

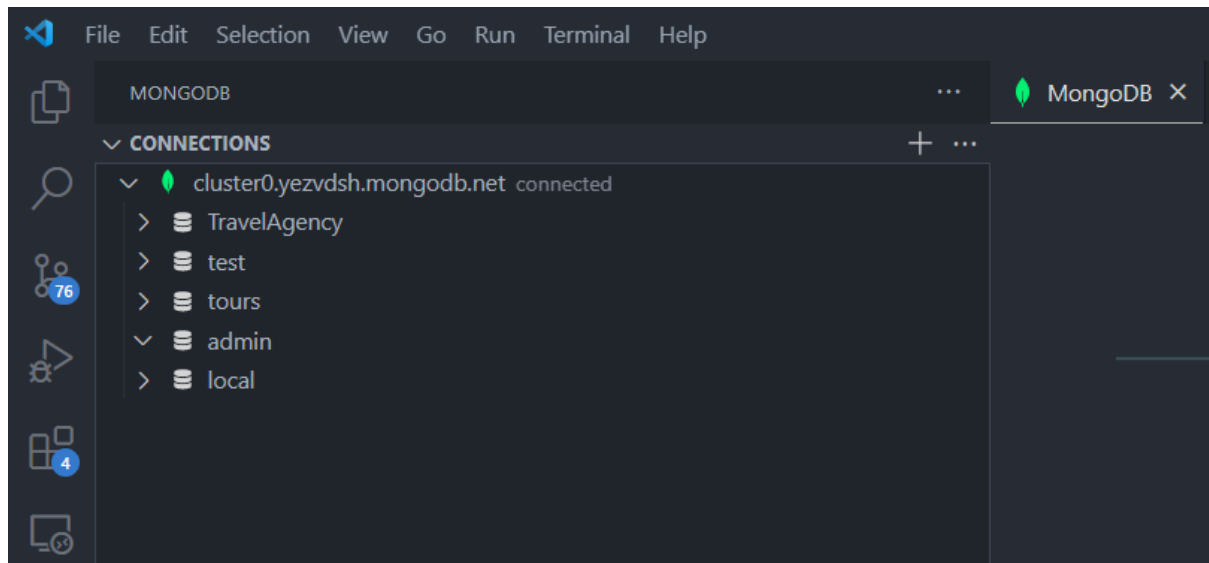


**AGH**

Laboratorium 4: MongoDB

Autor: Krzysztof Solecki

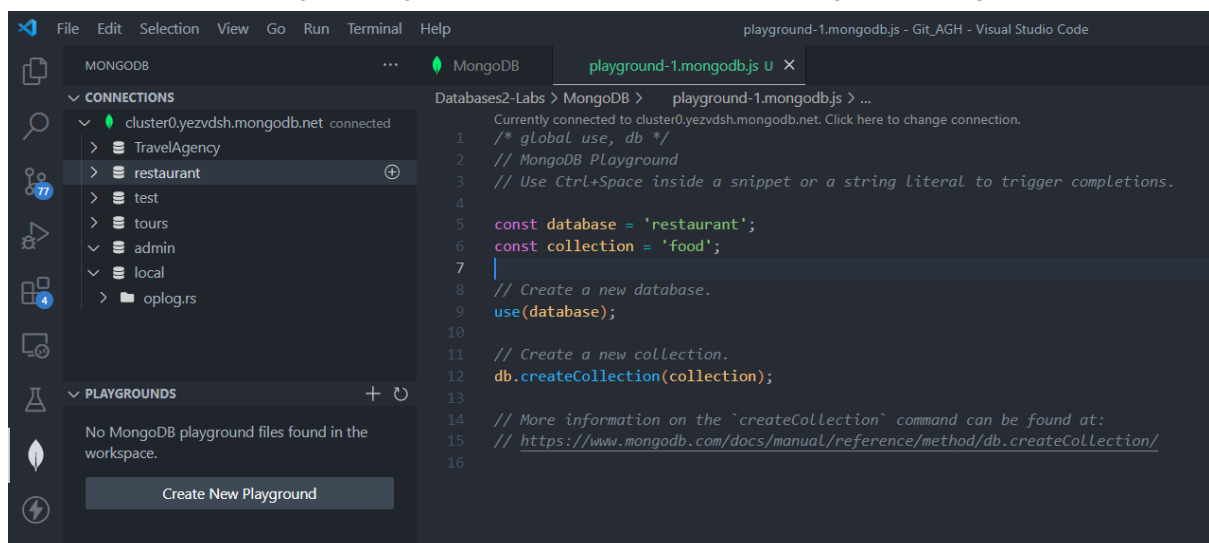
## 1. Połącz się z bazą MongoDB:



Połączenie ustanowiłem przy pomocy Wtyczki MongoDB w Visual Studio Code. Połączyłem się z klastrem utworzonym w serwisie MongoDB Atlas. Na zrzucie widoczne są już przykładowe bazy danych - są to bazy wykorzystane przeze mnie na potrzeby innych przedmiotów.

## 2. Stwórz bazę danych/kolekcję/dokument:

Utworzenie bazy danych “restaurant” i nowej kolekcji “food”:



Dodanie dwóch nowych dokumentów do kolekcji “food”:

```
const database = 'restaurant';
const collection = 'food';

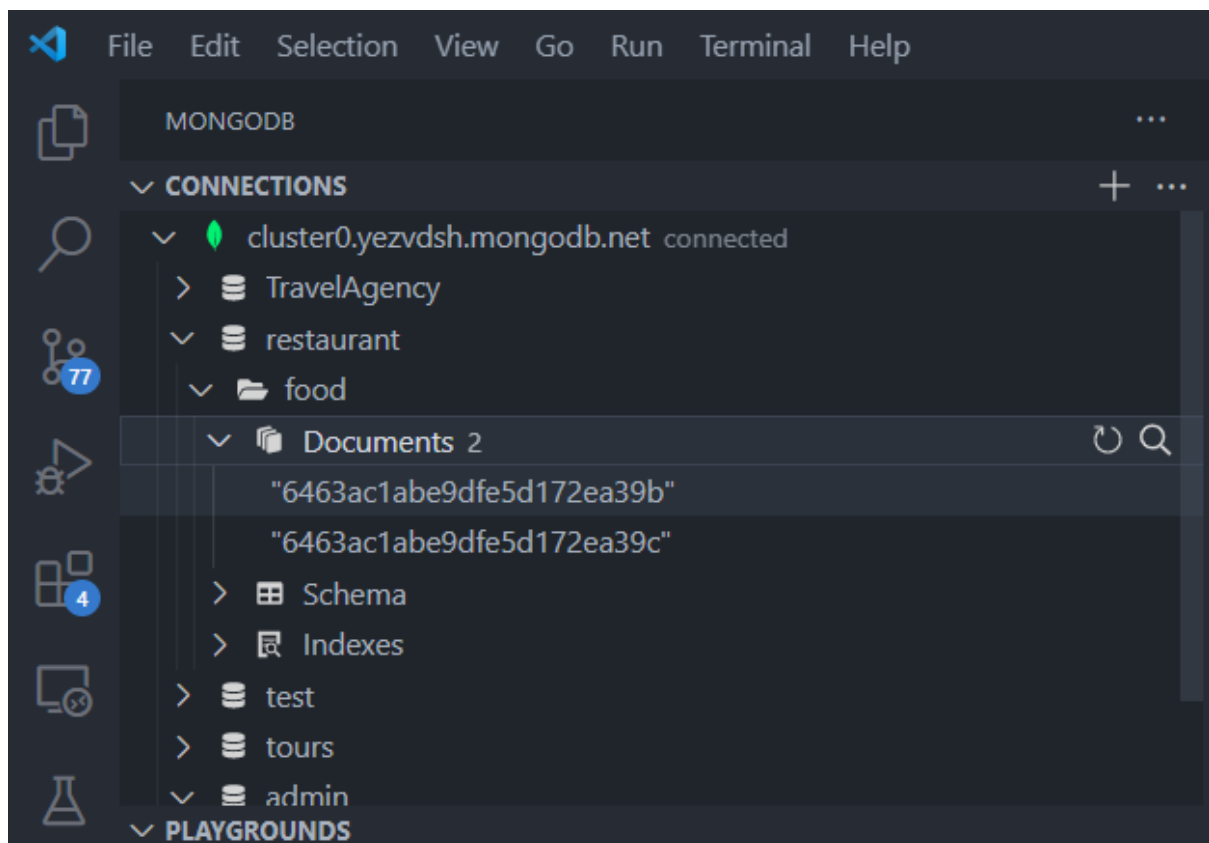
use(database);

db.food.insertMany(
  [{
    "part": "lunch",
    "foodEaten": [
      {
        "foodName": "screambled eggs",
        "serving": 2,
        "caloriesPerServe": 360,
        "proteinsPerServe": 34
      },
      {
        "foodName": "pancakes",
        "serving": 3,
        "caloriesPerServe": 367,
        "proteinsPerServe": 19
      },
      {
        "foodName": "sausages with beacon",
        "serving": 3,
        "caloriesPerServe": 512,
        "proteinsPerServe": 11
      }
    ]
  },
  {
    "part": "dinner",
    "foodEaten": [
      {
        "foodName": "chicken soup",
        "serving": 2,
        "caloriesPerServe": 100,
        "proteinsPerServe": 4
      },
      {
        "foodName": "sushi",
```

```

    "serving": 3,
    "caloriesPerServe": 430,
    "proteinsPerServe": 40
  }
]
}]
)

```



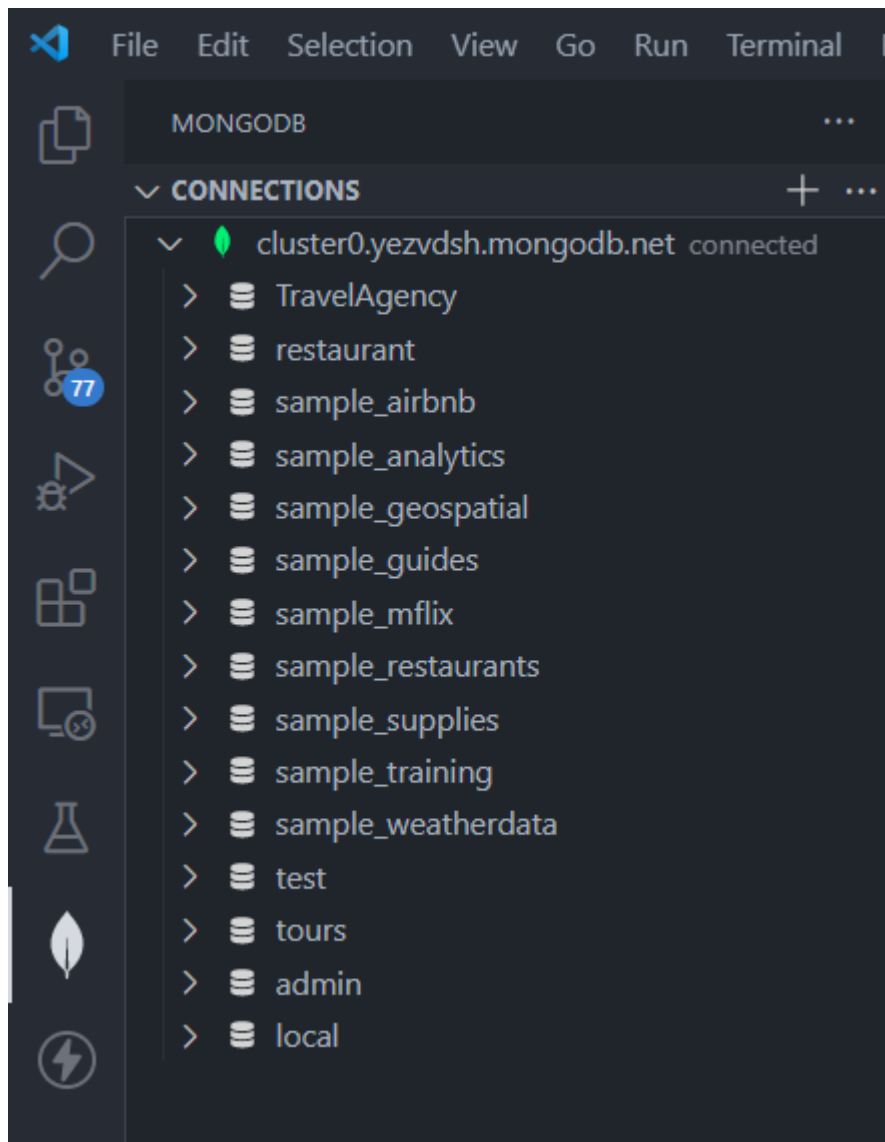
### 3. Zaimportuj przykładowe zbiory danych:

Zaimportowałem przykładowy zbiór danych MongoDB Atlas Sample Dataset:

```

PS C:\Users\Lenovo\Desktop\Git_AGH> atlas clusters loadSampleData Cluster0
● Sample Data Job 6463b0239c66b5390d713b7b created.
○ PS C:\Users\Lenovo\Desktop\Git_AGH> 

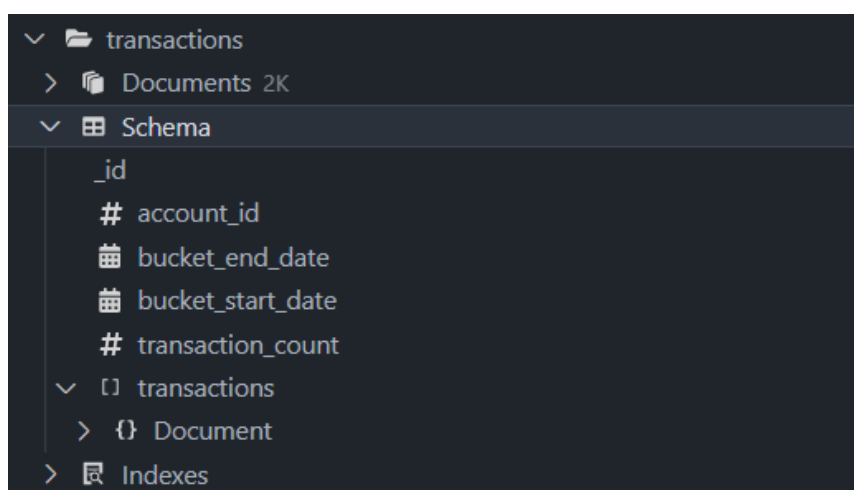
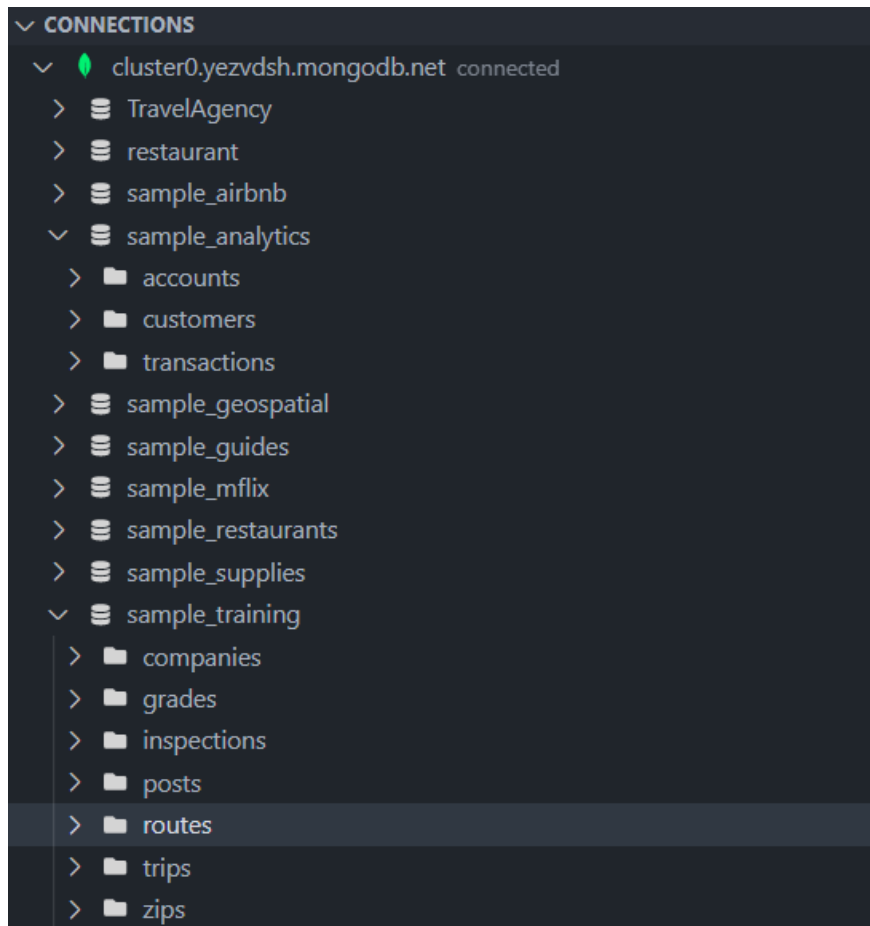
```

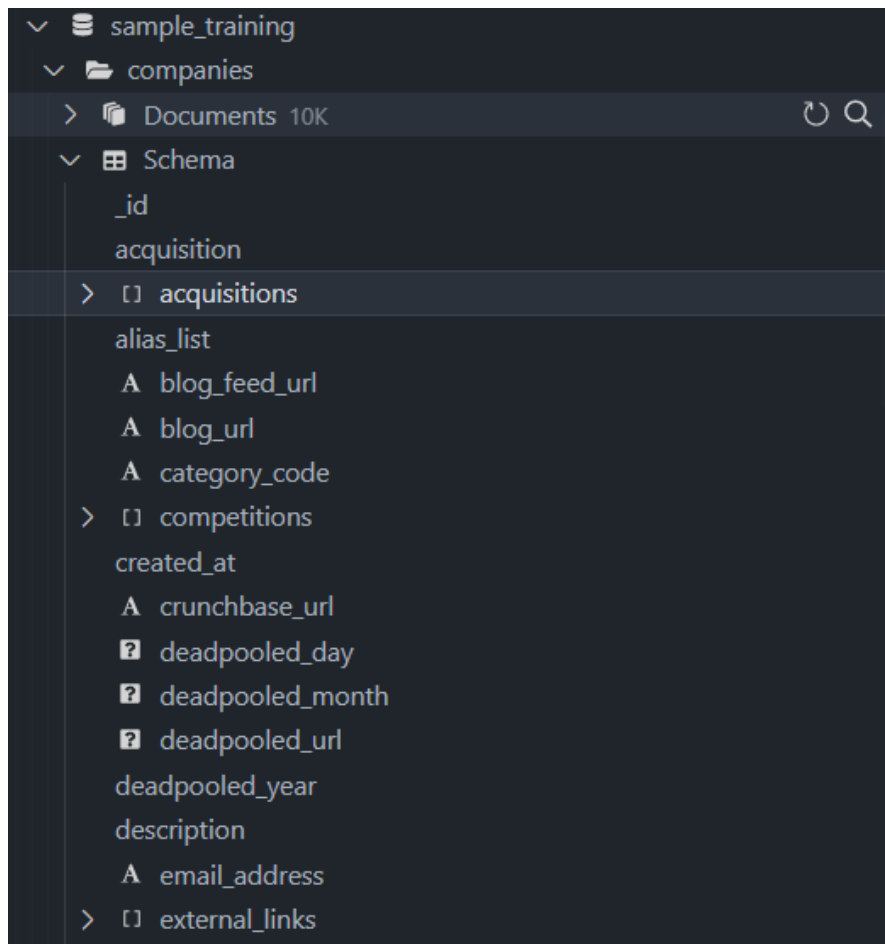


#### 4. Zapoznaj się ze strukturą przykładowych zbiorów danych/kolekcji:

- Sample Analytics Dataset - składa się z trzech kolekcji: Accounts, Customers i Transactions. W kolekcji transactions struktura dokumentu jest najbardziej złożona spośród tych trzech kolekcji np. posiada pole będące listą transakcji, gdzie każda z nich posiada również swoją strukturę (jest zagnieżdżonym dokumentem).
- Sample Training Dataset - składa się z siedmiu kolekcji: companies, grades, inspections, posts, routes, trips, zips.

W kolekcji companies mamy np. pole image będące zagnieżdżonym dokumentem składającym się z listy list dostępnych rozmiarów (pole numeryczne) oraz ścieżki do zdjęcia będącej stringiem.





## 5. Operacje CRUD:

- stwórz nową bazę danych, jako nazwy bazy danych użyj swoich inicjałów:

```
Databases2-Labs > MongoDB > playground-2.mongodb.js > ...
Currently connected to cluster0.yezvdsh.mongodb.net. Click here to change connection.
1 // 5. OPERACJE CRUD
2
3 // stwórz nową bazę danych, jako nazwy bazy danych użyj swoich inicjałów
4 const database = 'KS';
5 use(database);
6
```



- stwórz kolekcję "student" o informacje o studentach, przedmiotach ocenach z przedmiotów itp. o zaproponuj strukturę dokumentu o wykorzystaj typy proste/podstawowe, dokumenty zagnieżdżone, tablice itp. Wprowadź kilka przykładowych dokumentów.


```
const collection = 'student';
db.student.insertMany([
  {
    "firstName": "Adam",
    "lastName": "Nowakowski",
    "age": 22,
    "gender": "M",
    "major": "Ekonomia",
    "enrollmentYear": 2019,
    "courses": [
      {
        "name": "Mikroekonomia",
        "code": "ECO101",
        "grade": "B+"
      },
      {
        "name": "Statystyka ekonomiczna",
        "code": "ECO202",
        "grade": "A"
      }
    ],
    "address": {
      "street": "ul. Gospodarcza 15",
      "city": "Gdańsk",
      "country": "Polska"
    },
    "contact": {
      "email": "adam.nowakowski@example.com",
      "phone": "+48 987654321"
    }
  },
  {
    "firstName": "Maria",
    "lastName": "Wójcik",
```





```
"age": 23,
"gender": "F",
"major": "Historia",
"enrollmentYear": 2018,
"courses": [
  {
    "name": "Historia Polski",
    "code": "HIS101",
    "grade": "A-"
  },
  {
    "name": "Historia Europy",
    "code": "HIS201",
    "grade": "B"
  }
],
"address": {
  "street": "ul. Historyczna 8",
  "city": "Poznań",
  "country": "Polska"
},
"contact": {
  "email": "maria.wojcik@example.com",
  "phone": "+48 555444333"
}
},
{
  "firstName": "Piotr",
  "lastName": "Jankowski",
  "age": 20,
  "gender": "M",
  "major": "Medycyna",
  "enrollmentYear": 2021,
  "courses": [
    {
      "name": "Anatomia",
      "code": "MED101",
      "grade": "A"
    },
    {
      "name": "Biochemia",
      "code": "MED202",
      "grade": "B"
    }
  ]
}
```

```
    }
  ],
  "address": {
    "street": "ul. Medyczna 3",
    "city": "Wrocław",
    "country": "Polska"
  },
  "contact": {
    "email": "piotr.jankowski@example.com",
    "phone": "+48 777888999"
  }
},
{
  "firstName": "Katarzyna",
  "lastName": "Kowalczyk",
  "age": 22,
  "gender": "F",
  "major": "Socjologia",
  "enrollmentYear": 2019,
  "courses": [
    {
      "name": "Podstawy socjologii",
      "code": "SOC101",
      "grade": "B+"
    },
    {
      "name": "Metody badawcze",
      "code": "SOC202",
      "grade": "A-"
    }
  ],
  "address": {
    "street": "ul. Krakowska 13",
    "city": "Kraków",
    "country": "Polska"
  },
  "contact": {
    "email": "katarzyna.kowalczyk@example.com",
    "phone": "+48 888888999"
  }
}
])
```

▼  cluster0.yezvdsh.mongodb.net connected 

▼  KS

▼  student


▼  Documents 4

"646e45d9412379f8ae9e408c"

"646e45d9412379f8ae9e408d"

"646e45d9412379f8ae9e408e"

"646e45d9412379f8ae9e408f"

▼  Schema

\_id

> {} address

# age

> {} contact

> [] courses


# enrollmentYear

A firstName

A gender

A lastName

A major

>  Indexes

- przetestuj operacje wstawiania, modyfikacji/usuwania dokumentów:

☐ Wstawienie dokumentu:

```
//Dodanie pojedynczego dokumentu
db.student.insertOne({
  "firstName": "Marta",
  "lastName": "Nowicka",
  "age": 21,
  "gender": "F",
  "major": "Matematyka",
  "enrollmentYear": 2022,
  "courses": [
    {
      "name": "Analiza matematyczna",
      "code": "MAT101",
      "grade": "A"
    },
    {
      "name": "Algebra liniowa",
      "code": "MAT201",
      "grade": "A-"
    }
  ],
  "address": {
    "street": "ul. Matematyczna 5",
    "city": "Kraków",
    "country": "Polska"
  },
  "contact": {
    "email": "marta.nowicka@example.com",
    "phone": "+48 777111222"
  }
})
```

KS.student:{"\$oid":"646e47e122d34af30c222331"}.json > ...

```
1  {
2    "_id": {
3      "$oid": "646e47e122d34af30c222331"
4    },
5    "firstName": "Marta",
6    "lastName": "Nowicka",
7    "age": 21,
8    "gender": "F",
9    "major": "Matematyka",
10   "enrollmentYear": 2022,
11   "courses": [
12     {
13       "name": "Analiza matematyczna",
14       "code": "MAT101",
15       "grade": "A"
16     },
17     {
18       "name": "Algebra liniowa",
19       "code": "MAT201",
20       "grade": "A-"
21     }
22   ],
23   "address": {
24     "street": "ul. Matematyczna 5",
25     "city": "Kraków",
26     "country": "Polska"
27   },
28   "contact": {
29     "email": "marta.nowicka@example.com",
30     "phone": "+48 777111222"
31   }
32 }
```

## □ Modyfikacja dokumentu:

```
//Modyfikacja dokumentu
db.student.updateOne(
  { "firstName": "Adam" },
  { $set: { "lastName": "Nowak" } }
)
```

Przed zmianą:

```
KS.student:{"_id":"646e47e022d34af30c22232d"},json > ...
1 {
2   "_id": {
3     "$oid": "646e47e022d34af30c22232d"
4   },
5   "firstName": "Adam",
6   "lastName": "Nowakowski",
7   "age": 22,
8   "gender": "M",
9   "major": "Ekonomia",
10  "enrollmentYear": 2019,
11  "courses": [
12    {
13      "name": "Mikroekonomia",
14      "code": "EC0101",
15      "grade": "B+"
16    },
17    {
18      "name": "Statystyka ekonomiczna",
19      "code": "EC0202",
20      "grade": "A"
21    }
22  ],
23  "address": {
24    "street": "ul. Gospodarcza 15",
25    "city": "Gdańsk",
26    "country": "Polska"
27  },
28  "contact": {
29    "email": "adam.nowakowski@example.com",
30    "phone": "+48 987654321"
31  }
32 }
```

Po zmianie:

```
KS.student:{"$oid":"646e47e022d34af30c22232d"},json > ...
1  {
2    "_id": {
3      "$oid": "646e47e022d34af30c22232d"
4    },
5    "firstName": "Adam",
6    "lastName": "Nowak",
7    "age": 22,
8    "gender": "M",
9    "major": "Ekonomia",
10   "enrollmentYear": 2019,
11   "courses": [
12     {
13       "name": "Mikroekonomia",
14       "code": "EC0101",
15       "grade": "B+"
16     },
17     {
18       "name": "Statystyka ekonomiczna",
19       "code": "EC0202",
20       "grade": "A"
21     }
22   ],
23   "address": {
24     "street": "ul. Gospodarcza 15",
25     "city": "Gdańsk",
26     "country": "Polska"
27   },
28   "contact": {
29     "email": "adam.nowakowski@example.com",
30     "phone": "+48 987654321"
31   }
32 }
```

## ☐ Usunięcie dokumentu:

```
//Usunięcie dokumentu
db.student.deleteOne({ "firstName": "Adam" })
```

## Potwierdzenie wykonania zapytania:

Playground Result ✕

```
1 {
2   "acknowledged": true,
3   "deletedCount": 1
4 }
```

## ☐ Wyszukiwanie dokumentów:

```
//Wyszukiwanie dokumentów

// Znalezienie wszystkich studentów
db.student.find()

// Znalezienie studenta o nazwisku "Jankowski"
db.student.find({ "lastName": "Jankowski" })

💡 Znalezienie studentów z oceną "A" w kursie "Analiza Matematyczna"
db.student.find({ "courses": { $elemMatch: { "name": "Analiza matematyczna", "grade": "A" } } })
```

## Znalezienie wszystkich studentów:

```
[
  {
    "_id": {
      "$oid": "646e47e022d34af30c22232e"
    },
    "firstName": "Maria",
    "lastName": "Wójcik",
    "age": 23,
    "gender": "F",
    "major": "Historia",
    "enrollmentYear": 2018,
    "courses": [
      {
        "name": "Historia Polski",
```



```
        "code": "HIS101",
        "grade": "A-"
    },
    {
        "name": "Historia Europy",
        "code": "HIS201",
        "grade": "B"
    }
],
"address": {
    "street": "ul. Historyczna 8",
    "city": "Poznań",
    "country": "Polska"
},
"contact": {
    "email": "maria.wojcik@example.com",
    "phone": "+48 555444333"
}
},
{
    "_id": {
        "$oid": "646e47e022d34af30c22232f"
    },
    "firstName": "Piotr",
    "lastName": "Jankowski",
    "age": 20,
    "gender": "M",
    "major": "Medycyna",
    "enrollmentYear": 2021,
    "courses": [
        {
            "name": "Anatomia",
            "code": "MED101",
            "grade": "A"
        },
        {
            "name": "Biochemia",
            "code": "MED202",
            "grade": "B"
        }
    ],
    "address": {
        "street": "ul. Medyczna 3",
```

```
    "city": "Wrocław",
    "country": "Polska"
  },
  "contact": {
    "email": "piotr.jankowski@example.com",
    "phone": "+48 777888999"
  }
},
{
  "_id": {
    "$oid": "646e47e022d34af30c222330"
  },
  "firstName": "Katarzyna",
  "lastName": "Kowalczyk",
  "age": 22,
  "gender": "F",
  "major": "Socjologia",
  "enrollmentYear": 2019,
  "courses": [
    {
      "name": "Podstawy socjologii",
      "code": "SOC101",
      "grade": "B+"
    },
    {
      "name": "Metody badawcze",
      "code": "SOC202",
      "grade": "A-"
    }
  ],
  "address": {
    "street": "ul. Krakowska 13",
    "city": "Kraków",
    "country": "Polska"
  },
  "contact": {
    "email": "katarzyna.kowalczyk@example.com",
    "phone": "+48 888888999"
  }
},
{
  "_id": {
    "$oid": "646e47e122d34af30c222331"
```

```
    },
    "firstName": "Marta",
    "lastName": "Nowicka",
    "age": 21,
    "gender": "F",
    "major": "Matematyka",
    "enrollmentYear": 2022,
    "courses": [
      {
        "name": "Analiza matematyczna",
        "code": "MAT101",
        "grade": "A"
      },
      {
        "name": "Algebra liniowa",
        "code": "MAT201",
        "grade": "A-"
      }
    ],
    "address": {
      "street": "ul. Matematyczna 5",
      "city": "Kraków",
      "country": "Polska"
    },
    "contact": {
      "email": "marta.nowicka@example.com",
      "phone": "+48 777111222"
    }
  }
]
```

Znalezienie studenta o nazwisku "Jankowski":

```
Playground Result X
1  [
2    {
3      Edit Document
4      "_id": {
5        "$oid": "646e47e022d34af30c22232f"
6      },
7      "firstName": "Piotr",
8      "lastName": "Jankowski",
9      "age": 20,
10     "gender": "M",
11     "major": "Medycyna",
12     "enrollmentYear": 2021,
13     "courses": [
14       {
15         "name": "Anatomia",
16         "code": "MED101",
17         "grade": "A"
18       },
19       {
20         "name": "Biochemia",
21         "code": "MED202",
22         "grade": "B"
23       }
24     ],
25     "address": {
26       "street": "ul. Medyczna 3",
27       "city": "Wrocław",
28       "country": "Polska"
29     },
30     "contact": {
31       "email": "piotr.jankowski@example.com",
32       "phone": "+48 777888999"
33     }
34   ]
```

Znalezienie studentów z oceną "A" w kursie "Analiza matematyczna":

```
Playground Result X
1  [
2    {
3      Edit Document
4      "_id": {
5        "$oid": "646e47e122d34af30c222331"
6      },
7      "firstName": "Marta",
8      "lastName": "Nowicka",
9      "age": 21,
10     "gender": "F",
11     "major": "Matematyka",
12     "enrollmentYear": 2022,
13     "courses": [
14       {
15         "name": "Analiza matematyczna",
16         "code": "MAT101",
17         "grade": "A"
18       },
19       {
20         "name": "Algebra liniowa",
21         "code": "MAT201",
22         "grade": "A-"
23       }
24     ],
25     "address": {
26       "street": "ul. Matematyczna 5",
27       "city": "Kraków",
28       "country": "Polska"
29     },
30     "contact": {
31       "email": "marta.nowicka@example.com",
32       "phone": "+48 777111222"
33     }
34   ]
}
```