CENG 316 Software Engineering

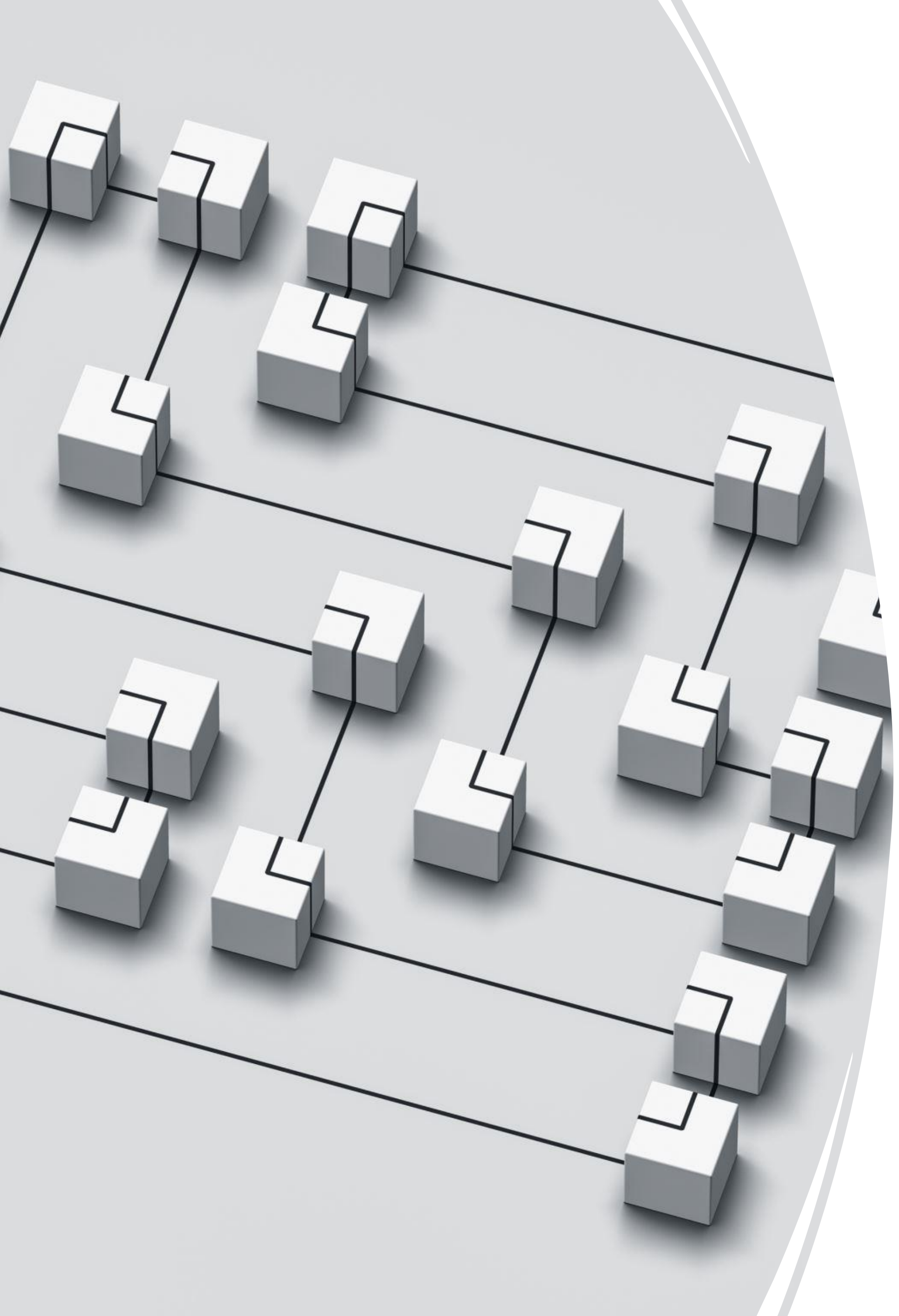
Popular Database Options

Databases

- Databases are software systems that allow you to store, manage, and retrieve data. They are essential for building web applications, as they enable efficient storage and retrieval of large amounts of data

Relational Databases

- Relational databases are based on the relational model of data, where data is stored in tables with predefined columns and rows, and the relationships between tables are defined by foreign keys
- Relational databases use SQL (Structured Query Language) to query and manipulate data, and are known for their data consistency and ability to enforce data integrity rules
- They are widely used in applications that require strict data consistency, such as banking and finance, where the accuracy of data is critical



NoSQL databases

- NoSQL databases are designed to store and manage large volumes of unstructured or semi-structured data, such as text, images, and videos.
- They use various data models, such as document, key-value, graph, and column-family, to store data, and typically do not enforce strict data consistency rules
- NoSQL databases are known for their ability to scale horizontally, handle high volumes of data, and provide fast data access
- They are often used in applications that require high scalability and performance, such as social media, e-commerce, and gaming

How to Choose

The choice between relational databases and NoSQL databases depends on the specific requirements of your application

If your application requires strict data consistency, relational databases may be a better choice

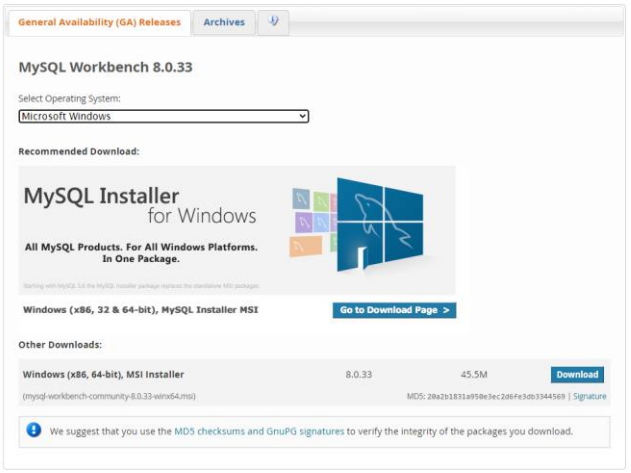
If your application deals with large volumes of unstructured or semi-structured data and needs to scale horizontally, NoSQL databases may be a better choice

Some of the Famous Options

- MySQL
- PostgreSQL
- MongoDB
- Microsoft SQL Server
- Oracle Database
- Amazon Aurora
- Google Cloud SQL
- Firebase Realtime Database
- Redis
- Cassandra

MySQL

- One of the most widely used relational databases, MySQL is an open-source database that is easy to use and supports various programming languages.



General Availability (GA) Releases | Archives

MySQL Workbench 8.0.33

Select Operating System: Microsoft Windows

Recommended Download:

MySQL Installer for Windows

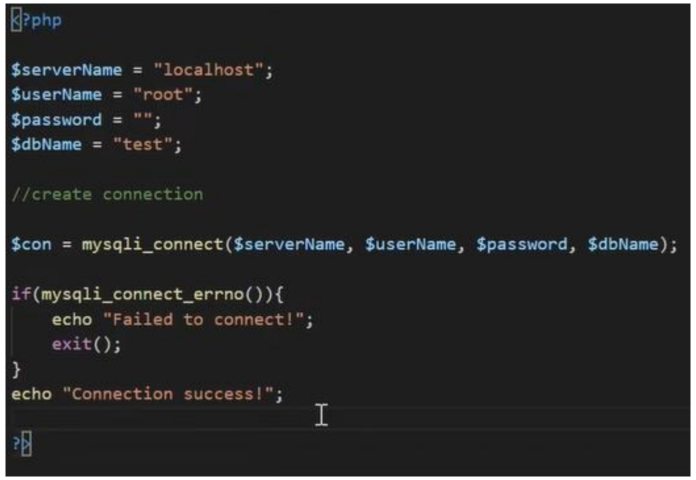
All MySQL Products. For All Windows Platforms. In One Package.

Windows (x86, 32 & 64-bit), MySQL Installer MSI — Go to Download Page >

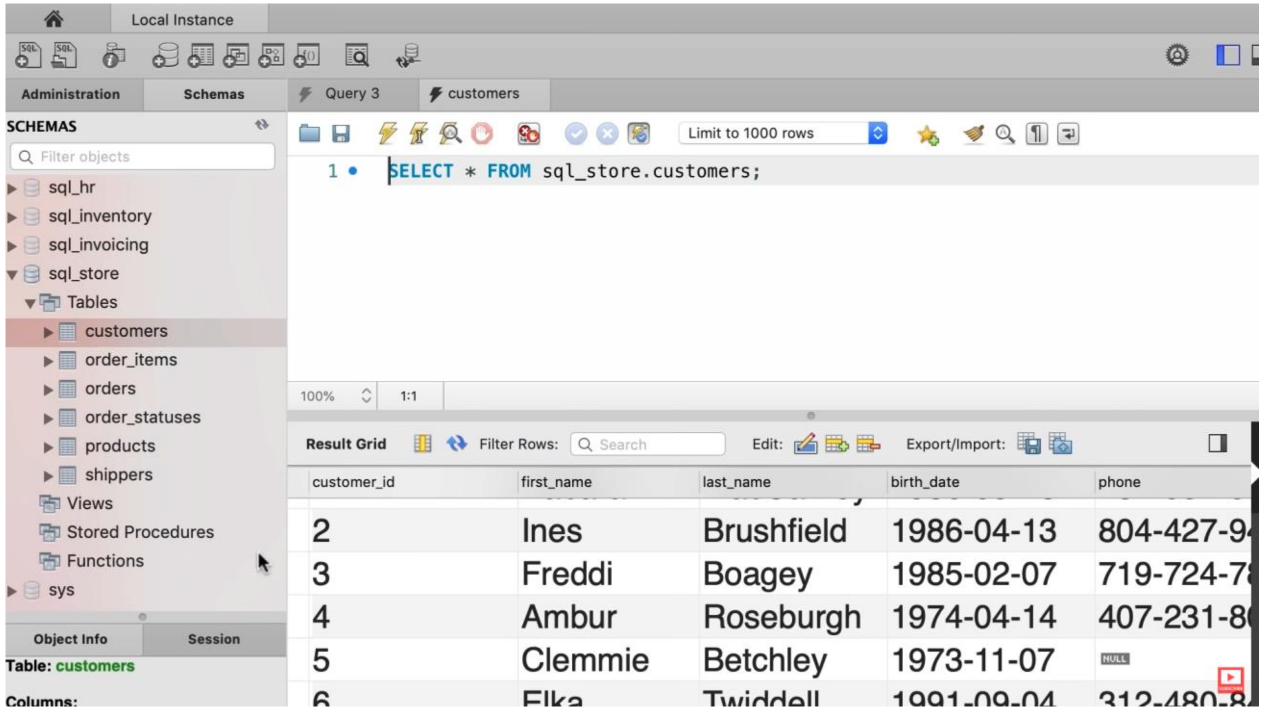
Other Downloads:

Windows (x86, 64-bit), MSI Installer (mysql-workbench-community-8.0.33-winx64.msi)	8.0.33	45.5M	Download MD5: 20a2b1831a950e3ec2d6fe3db3344569 | Signature

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.



```php
<?php

$serverName = "localhost";
$userName = "root";
$password = "";
$dbName = "test";

//create connection

$con = mysqli_connect($serverName, $userName, $password, $dbName);

if(mysqli_connect_errno()){
    echo "Failed to connect!";
    exit();
}
echo "Connection success!";

?>
```



Local Instance

Administration | Schemas | Query 3 | customers

SCHEMAS

- sql_hr
- sql_inventory
- sql_invoicing
- sql_store
  - Tables
    - customers
    - order_items
    - orders
    - order_statuses
    - products
    - shippers
  - Views
  - Stored Procedures
  - Functions
- sys

Limit to 1000 rows

```sql
SELECT * FROM sql_store.customers;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

customer_id	first_name	last_name	birth_date	phone
2	Ines	Brushfield	1986-04-13	804-427-9
3	Freddi	Boagey	1985-02-07	719-724-7
4	Ambur	Roseburgh	1974-04-14	407-231-8
5	Clemmie	Betchley	1973-11-07	NULL
6	Elka	Twiddell	1991-09-04	312-480-8

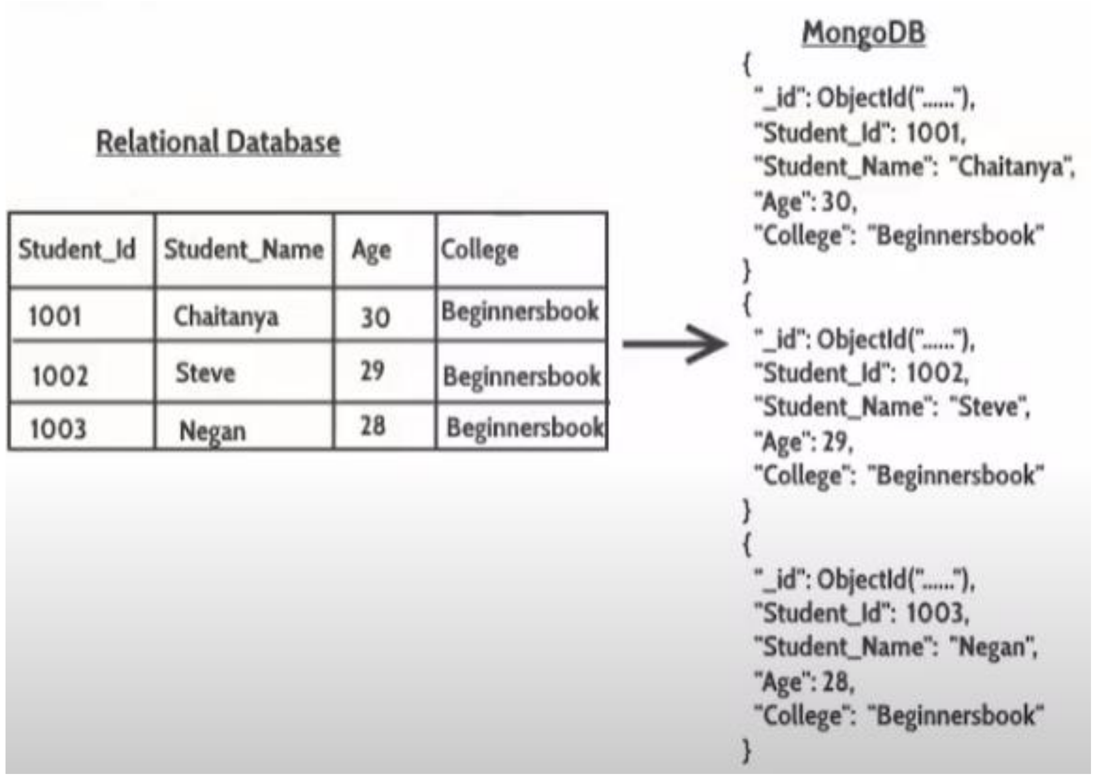
Object Info | Session

Table: customers

Columns:

MongoDB

- A document-oriented database, MongoDB stores data in a flexible JSON-like format, making it a good choice for applications that require high scalability and performance.

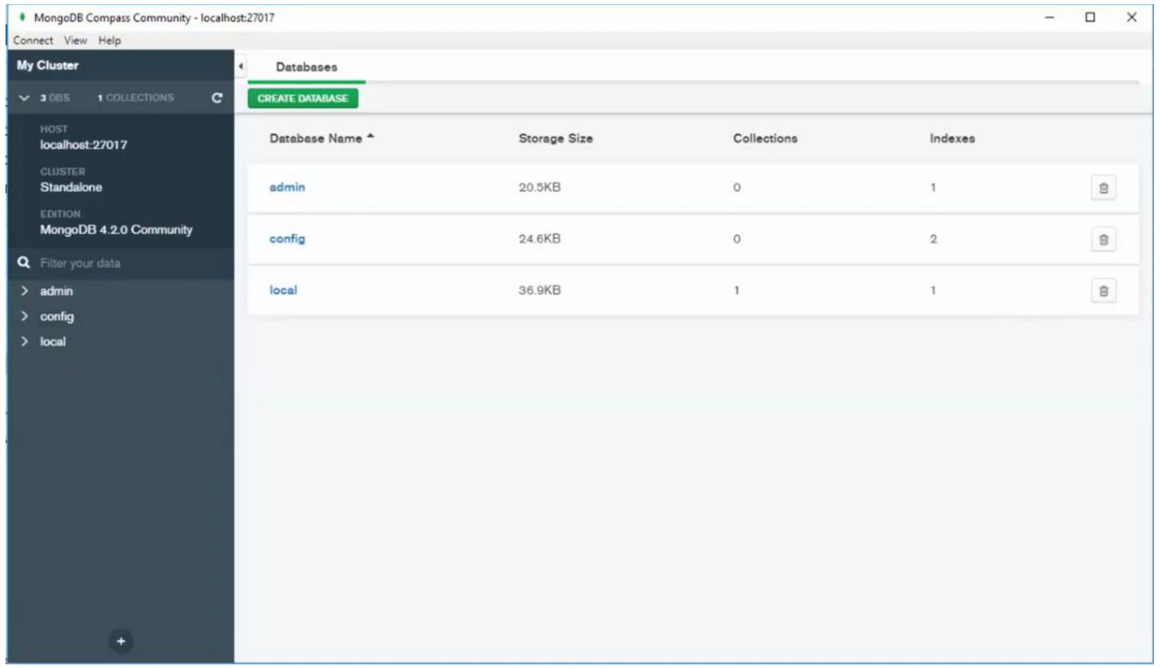


Relational Database

Student_Id	Student_Name	Age	College
1001	Chaitanya	30	Beginnersbook
1002	Steve	29	Beginnersbook
1003	Negan	28	Beginnersbook

MongoDB

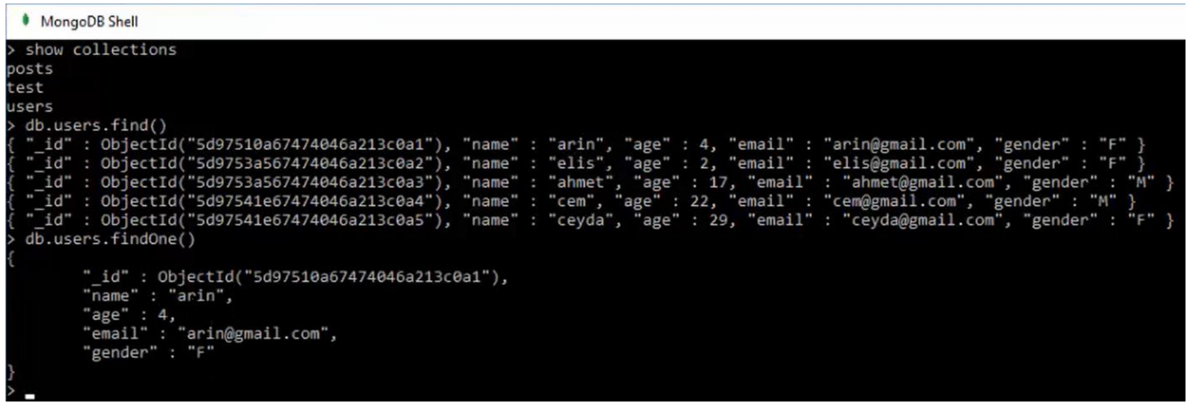
```
{
 "_id": ObjectId("......"),
 "Student_Id": 1001,
 "Student_Name": "Chaitanya",
 "Age": 30,
 "College": "Beginnersbook"
}
{
 "_id": ObjectId("......"),
 "Student_Id": 1002,
 "Student_Name": "Steve",
 "Age": 29,
 "College": "Beginnersbook"
}
{
 "_id": ObjectId("......"),
 "Student_Id": 1003,
 "Student_Name": "Negan",
 "Age": 28,
 "College": "Beginnersbook"
}
```



HOST
localhost:27017
CLUSTER
Standalone
EDITION
MongoDB 4.2.0 Community

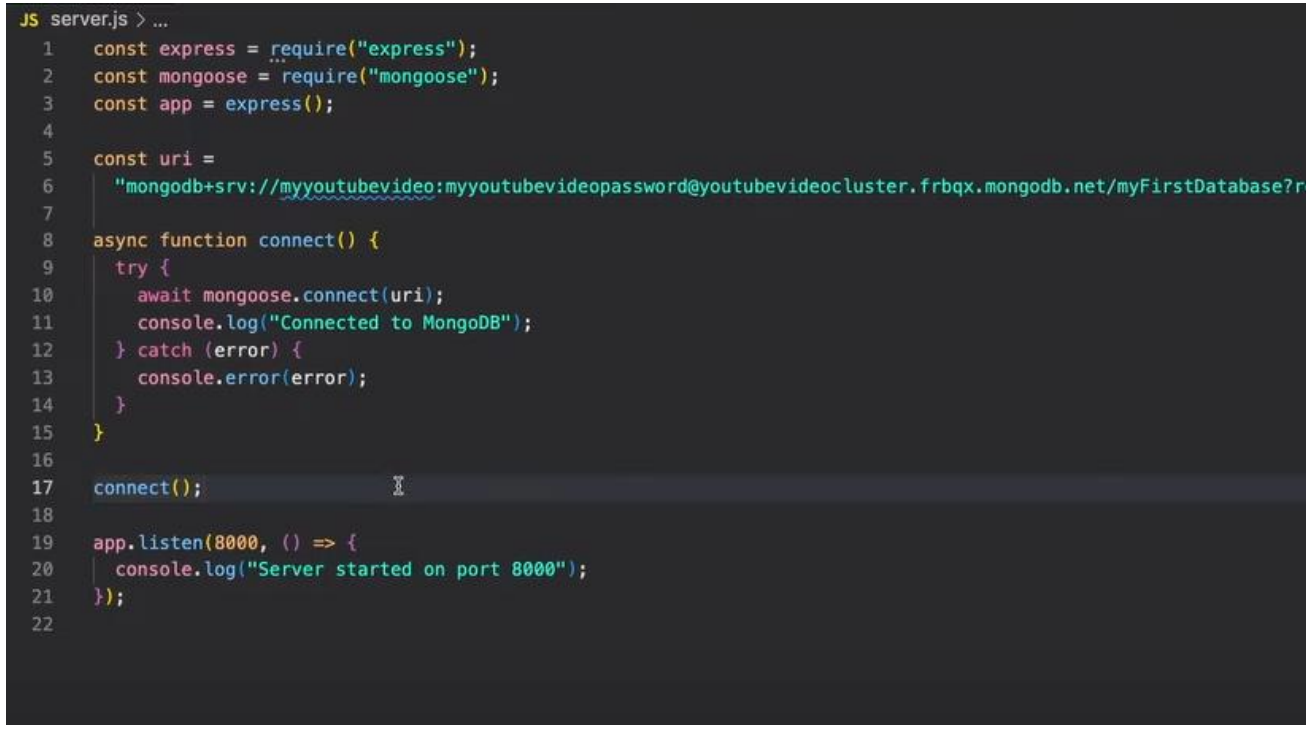
Database Name	Storage Size	Collections	Indexes
admin	20.5KB	0	1
config	24.6KB	0	2
local	36.9KB	1	1

MongoDB



MongoDB Shell

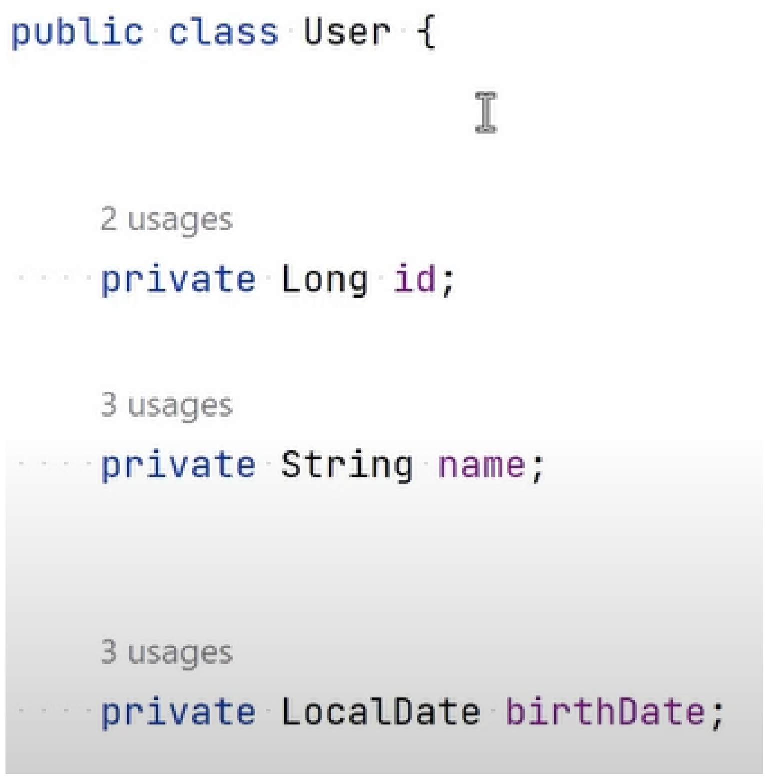
```
> show collections
posts
test
users
> db.users.find()
{ "_id" : ObjectId("5d97510a67474046a213c0a1"), "name" : "arin", "age" : 4, "email" : "arin@gmail.com", "gender" : "F" }
{ "_id" : ObjectId("5d9753a567474046a213c0a2"), "name" : "elis", "age" : 2, "email" : "elis@gmail.com", "gender" : "F" }
{ "_id" : ObjectId("5d9753a567474046a213c0a3"), "name" : "ahmet", "age" : 17, "email" : "ahmet@gmail.com", "gender" : "M" }
{ "_id" : ObjectId("5d97541e67474046a213c0a4"), "name" : "cem", "age" : 22, "email" : "cem@gmail.com", "gender" : "M" }
{ "_id" : ObjectId("5d97541e67474046a213c0a5"), "name" : "ceyda", "age" : 29, "email" : "ceyda@gmail.com", "gender" : "F" }
> db.users.findOne()
{
        "_id" : ObjectId("5d97510a67474046a213c0a1"),
        "name" : "arin",
        "age" : 4,
        "email" : "arin@gmail.com",
        "gender" : "F"
}
> _
```



JS server.js > ...

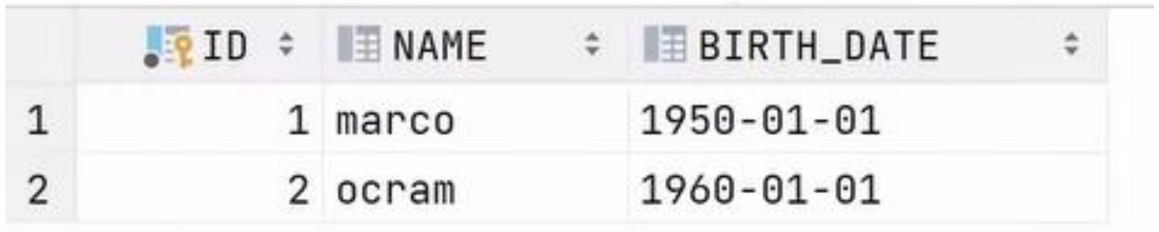
```js
const express = require("express");
const mongoose = require("mongoose");
const app = express();

const uri =
  "mongodb+srv://myyoutubevideo:myyoutubevideopassword@youtubevideocluster.frbqx.mongodb.net/myFirstDatabase?r

async function connect() {
  try {
    await mongoose.connect(uri);
    console.log("Connected to MongoDB");
  } catch (error) {
    console.error(error);
  }
}

connect();

app.listen(8000, () => {
  console.log("Server started on port 8000");
});

```

Object-Mapping Libraries



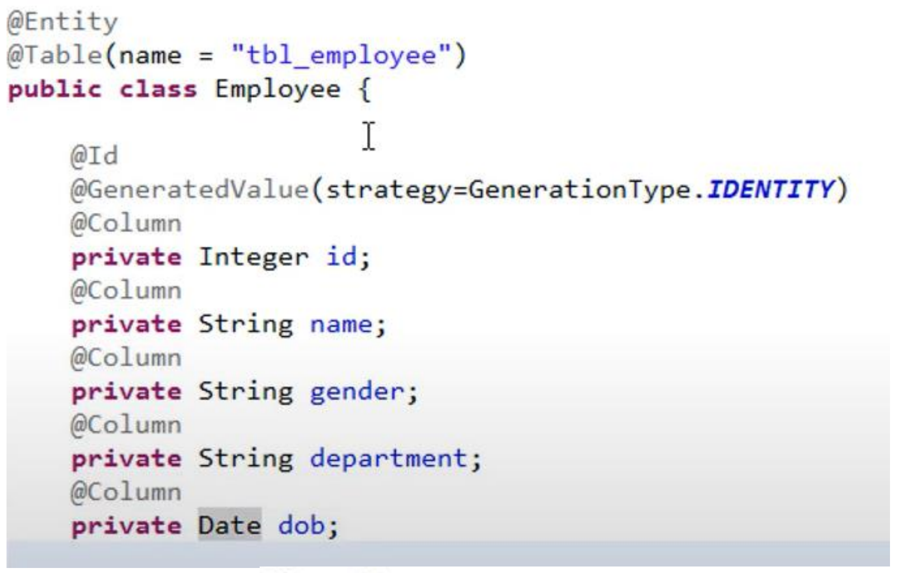
```
public class User {

    2 usages
    private Long id;

    3 usages
    private String name;

    3 usages
    private LocalDate birthDate;
```

- ORM stands for Object-Relational Mapping, which is a programming technique that maps objects in an object-oriented programming language to tables in a relational database
- Hibernate is an open-source object-relational mapping (ORM) tool for Java. It provides a framework for mapping Java objects to relational databases, allowing developers to work with Java objects rather than SQL queries

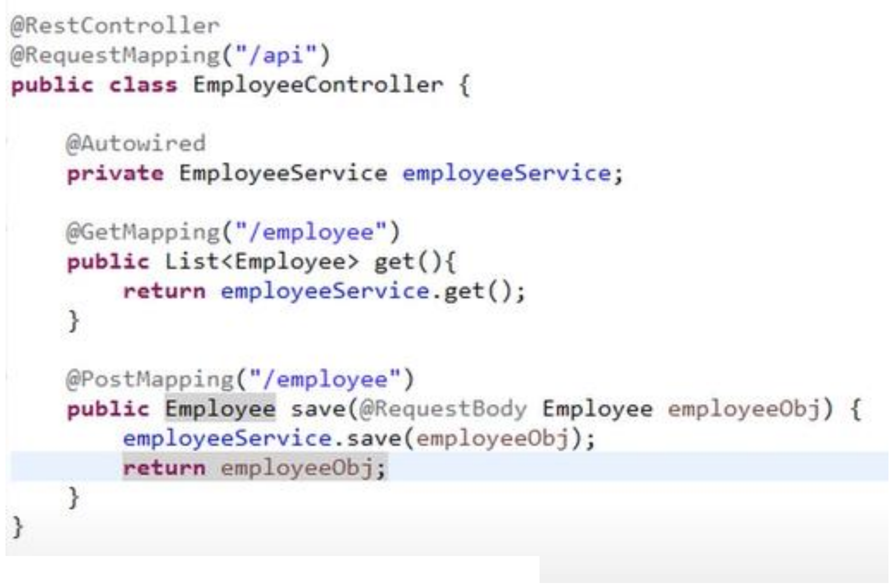


	ID	NAME	BIRTH_DATE
1	1	marco	1950-01-01
2	2	ocram	1960-01-01

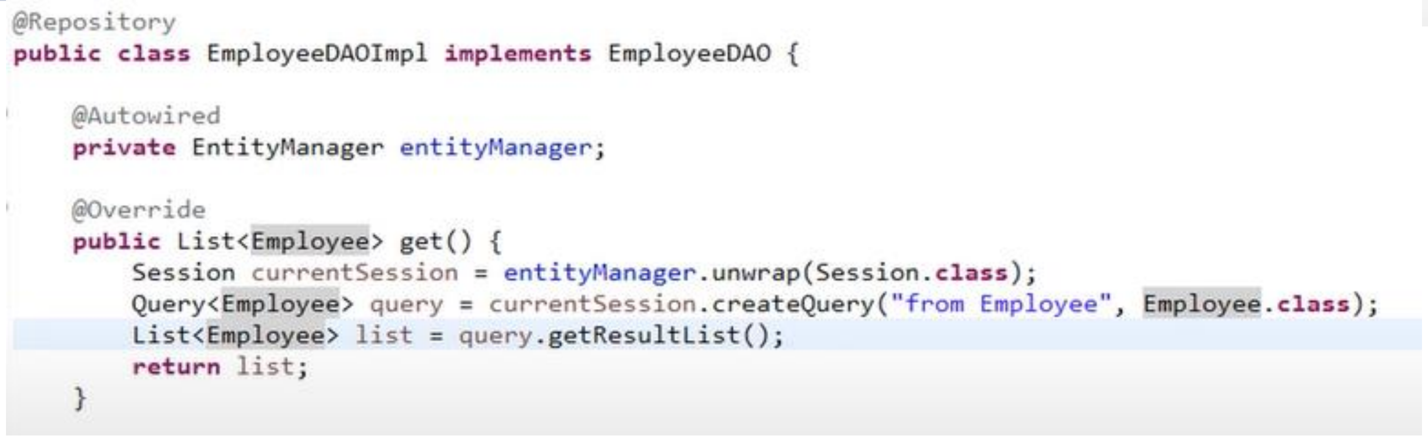
Object-Mapping Libraries



```java
@Entity
@Table(name = "tbl_employee")
public class Employee {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    @Column
    private Integer id;
    @Column
    private String name;
    @Column
    private String gender;
    @Column
    private String department;
    @Column
    private Date dob;
```



```java
@RestController
@RequestMapping("/api")
public class EmployeeController {

    @Autowired
    private EmployeeService employeeService;

    @GetMapping("/employee")
    public List<Employee> get(){
        return employeeService.get();
    }

    @PostMapping("/employee")
    public Employee save(@RequestBody Employee employeeObj) {
        employeeService.save(employeeObj);
        return employeeObj;
    }
}
```



```java
@Repository
public class EmployeeDAOImpl implements EmployeeDAO {

    @Autowired
    private EntityManager entityManager;

    @Override
    public List<Employee> get() {
        Session currentSession = entityManager.unwrap(Session.class);
        Query<Employee> query = currentSession.createQuery("from Employee", Employee.class);
        List<Employee> list = query.getResultList();
        return list;
    }
```