: "01-123"
```

Użytkownicy

Stworzenie admina

```
db.createUser(  
  {  
    user: "admin",  
    pwd: passwordPrompt(),  
    roles: [  
      { role: "userAdminAnyDatabase", db: "admin" },  
      { role: "readWriteAnyDatabase", db: "admin" }  
    ]  
  }  
)
```

```
> use admin  
< 'switched to db admin'  
> db.createUser(  
  {  
    user: "admin",  
    pwd: passwordPrompt(),  
    roles: [  
      { role: "userAdminAnyDatabase", db: "admin" },  
      { role: "readWriteAnyDatabase", db: "admin" }  
    ]  
  }  
)  
< { ok: 1 }
```

Stworzenie magazyniera

```
db.createUser(  
  {  
    user: "magazynier",  
    pwd: "haslo",  
    roles: [ { role: "readWrite", db: "stacja_paliw" } ]  
  }  
)
```

```
> db.createUser(  
  {  
    user: "magazynier",  
    pwd: "haslo",  
    roles: [ { role: "readWrite", db: "stacja_paliw" } ]  
  }  
)  
< { ok: 1 }
```

URI ⓘ

Edit Connection String ☒

```
mongodb://magazynier:haslo@localhost:27017/?  
authMechanism=DEFAULT&authSource=sacja_paliw
```

▼ Advanced Connection Options

General

Authentication

TLS/SSL

Proxy/SSH Tunnel

Advanced

Authentication Method

None

Username/Password

X.509

Kerberos

LDAP

AWS IAM

Username

magazynier

Password

.....

Authentication Database ⓘ

sacja_paliw

Optional

Authentication Mechanism

Default

SCRAM-SHA-1

SCRAM-SHA-256

Connect

Stworzenie konta dla spedycji

```
db.createUser(  
  {  
    user: "spedycja",  
    pwd: "haslo2",  
    roles: [ { role: "read", db: "stacja_paliw" } ]  
  }  
)
```

```
> db.createUser(  
  {  
    user: "spedycja",  
    pwd: "haslo2",  
    roles: [ { role: "read", db: "stacja_paliw" } ]  
  }  
)  
< { ok: 1 }
```

URI ⓘ

Edit Connection String ☒

```
mongodb://spedycja:haslo2@localhost:27017/?  
authMechanism=DEFAULT&authSource=sacja_paliw
```

▼ Advanced Connection Options

General

Authentication

TLS/SSL

Proxy/SSH Tunnel

Advanced

Authentication Method

None

Username/Password

X.509

Kerberos

LDAP

AWS IAM

Username

spedycja

Password

Authentication Database ⓘ

sacja_paliw

Optional

Authentication Mechanism

Default

SCRAM-SHA-1

SCRAM-SHA-256

Eksport/import kopii bazy danych

Export:

Export Collection stacja_paliw.gas_stations



☐ Export query with filters — 3 results (Recommended)

```
db.gas_stations.find(  
  {}  
)
```

☒ Export Full Collection

CANCEL

SELECT FIELDS

localhost:27017

4 DBS9 COLLECTIONS

☆ FAVORITE

HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 5.0.7 Community

{ } My Queries

☰ Databases

🔍 Filter your data

▶ admin

▶ config

▶ local

▼ stacja_paliw

📁 gas_stations...

📁 orders

📁 orders_details

📁 warehouses

📁 workers

Documents
stacja_paliw.gas_s...

+

stacja_paliw.gas_stations

DocumentsAggregationsSchemaExplain Plan

FILTER { }

Export Collection

ADD DATA

VIEW

{}[]

_id: ObjectId('624888f1f4fb08e41ea6e3a3')

> products: Array

> address: Object

_id: ObjectId('624888f1f4fb08e41ea6e3a4')

> products: Array

> address: Object

_id: ObjectId('624888f1f4fb08e41ea6e3a5')

> products: Array

> address: Object

Export Collection stacja_paliw.gas_stations



Select Export File Type

JSON

CSV

Output

C:\Users\Marcin\Desktop\export_gas_stations.json

 BROWSE

< BACK

CANCEL

EXPORT

Import:

Documents
stacja_paliw.gas_s...

+

stacja_paliw.gas_stations


Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' }

ADD DATA

VIEW

Import File
Insert Document



This collection has no data

It only takes a few seconds to import data from a JSON or CSV file

Import Data

Import To Collection stacja_paliw.gas_stations



Select File

 export_gas_stations.json

Select Input File Type

JSON

CSV

Options

☐ Stop on errors

CANCEL

IMPORT

stacja_paliw.gas_stations

Documents

Aggregations

Schema

Explain Plan

Indexes

Va

 FILTER { field: 'value' }

 ADD DATA



VIEW



```
_id: ObjectId('624888f1f4fb08e41ea6e3a3')  
> products: Array  
> address: Object
```

```
_id: ObjectId('624888f1f4fb08e41ea6e3a4')  
> products: Array  
> address: Object
```

```
_id: ObjectId('624888f1f4fb08e41ea6e3a5')  
> products: Array  
> address: Object
```

Zapytania w LiteDB

Wyszukanie dwóch pierwszych pracowników posortowanych alfabetycznie

SELECT \$ FROM workers ORDER BY surname LIMIT 2

```
SELECT $ FROM workers ORDER BY surname LIMIT 2
```

Grid	Text	Parameters
	<pre>/* 1 */ { "_id": {"\$oid": "6248a4105307713668a0fa7f"}, "name": "Jan", "surname": "Kowalski", "email": "jk@gmail.com", "place_of_employment": { "type": "gas_station", "id": "624888f1f4fb08e41ea6e3a3" }, "job_position": "gas_station_worker" } /* 2 */ { "_id": {"\$oid": "6248a4105307713668a0fa81"}, "name": "Michał", "surname": "Małysz", "email": "mm@gmail.com", "place_of_employment": { "type": "warehouse", "id": "6248896ef4fb08e41ea6e3a8" }, "job_position": "storekeeper" } }</pre>	

Wyszukanie produktów w magazynach w mieście Lublin

SELECT products FROM warehouses WHERE address.city='Lublin'

```
SELECT products FROM warehouses WHERE address.city='Lublin'
```

Grid
Text
Parameters

```
{
  "products":
  [
    {
      "name": "Super 95",
      "type": "95",
      "stock": {"$numberDecimal": "4000"}
    },
    {
      "name": "Diesel+",
      "type": "Diesel",
      "stock": {"$numberDecimal": "6000"}
    }
  ]
}
```

Wyszukanie zamówień z Polski do Polski

```
SELECT $ FROM orders WHERE target.address.country='Poland' AND origin.address.country='Poland'
```

Grid
Text
Parameters

	_id	products	origin	target	o
1	["\$oid":"6248a8f35307713668a0fa8c"]	[{"name":"Super Diesel","type":"Diesel","stock":{"\$numberDecimal":"500..."	["id":"6248896ef4fb08e41ea6e3a8","address":{"country":"Poland","region...	["id":"624888f14fb08e41ea6e3a4","address":{"country":"Poland","region...	["i
2	["\$oid":"6248a8f35307713668a0fa8d"]	[{"name":"Super Diesel","type":"Diesel","stock":{"\$numberDecimal":"300...	["id":"6248896ef4fb08e41ea6e3a8","address":{"country":"Poland","region...	["id":"624888f14fb08e41ea6e3a4","address":{"country":"Poland","region...	["i
3	["\$oid":"6248a8f35307713668a0fa8e"]	[{"name":"Super Diesel","type":"Diesel","stock":{"\$numberDecimal":"200...	["id":"6248896ef4fb08e41ea6e3a8","address":{"country":"Poland","region...	["id":"624888f14fb08e41ea6e3a4","address":{"country":"Poland","region...	["i

Wyszukanie id stacji gdzie jako pierwszy produkt jest Diesel

```
SELECT _id FROM gas_stations WHERE products[0].type='Diesel'
```

```
SELECT _id FROM gas_stations WHERE products[0].type='Diesel'|
```

Grid

Text

Parameters

```
/* 1 */
{
  "_id": {"$oid": "624888f1f4fb08e41ea6e3a3"}
}
/* 2 */
{
  "_id": {"$oid": "624888f1f4fb08e41ea6e3a5"}
}
```

Wyszukanie zamówień, gdzie występuje produkt w ilości większej lub równej 6000

SELECT \$ FROM orders WHERE products[*].stock ANY>=6000

SELECT \$ FROM orders WHERE products[*].stock ANY>=6000

Grid	Text	Parameters	
	_id	products	
	origin	target	
1	["\$oid":"624888f1f4fb08e41ea6e3a3"]	[{"name":"Super Diesel","type":"Diesel","stock":{"\$numberDecimal":"300..."}]	["id":"624888f1f4fb08e41ea6e3a4","address":{"country":"Poland","region..."}]
2	["\$oid":"624888f1f4fb08e41ea6e3a5"]	[{"name":"Super Diesel","type":"Diesel","stock":{"\$numberDecimal":"200..."}]	["id":"624888f1f4fb08e41ea6e3a8","address":{"country":"Poland","region..."}]

Zliczenie ilości pracowników z podziałem na miejsce pracy

SELECT @key AS place_of_employment_type, COUNT(*) FROM workers GROUP BY place_of_employment.type

```
SELECT @key AS place_of_employment_type, COUNT(*) FROM workers GROUP BY place_of_employment.type
```

Grid

Text

Parameters

```
/* 1 */
{
  "place_of_employment_type": "gas_station",
  "expr": 1
}
/* 2 */
{
  "place_of_employment_type": "warehouse",
  "expr": 2
}
|
```

Zliczenie stacji benzynowych w danym kraju oraz wyświetlanie produktów w tych krajach

```
SELECT @key AS country, COUNT(*) AS total, *.products AS products FROM gas_stations GROUP BY address.country
```

```
SELECT @key AS country, COUNT(*) AS total, *.products AS products FROM gas_stations GROUP BY address.country
```

Grid Text Parameters

```
{
  "country": "Poland",
  "total": 3,
  "products":
  [
    [
      {
        "name": "Super Diesel",
        "type": "Diesel",
        "stock": {"$numberDecimal": "5000"},
        "price": {"$numberDecimal": "6.70"}
      },
      {
        "name": "Diesel+",
        "type": "Diesel",
        "stock": {"$numberDecimal": "4000"},
        "price": {"$numberDecimal": "6.70"}
      }
    ],
    [
      {
        "name": "Super 95",
        "type": "95",
        "stock": {"$numberDecimal": "4000"},
        "price": {"$numberDecimal": "5.70"}
      },
      {
        "name": "Diesel+",
        "type": "Diesel",
        "stock": {"$numberDecimal": "6000"},
        "price": {"$numberDecimal": "6.62"}
      }
    ],
    [
      {
        "name": "Super Diesel",
        "type": "Diesel",
        "stock": {"$numberDecimal": "3000"},
        "price": {"$numberDecimal": "6.20"}
      },
      {
        "name": "Diesel+",
        "type": "Diesel",
        "stock": {"$numberDecimal": "5600"},
        "price": {"$numberDecimal": "7.70"}
      },
      {
        "name": "Super 95",
        "type": "95",
        "stock": {"$numberDecimal": "3000"},
        "price": {"$numberDecimal": "6.70"}
      }
    ]
  ]
}
```

Wyświetlanie średniej ceny produktów w Polsce z podziałem na miasta

```
SELECT @key AS city, AVG(*.products[*].price) AS average_of_products_price FROM gas_stations WHERE address.country="Poland" GROUP BY address.city
```

Grid	Text	Parameters
------	------	------------

Grid	Text	Parameters
------	------	------------

Zliczenie ilości produktów z podziałem na region

```
SELECT @key AS region, SUM(*.products[*].stock) FROM warehouses GROUP BY address.region
```

Grid

Text

Parameters

```
/* 1 */
{
  "region": "Lubelskie",
  "products": {"$numberDecimal": "10000"}
}
/* 2 */
{
  "region": "Mazowieckie",
  "products": {"$numberDecimal": "9000"}
}
/* 3 */
{
  "region": "Wielkopolskie",
  "products": {"$numberDecimal": "11600"}
}
```