

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ  
УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту



**Лабораторна робота №6**

з дисципліни

«Організація баз даних та знань»

**Виконав:**

студент групи ШІ-23

Полуліх Ю.Ю.

**Викладач:**

Поберейко П. Б.

Львів – 2023 р.

**Тема:** розробка NoSQL бази даних

**Мета:** здобуття практичних навичок створення та обробки бази даних типу NoSQL на прикладі СУБД MongoDB.

### Хід роботи

*Створення бази даних, почнемо з створення валідації для кожної з колекцій:*

#### *Airlines collection:*

```
db.createCollection("airlines", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["Name", "Address", "IATA_Code", "Phone"],
      properties: {
        Name: { bsonType: "string" },
        Address: { bsonType: "string" },
        IATA_Code: {
          bsonType: "string",
          minLength: 3,
          maxLength: 3,
          uniqueItems: true,
        },
        Phone: { bsonType: "string" },
      },
    },
  },
});
```

#### *Flights collection:*

```
db.createCollection("flights", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["AirplaneId", "AirlineId",
        "PointOfDeparture", "Destination", "DateTimeOfDeparture",
        "DateTimeOfArrival"],
      properties: {
        AirplaneId: { bsonType: "objectId" },
        AirlineId: { bsonType: "objectId" },
        PointOfDeparture: { bsonType: "string" },
        Destination: { bsonType: "string" },
        DateTimeOfDeparture: { bsonType: "date" },
        DateTimeOfArrival: {
```

```

        bsonType: "date"
    },
},
},
},
});

```

#### **staff collection:**

```

db.createCollection("staff", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["Name", "Position", "DateOfEmployment",
"ContactPhone", "AirlineId"],
      properties: {
        Name: { bsonType: "string" },
        Position: { bsonType: "string" },
        DateOfEmployment: { bsonType: "date" },
        ContactPhone: { bsonType: "string" },
        AirlineId: { bsonType: "objectId" },
        AirplaneIds: {
          bsonType: "array",
          items: {
            bsonType: "objectId",
          },
        },
      },
    },
  },
});

```

#### **Airplanes collection:**

```

db.createCollection("airplanes", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["AirlineId", "ModelName", "Manufacturer",
"YearOfManufacture", "NumberOfSeats", "RegistrationNumber"],
      properties: {
        AirlineId: { bsonType: "objectId" },
        ModelName: { bsonType: "string" },
        Manufacturer: { bsonType: "string" },
        YearOfManufacture: { bsonType: "date" },
        NumberOfSeats: { bsonType: "int" },
        RegistrationNumber: {

```

```

        bsonType: "string",
        uniqueItems: true,
    },
},
},
},
});

```

### ***Booking collection:***

```

db.createCollection("booking", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["AirlineId", "FlightId", "Passenger",
        "Date", "PlaceOfPassenger", "Status"],
      properties: {
        AirlineId: { bsonType: "objectId" },
        FlightId: { bsonType: "objectId" },
        Passenger: {
          bsonType: "object",
          required: ["Name", "DateOfBirth", "Passport",
            "ContactInformation"],
          properties: {
            Name: { bsonType: "string" },
            DateOfBirth: { bsonType: "date" },
            Passport: { bsonType: "long" },
            ContactInformation: { bsonType: "string" }
          }
        },
        Date: { bsonType: "date" },
        PlaceOfPassenger: { bsonType: "int" },
        Status: { bsonType: "string" }
      }
    }
  }
});

```

### **Заповнення даних:**

#### ***Airlines collection:***

```

db.airlines.insertMany(
[
{
  "Name": "SkyHigh",

```

```

    "Address": "1234 Jetway Blvd",
    "IATA_Code": "SKH",
    "Phone": "123-456-7890"
  },
  {
    "Name": "BlueSky",
    "Address": "5678 Skyline Dr, Clear Sky",
    "IATA_Code": "BLU",
    "Phone": "987-654-3210"
  },
  {
    "Name": "StarFlight",
    "Address": "9101 Star Route, Galaxy",
    "IATA_Code": "STF",
    "Phone": "456-789-0123"
  }
];

```

▶

```

_id: ObjectId('6550f7608cd7f6e972e11c6f')
Name: "SkyHigh"
Address: "1234 Jetway Blvd"
IATA_Code: "SKH"
Phone: "123-456-7890"

```

```

_id: ObjectId('6550f7608cd7f6e972e11c70')
Name: "BlueSky"
Address: "5678 Skyline Dr, Clear Sky"
IATA_Code: "BLU"
Phone: "987-654-3210"

```

```

_id: ObjectId('6550f7608cd7f6e972e11c71')
Name: "StarFlight"
Address: "9101 Star Route, Galaxy"
IATA_Code: "STF"
Phone: "456-789-0123"

```

### *Airplanes collection:*

```

db.airplanes.insertMany([
  {
    "AirlineId": ObjectId('6550f7608cd7f6e972e11c6f'),
    "ModelName": "A320",
    "Manufacturer": "Airbus",
    "YearOfManufacture": ISODate("2015-01-01"),
    "NumberOfSeats": 150,
    "RegistrationNumber": "A320-111"
  },
  {

```

```

    "AirlineId": ObjectId('6550f7608cd7f6e972e11c70'),
    "ModelName": "737",
    "Manufacturer": "Boeing",
    "YearOfManufacture": ISODate("2016-02-01"),
    "NumberOfSeats": 180,
    "RegistrationNumber": "B737-222"
  },
  {
    "AirlineId": ObjectId("6550f7608cd7f6e972e11c71"),
    "ModelName": "757",
    "Manufacturer": "Boeing",
    "YearOfManufacture": ISODate("2017-03-01"),
    "NumberOfSeats": 200,
    "RegistrationNumber": "B757-333"
  }
];

```

```

_id: ObjectId('6550f9f08cd7f6e972e11c72')
AirlineId: ObjectId('6550f7608cd7f6e972e11c6f')
ModelName: "A320"
Manufacturer: "Airbus"
YearOfManufacture: 2015-01-01T00:00:00.000+00:00
NumberOfSeats: 150
RegistrationNumber: "A320-111"

```

```

_id: ObjectId('6550f9f08cd7f6e972e11c73')
AirlineId: ObjectId('6550f7608cd7f6e972e11c70')
ModelName: "737"
Manufacturer: "Boeing"
YearOfManufacture: 2016-02-01T00:00:00.000+00:00
NumberOfSeats: 180
RegistrationNumber: "B737-222"

```

```

_id: ObjectId('6550f9f08cd7f6e972e11c74')
AirlineId: ObjectId('6550f7608cd7f6e972e11c71')
ModelName: "757"
Manufacturer: "Boeing"
YearOfManufacture: 2017-03-01T00:00:00.000+00:00
NumberOfSeats: 200
RegistrationNumber: "B757-333"

```

### *Flights collection:*

```

db.flights.insertMany(
[
{
  "AirplaneId": ObjectId("6550f9f08cd7f6e972e11c72"),
  "AirlineId": ObjectId("6550f7608cd7f6e972e11c6f"),
  "PointOfDeparture": "Cloud City",
  "Destination": "Clear Sky",
  "DateTimeOfDeparture": ISODate("2023-11-12 08:00:00"),

```

```

    "DateTimeOfArrival": ISODate("2023-11-12 11:00:00")
  },
  {
    "AirplaneId": ObjectId("6550f9f08cd7f6e972e11c73"),
    "AirlineId": ObjectId("6550f7608cd7f6e972e11c70"),
    "PointOfDeparture": "Clear Sky",
    "Destination": "Galaxy",
    "DateTimeOfDeparture": ISODate("2023-11-12 09:30:00"),
    "DateTimeOfArrival": ISODate("2023-11-12 12:30:00")
  },
  {
    "AirplaneId": ObjectId("6550f9f08cd7f6e972e11c74"),
    "AirlineId": ObjectId("6550f7608cd7f6e972e11c71"),
    "PointOfDeparture": "Galaxy",
    "Destination": "Shoreline",
    "DateTimeOfDeparture": ISODate("2023-11-12 07:45:00"),
    "DateTimeOfArrival": ISODate("2023-11-12 10:50:00")
  }
]);

```

```

    _id: ObjectId('6550fb688cd7f6e972e11c75')
    2023-11-12T07:45:00.000+00:00 ('6550f9f08cd7f6e972e11c72')
    AirplaneId: ObjectId('6550f9f08cd7f6e972e11c72')
    AirlineId: ObjectId('6550f7608cd7f6e972e11c6f')
    PointOfDeparture: "Cloud City"
    Destination: "Clear Sky"
    DateTimeOfDeparture: 2023-11-12T08:00:00.000+00:00
    DateTimeOfArrival: 2023-11-12T11:00:00.000+00:00

```

```

    _id: ObjectId('6550fb688cd7f6e972e11c76')
    AirplaneId: ObjectId('6550f9f08cd7f6e972e11c73')
    AirlineId: ObjectId('6550f7608cd7f6e972e11c70')
    PointOfDeparture: "Clear Sky"
    Destination: "Galaxy"
    DateTimeOfDeparture: 2023-11-12T09:30:00.000+00:00
    DateTimeOfArrival: 2023-11-12T12:30:00.000+00:00

```

```

    ▶ _id: ObjectId('6550fb688cd7f6e972e11c77')
    AirplaneId: ObjectId('6550f9f08cd7f6e972e11c74')
    AirlineId: ObjectId('6550f7608cd7f6e972e11c71')
    PointOfDeparture: "Galaxy"
    Destination: "Shoreline"
    DateTimeOfDeparture: 2023-11-12T07:45:00.000+00:00
    DateTimeOfArrival: 2023-11-12T10:50:00.000+00:00

```

### Staff collection:

```

db.staff.insertMany(
[
{
  "Name": "Alice Johnson",

```

```

    "Position": "Pilot",
    "DateOfEmployment": ISODate("2018-02-01"),
    "ContactPhone": "678-901-2345",
    "AirlineId": ObjectId("6550f7608cd7f6e972e11c6f"),
    "AirplaneIds": [ObjectId("6550f9f08cd7f6e972e11c72")]},
{
    "Name": "Bob Smith",
    "Position": "Co-Pilot",
    "DateOfEmployment": ISODate("2019-06-15"),
    "ContactPhone": "789-012-3456",
    "AirlineId": ObjectId("6550f7608cd7f6e972e11c70"),
    "AirplaneIds": [ObjectId("6550f9f08cd7f6e972e11c73")]
},
{
    "Name": "Charlie Reeds",
    "Position": "Flight Attendant",
    "DateOfEmployment": ISODate("2020-05-23"),
    "ContactPhone": "456-789-0123",
    "AirlineId": ObjectId("6550f7608cd7f6e972e11c71"),
    "AirplaneIds": [ObjectId("6550f9f08cd7f6e972e11c74")]
}
]);

```

### ***Booking collection:***

```

db.booking.insertMany([
{
    "AirlineId": ObjectId('6550f7608cd7f6e972e11c6f'),
    "FlightId": ObjectId('6550fb688cd7f6e972e11c75'),
    "Passenger": {
        "Name": "John Smith",
        "DateOfBirth": ISODate("1980-06-15"),
        "Passport": 123456789,
        "ContactInformation": "john.smith@email.com"
    },
    "Date": ISODate("2023-11-10"),
    "PlaceOfPassenger": 12,
    "Status": "Confirmed"
},
{
    "AirlineId": ObjectId('6550f7608cd7f6e972e11c70'),
    "FlightId": ObjectId('6550fb688cd7f6e972e11c76'),
    "Passenger": {

```



```

    "Name": "Emily Johnson",
    "DateOfBirth": ISODate("1992-04-22"),
    "Passport": 987654321,
    "ContactInformation": "emily.j@email.com"
  },
  "Date": ISODate("2023-11-11"),
  "PlaceOfPassenger": 7,
  "Status": "Confirmed"
},
{
  "AirlineId": ObjectId('6550f7608cd7f6e972e11c71'),
  "FlightId": ObjectId('6550fb688cd7f6e972e11c77'),
  "Passenger": {
    "Name": "Michael Brown",
    "DateOfBirth": ISODate("1978-02-12"),
    "Passport": 111222333,
    "ContactInformation": "m.brown@email.com"
  },
  "Date": ISODate("2023-11-12"),
  "PlaceOfPassenger": 31,
  "Status": "Confirmed"
}
]);

```

```

_id: ObjectId('655101c58cd7f6e972e11c87')
AirlineId: ObjectId('6550f7608cd7f6e972e11c6f')
FlightId: ObjectId('6550fb688cd7f6e972e11c75')
▸ Passenger: Object
Date: 2023-11-10T00:00:00.000+00:00
PlaceOfPassenger: 12
Status: "Confirmed"

```

```

_id: ObjectId('655101c58cd7f6e972e11c88')
AirlineId: ObjectId('6550f7608cd7f6e972e11c70')
FlightId: ObjectId('6550fb688cd7f6e972e11c76')
▸ Passenger: Object
Date: 2023-11-11T00:00:00.000+00:00
PlaceOfPassenger: 7
Status: "Confirmed"

```

```

_id: ObjectId('655101c58cd7f6e972e11c89')
AirlineId: ObjectId('6550f7608cd7f6e972e11c71')
FlightId: ObjectId('6550fb688cd7f6e972e11c77')
▸ Passenger: Object
Date: 2023-11-12T00:00:00.000+00:00
PlaceOfPassenger: 31
Status: "Confirmed"

```

## 1. Додавання, оператор insertOne():

```
> db.airlines.insertOne({"Name" : "Ukraine Airline", "Address" : "Lvivska street", "IATA_Code" : "UKA", "Phone" : "312-141-4233"})
< {
  acknowledged: true,
  insertedId: ObjectId("655110688cd7f6e972e11c8d")
}
```

### Результат:

```
_id: ObjectId('655110688cd7f6e972e11c8d')
Name: "Ukraine Airline"
Address: "Lvivska street"
IATA_Code: "UKA"
Phone: "312-141-4233"
```

## 2. Зміна, оператор updateOne()

```
> db.airlines.updateOne({"_id" : ObjectId("655110688cd7f6e972e11c8d")}, {$set : {"name" : "UIA"}})
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

### Результат:

```
▶ _id: ObjectId('655110688cd7f6e972e11c8d')
   Name: "UIA"
   Address: "Lvivska street"
   IATA_Code: "UKA"
   Phone: "312-141-4233"
```

## 3. Видалення, оператор deleteOne():

```
> db.airlines.deleteOne({"_id" : ObjectId('655110688cd7f6e972e11c8d')})
< {
  acknowledged: true,
  deletedCount: 1
}
```

**Висновок:** В ході виконання лабораторної роботи було створено схему бази даних для предметної області Авіакомпанії. Використовуючи підходи MongoDB були визначені структури даних та їх обмеження. Перетворені сутності відображають взаємозв'язки та особливості об'єктної моделі у MongoDB, забезпечуючи гнучкість та можливість розширення системи. Реалізація функцій

редагування, додавання та вилучення даних дозволяє забезпечити ефективне взаємодію з базою даних відповідно до вимог користувача.