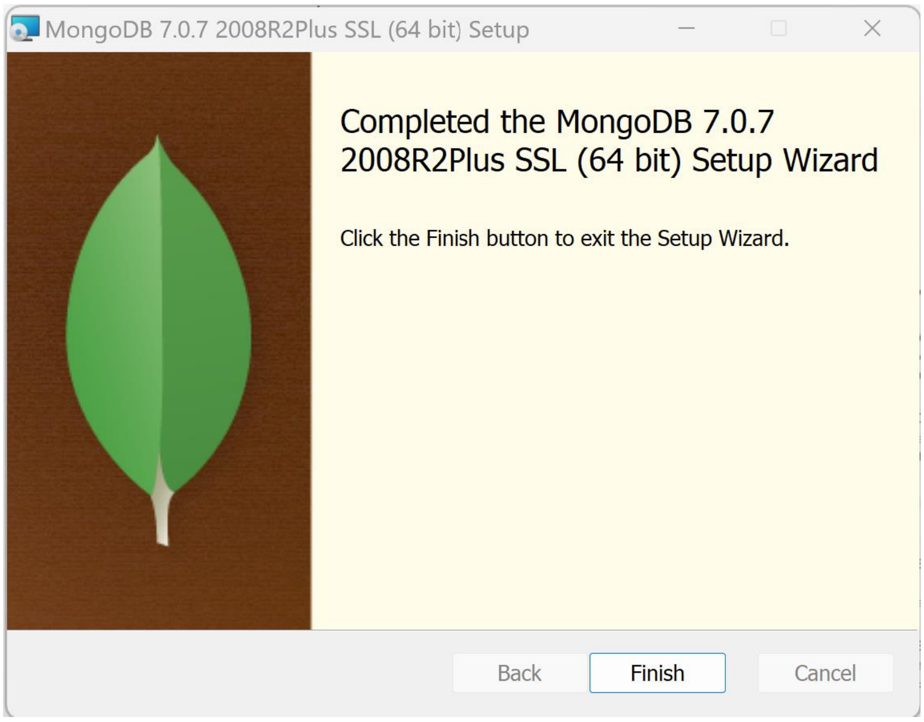
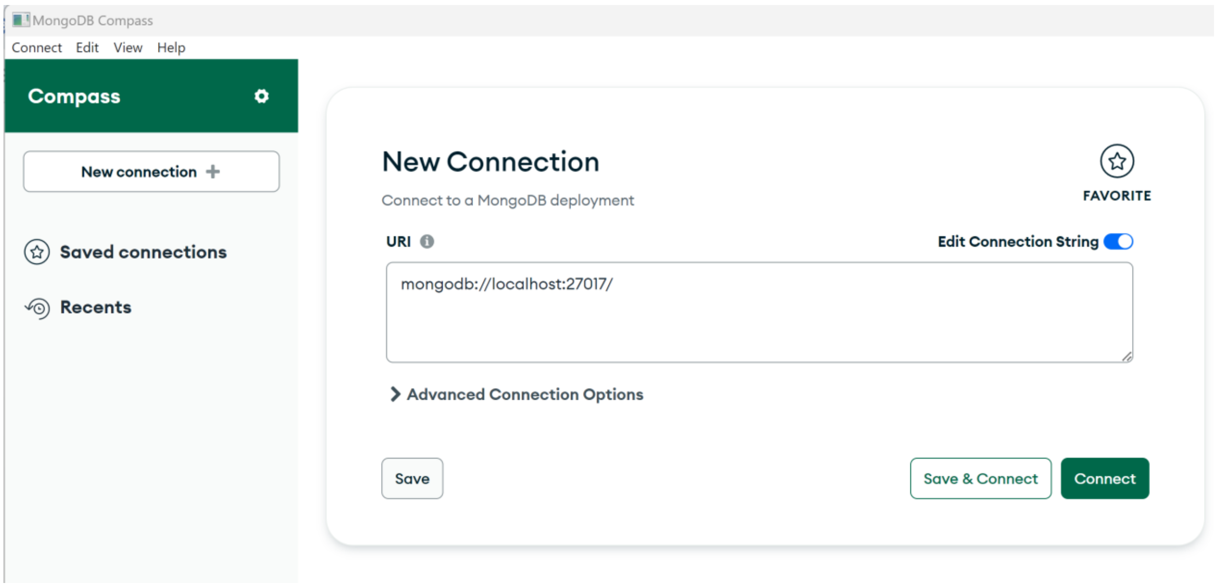
Install MongoDB


MongoDB 7.0.7 2008R2Plus SSL (64 bit) Setup

Completed the MongoDB 7.0.7 2008R2Plus SSL (64 bit) Setup Wizard

Click the Finish button to exit the Setup Wizard.

MongoDB Compass Explorer



New Connection

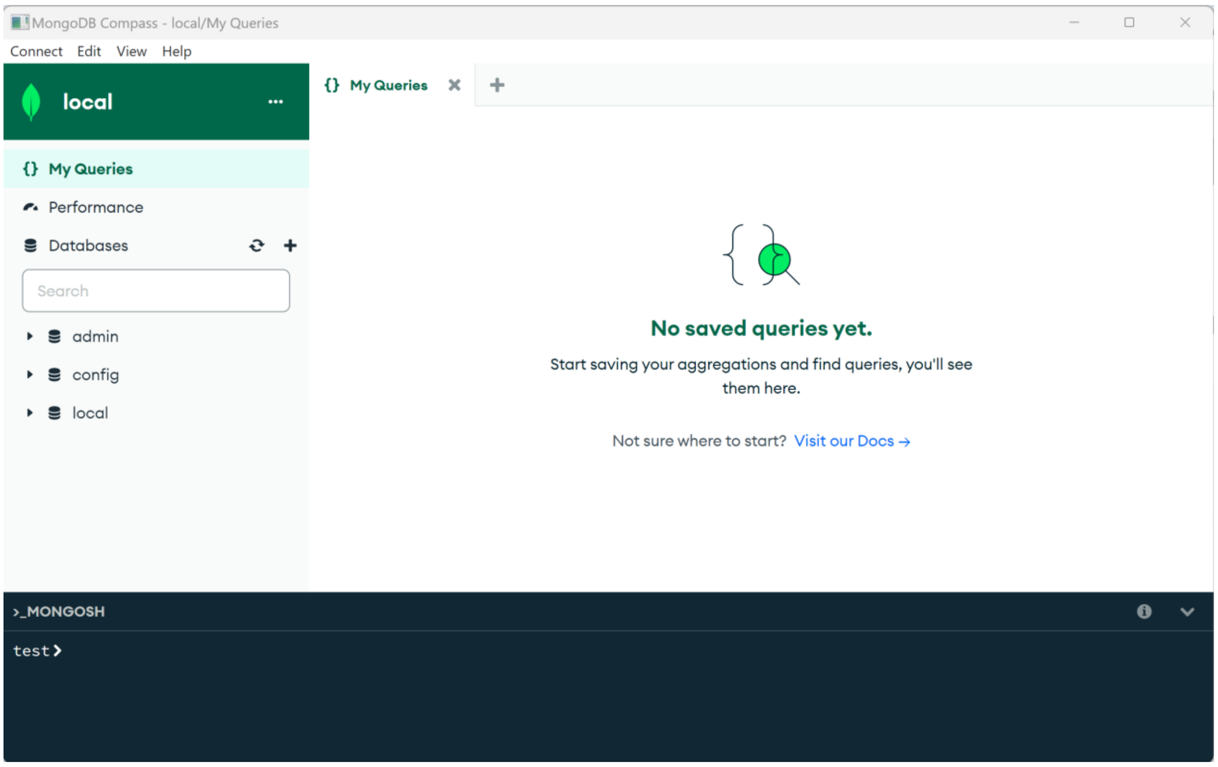
Connect to a MongoDB deployment

URI

Edit Connection String

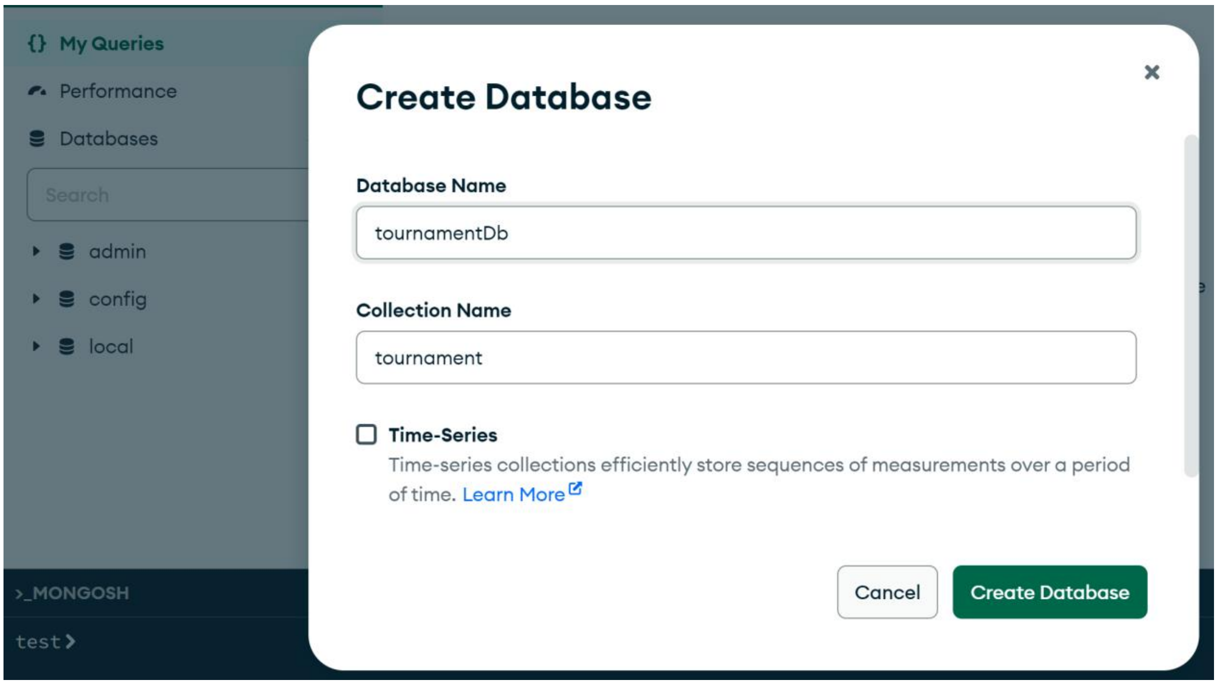
```
mongodb://localhost:27017/
```

Save connection name = « local »


MongoDB Compass - local/My Queries
Connect Edit View Help
local
My Queries
Performance
Databases
Search
admin
config
local
My Queries
No saved queries yet.
Start saving your aggregations and find queries, you'll see them here.
Not sure where to start? Visit our Docs →
>_MONGOSH

```
test>
```

Create DB « tournamentDb », collection : « tournament »



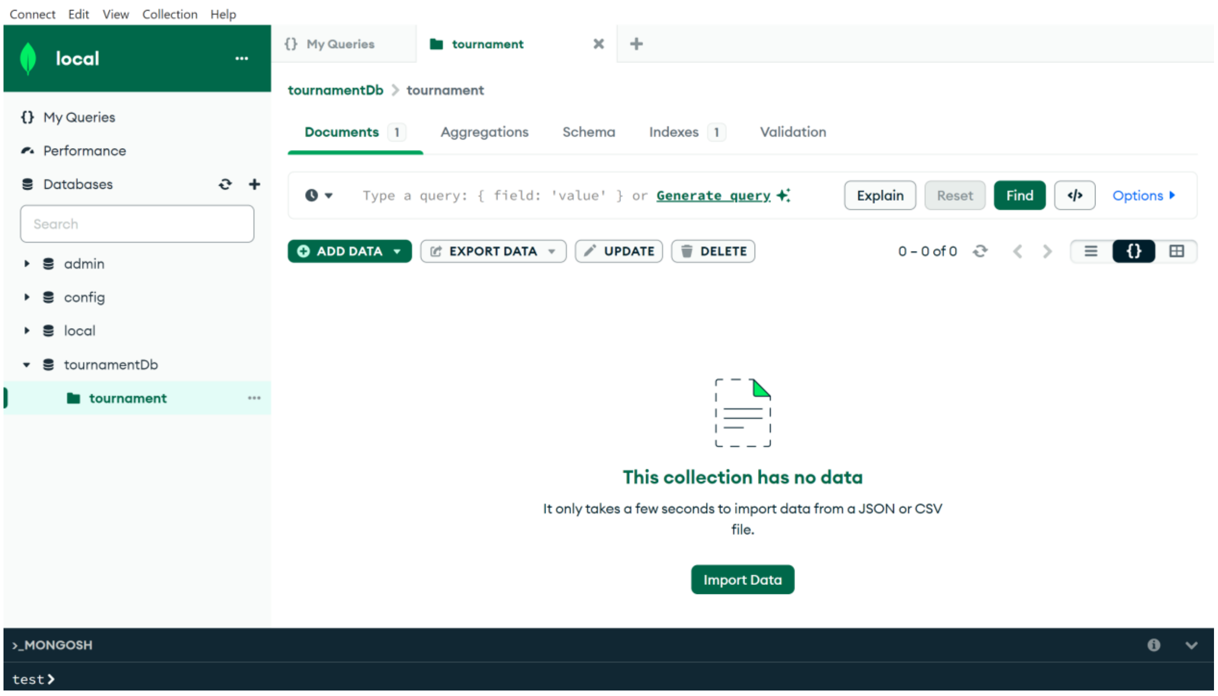
Create Database

Database Name: tournamentDb

Collection Name: tournament

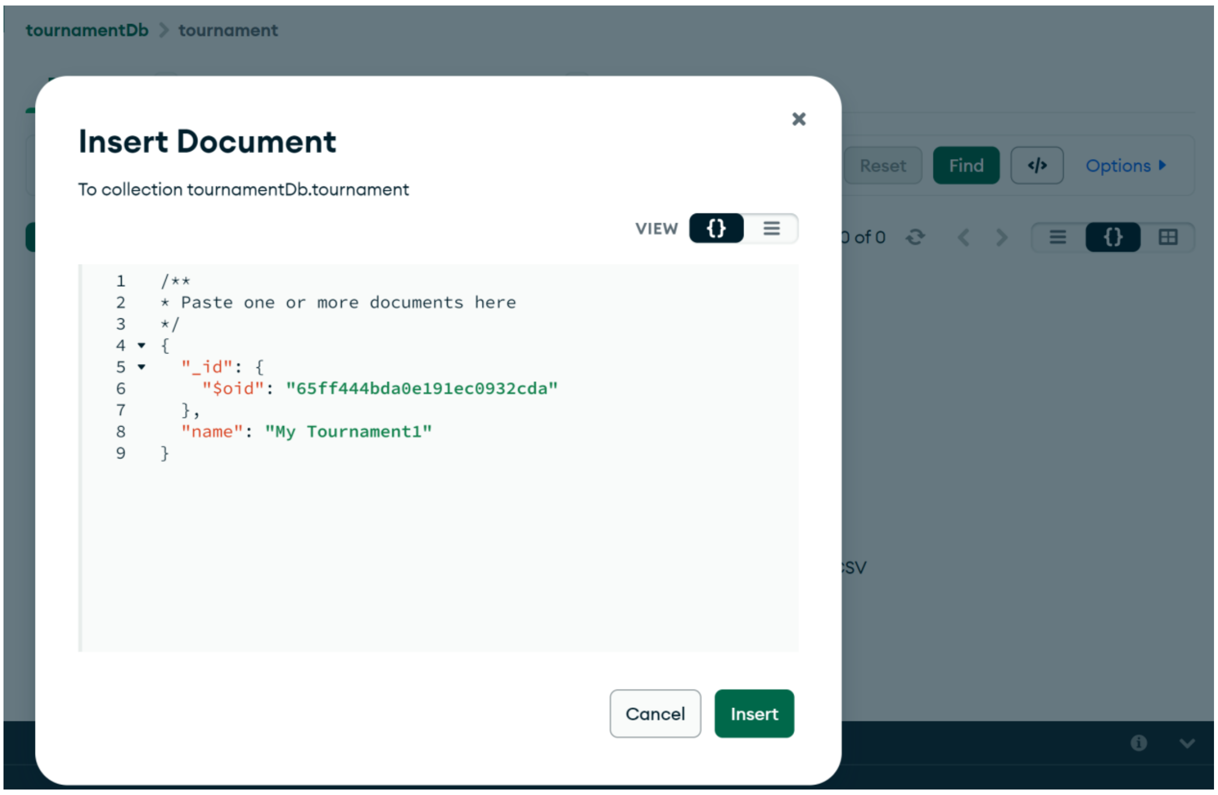
☐ Time-Series
Time-series collections efficiently store sequences of measurements over a period of time. Learn More

List (empty) document in collection



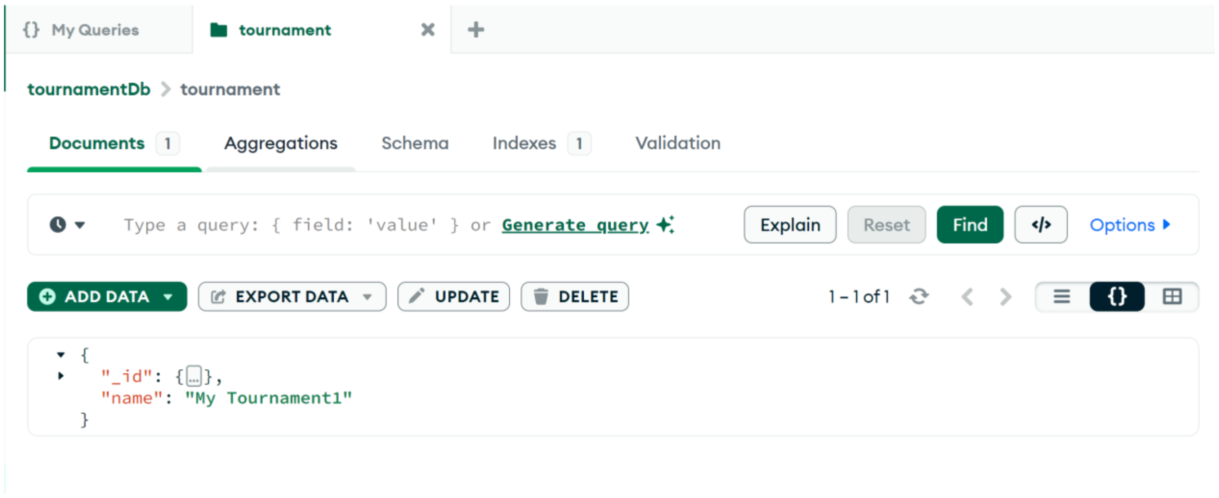
```
test>
```

Create document « tournament », «name » : « My Tournament1 »


tournamentDb > tournament
Insert Document
To collection tournamentDb.tournament
VIEW
Cancel
Insert
Reset
Find
Options

```
/**
* Paste one or more documents here
*/
{
  "_id": {
    "$oid": "65ff444bda0e191ec0932cda"
  },
  "name": "My Tournament1"
}
```

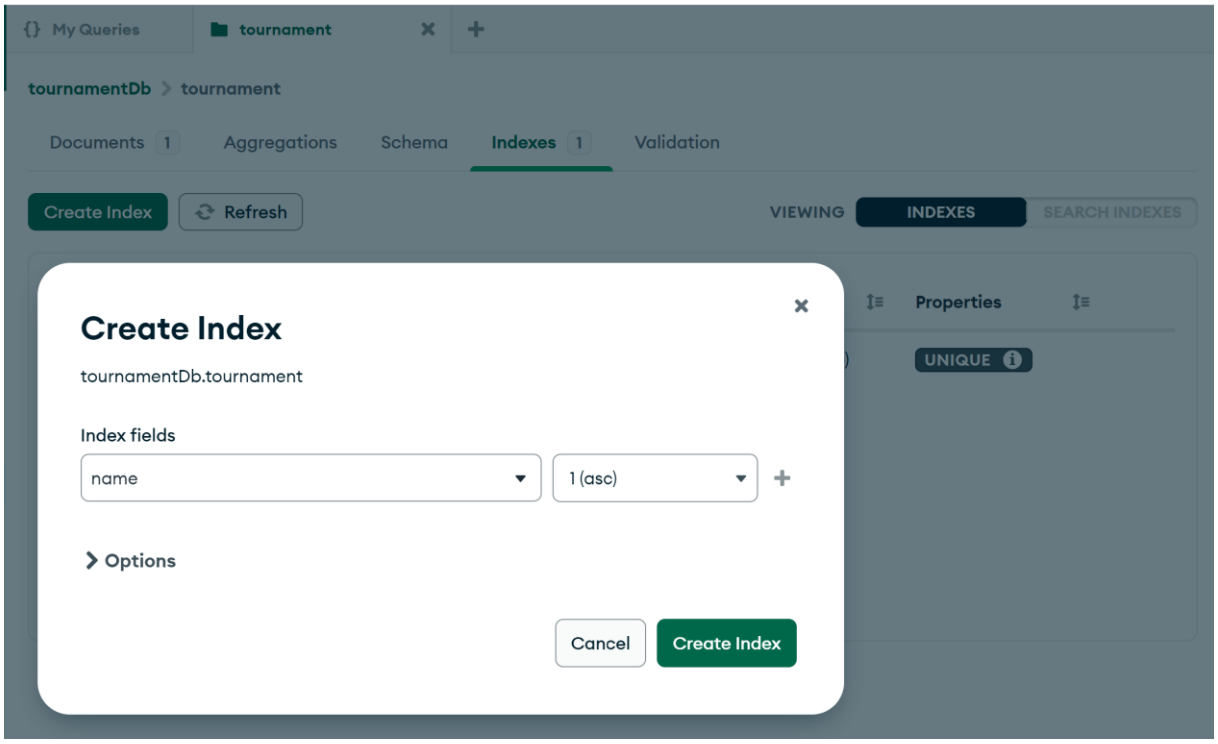
List document(s)



```
{
  "_id": {…},
  "name": "My Tournament1"
}
```

Index

Create Index by « name »



Create Index

tournamentDb.tournament

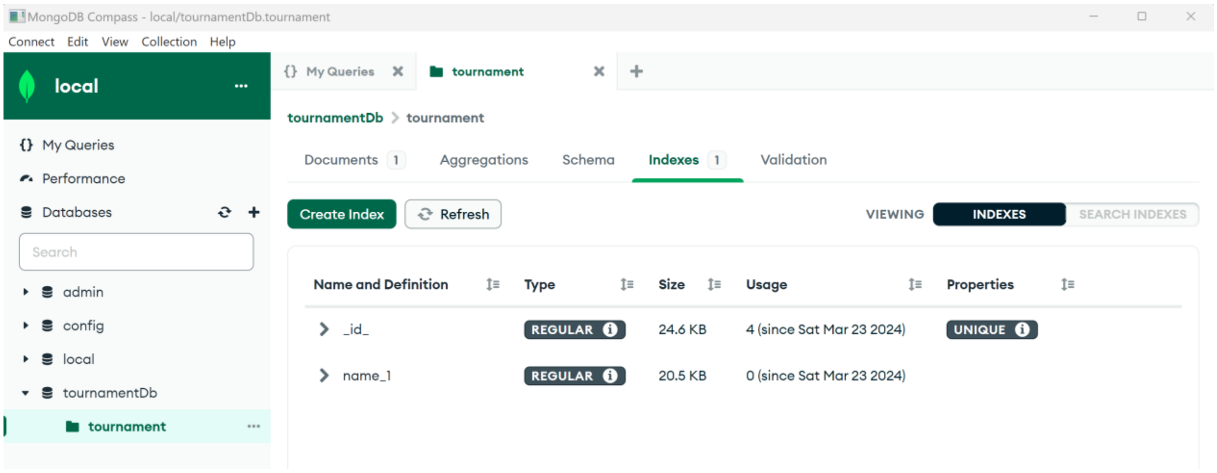
Index fields

name | 1 (asc)

Options

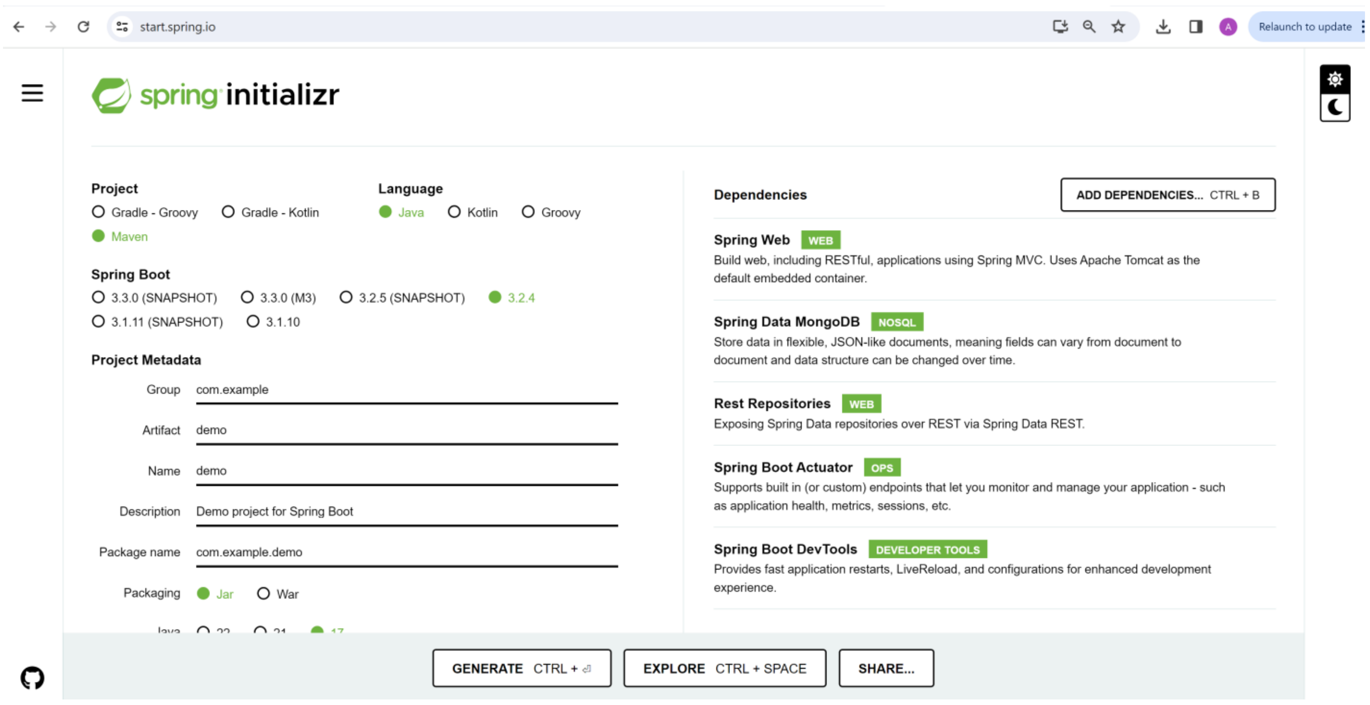
Cancel | Create Index

List Indexes (by « id » + by « name»)



Name and Definition	Type	Size	Usage	Properties
_id_	REGULAR	24.6 KB	4 (since Sat Mar 23 2024)	UNIQUE
name_1	REGULAR	20.5 KB	0 (since Sat Mar 23 2024)	

SpringInitializr


start.spring.io

spring initializr

Project
Gradle - Groovy Gradle - Kotlin [✓ Maven]

Language
[✓ Java] Kotlin Groovy

Spring Boot
3.3.0 (SNAPSHOT) 3.3.0 (M3) 3.2.5 (SNAPSHOT) [✓ 3.2.4]
3.1.11 (SNAPSHOT) 3.1.10

Project Metadata

Group	com.example
Artifact	demo
Name	demo
Description	Demo project for Spring Boot
Package name	com.example.demo
Packaging	[✓ Jar] War
Java	22 21 [✓ 17]

Dependencies ADD DEPENDENCIES... CTRL + B

Spring Web WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

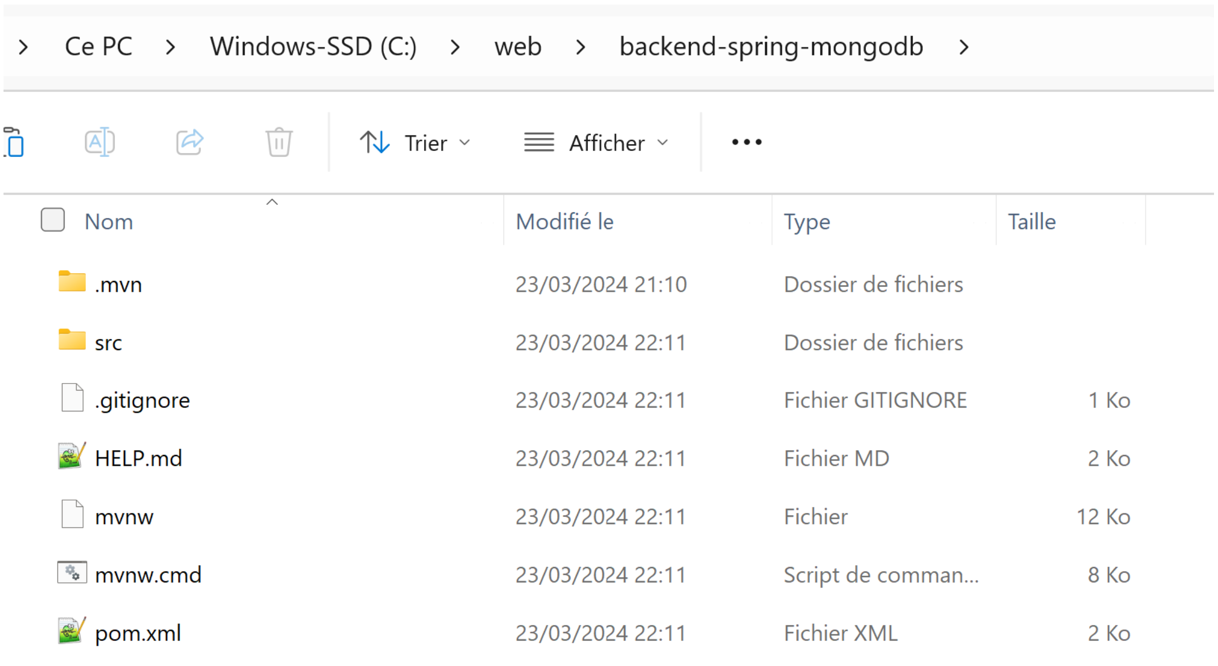
Spring Data MongoDB NOSQL
Store data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time.

Rest Repositories WEB
Exposing Spring Data repositories over REST via Spring Data REST.

Spring Boot Actuator OPS
Supports built in (or custom) endpoints that let you monitor and manage your application - such as application health, metrics, sessions, etc.

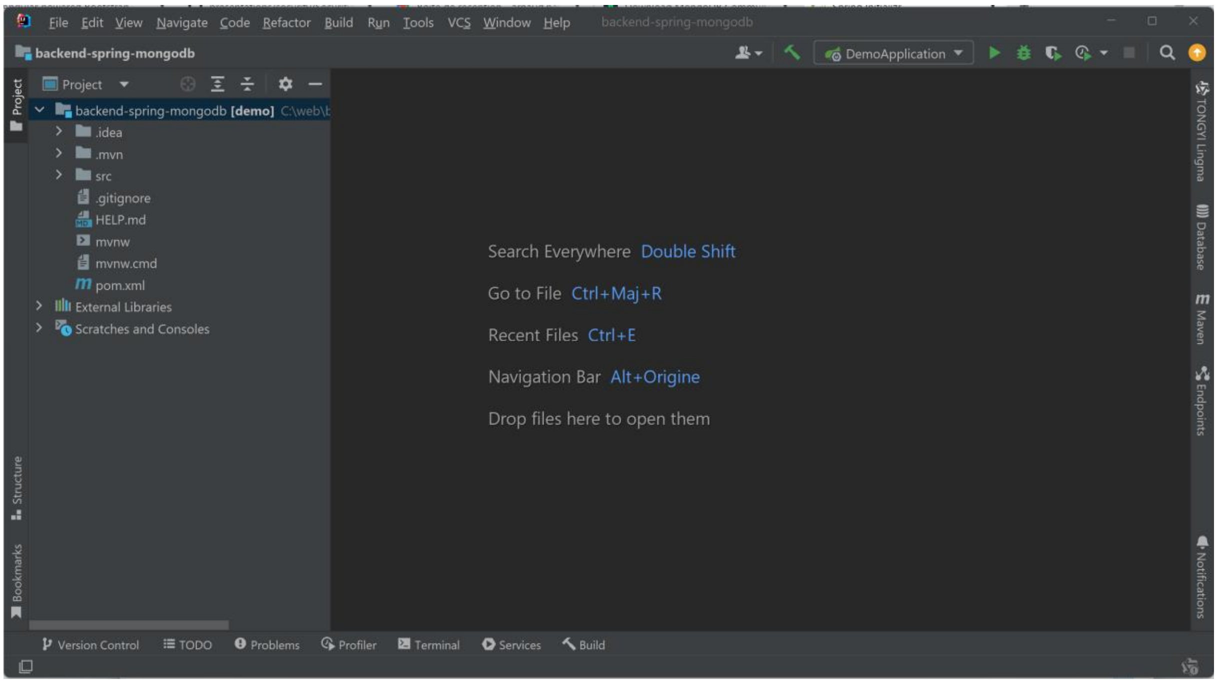
Spring Boot DevTools DEVELOPER TOOLS
Provides fast application restarts, LiveReload, and configurations for enhanced development experience.

Click “Generate” (Download zip), then unzip (and rename dir “backend-spring-mondb”)


> Ce PC > Windows-SSD (C:) > web > backend-spring-mongodb >

Nom	Modifié le	Type	Taille
.mvn	23/03/2024 21:10	Dossier de fichiers	
src	23/03/2024 22:11	Dossier de fichiers	
.gitignore	23/03/2024 22:11	Fichier GITIGNORE	1 Ko
HELP.md	23/03/2024 22:11	Fichier MD	2 Ko
mvnw	23/03/2024 22:11	Fichier	12 Ko
mvnw.cmd	23/03/2024 22:11	Script de comman...	8 Ko
pom.xml	23/03/2024 22:11	Fichier XML	2 Ko

Import as Maven project in IntelliJ



```
backend-spring-mongodb [demo] C:\web\b
  .idea
  .mvn
  src
  .gitignore
  HELP.md
  mvnw
  mvnw.cmd
  pom.xml
External Libraries
Scratches and Consoles
```

Search Everywhere Double Shift

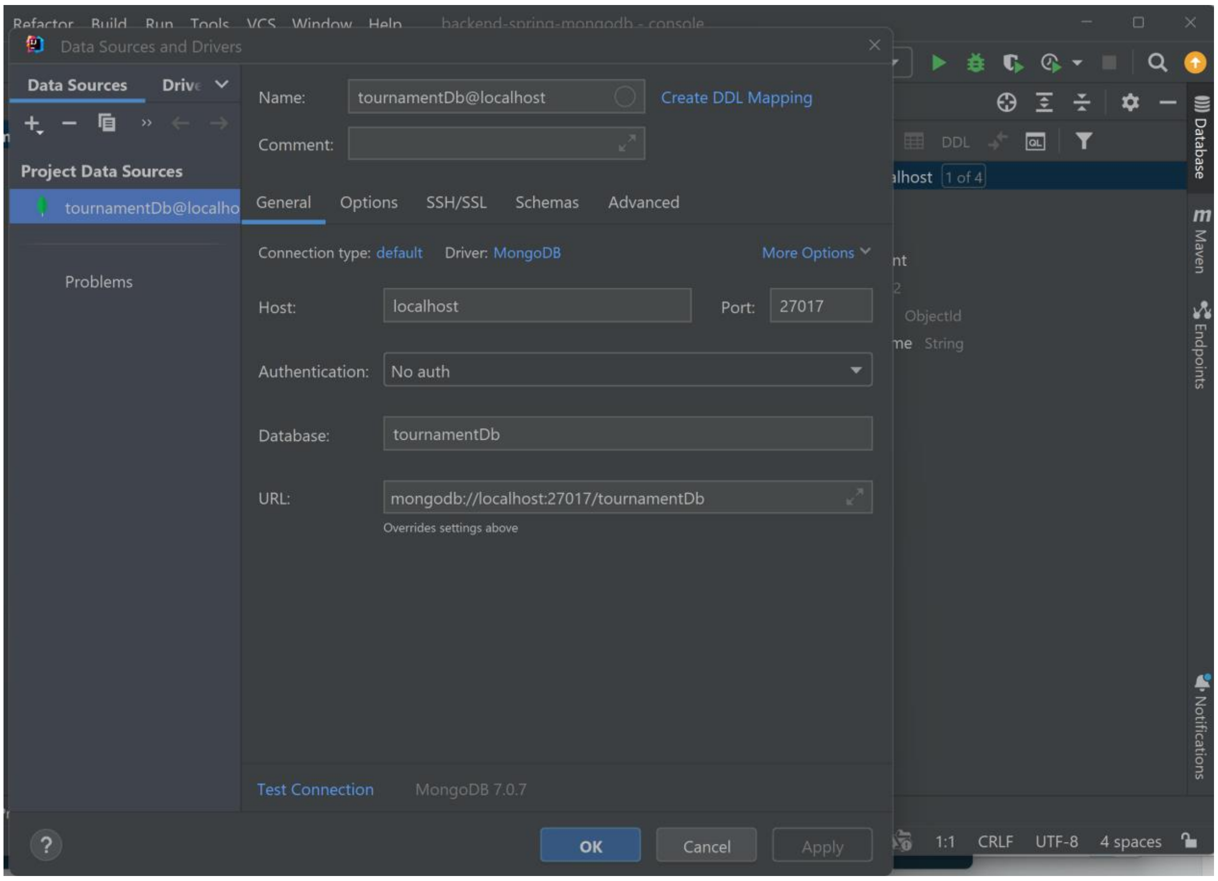
Go to File Ctrl+Maj+R

Recent Files Ctrl+E

Navigation Bar Alt+Origine

Drop files here to open them

IntelliJ Database Explorer



Data source: tournamentDb@localhost

Name: tournamentDb@localhost

Comment:

Connection type: default Driver: MongoDB

Host: localhost Port: 27017

Authentication: No auth

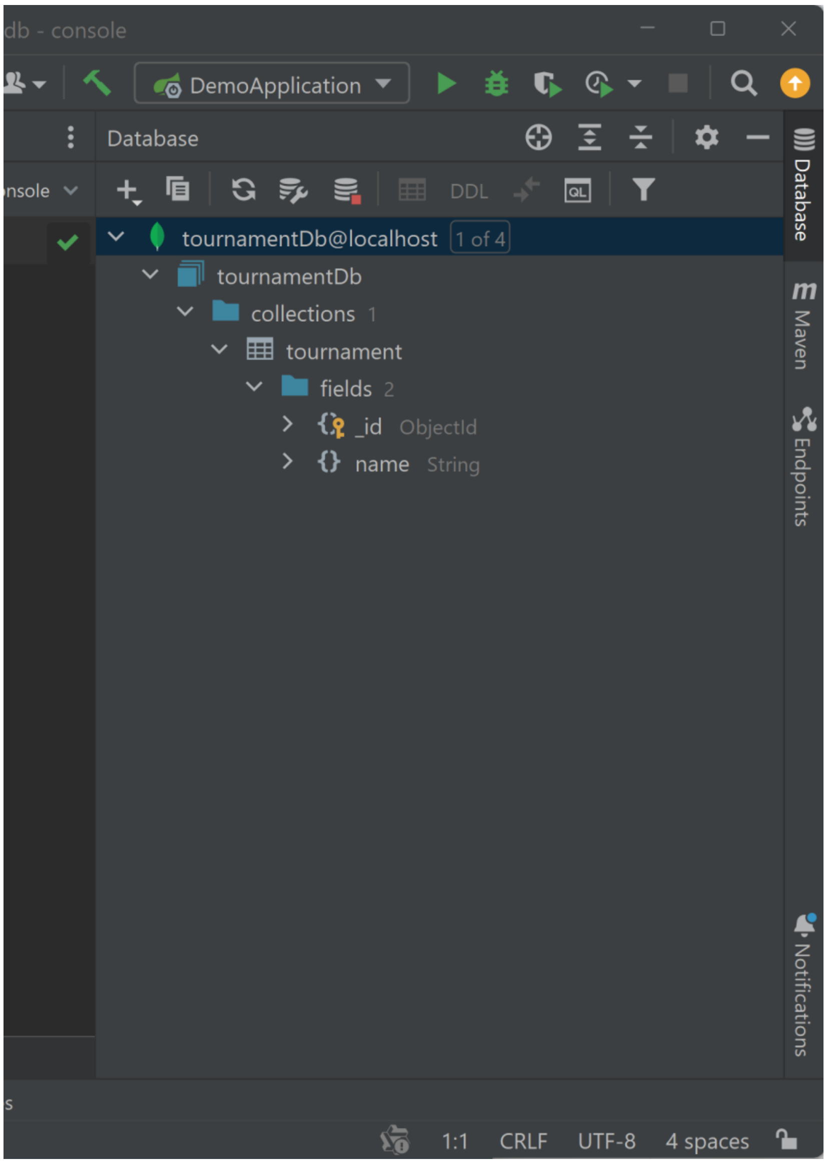
Database: tournamentDb

URL: mongodb://localhost:27017/tournamentDb

Overrides settings above

MongoDB 7.0.7

Database -> Collection -> schema visible in IntelliJ



```
tournamentDb@localhost 1 of 4
  tournamentDb
    collections 1
      tournament
        fields 2
          _id ObjectId
          name String
```

1:1 CRLF UTF-8 4 spaces

Write Entity class

```java
package com.example.demo.domain;

import org.springframework.data.annotation.Id;
import org.springframework.data.mongodb.core.mapping.Document;

import java.time.LocalDate;

@Document
public class Tournament {

    @Id
    public String id;

    public String name;

    public LocalDate startDate;
    public LocalDate endDate;

}
```

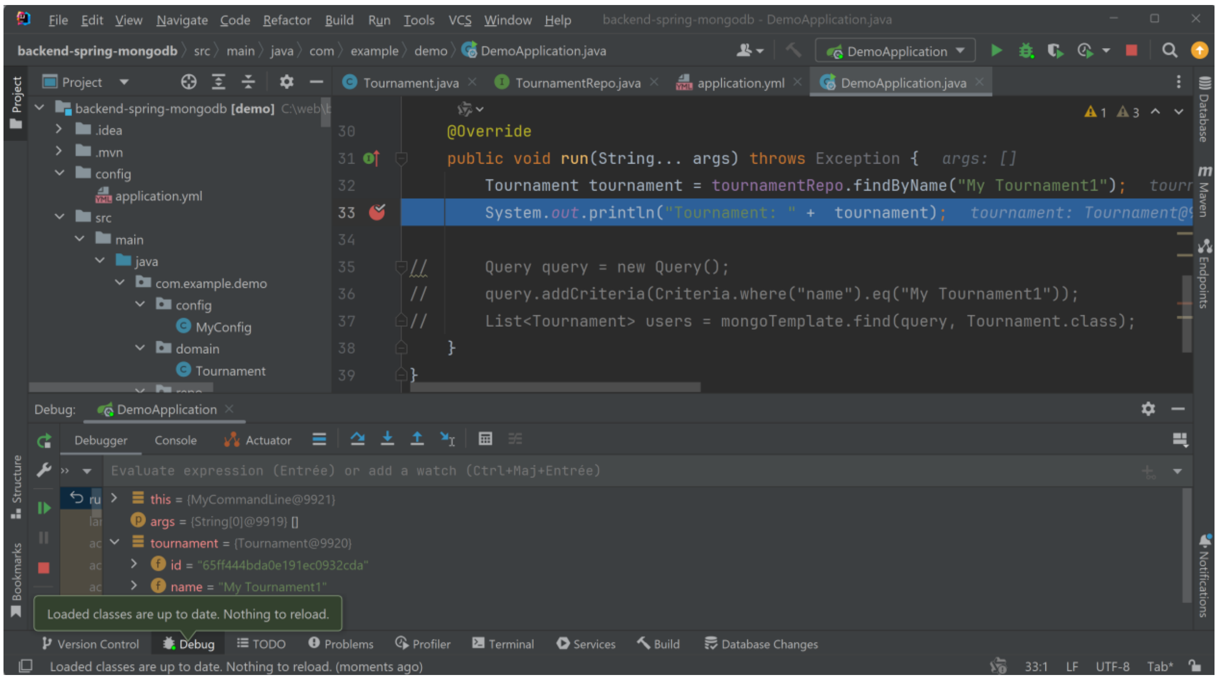
Write Repository (= CRUD)

```java
package com.example.demo.repo;

import com.example.demo.domain.Tournament;
import org.springframework.data.mongodb.repository.MongoRepository;

public interface TournamentRepo extends MongoRepository<Tournament,String>
{

    Tournament findByName(String name);

}
```

Adding config/application.yml

```yaml

spring:
  data:
    mongodb:
      host: localhost
      port: 27017
      database: tournamentDb
```

Launching App (Debug)



```java
    @Override
    public void run(String... args) throws Exception {   args: []
        Tournament tournament = tournamentRepo.findByName("My Tournament1");   tourr
        System.out.println("Tournament: " +  tournament);   tournament: Tournament@9

//      Query query = new Query();
//      query.addCriteria(Criteria.where("name").eq("My Tournament1"));
//      List<Tournament> users = mongoTemplate.find(query, Tournament.class);
    }
}
```

```
this = {MyCommandLine@9921}
args = {String[0]@9919} []
tournament = {Tournament@9920}
    id = "65ff444bda0e191ec0932cda"
    name = "My Tournament1"
```

Loaded classes are up to date. Nothing to reload.

Loaded classes are up to date. Nothing to reload. (moments ago)

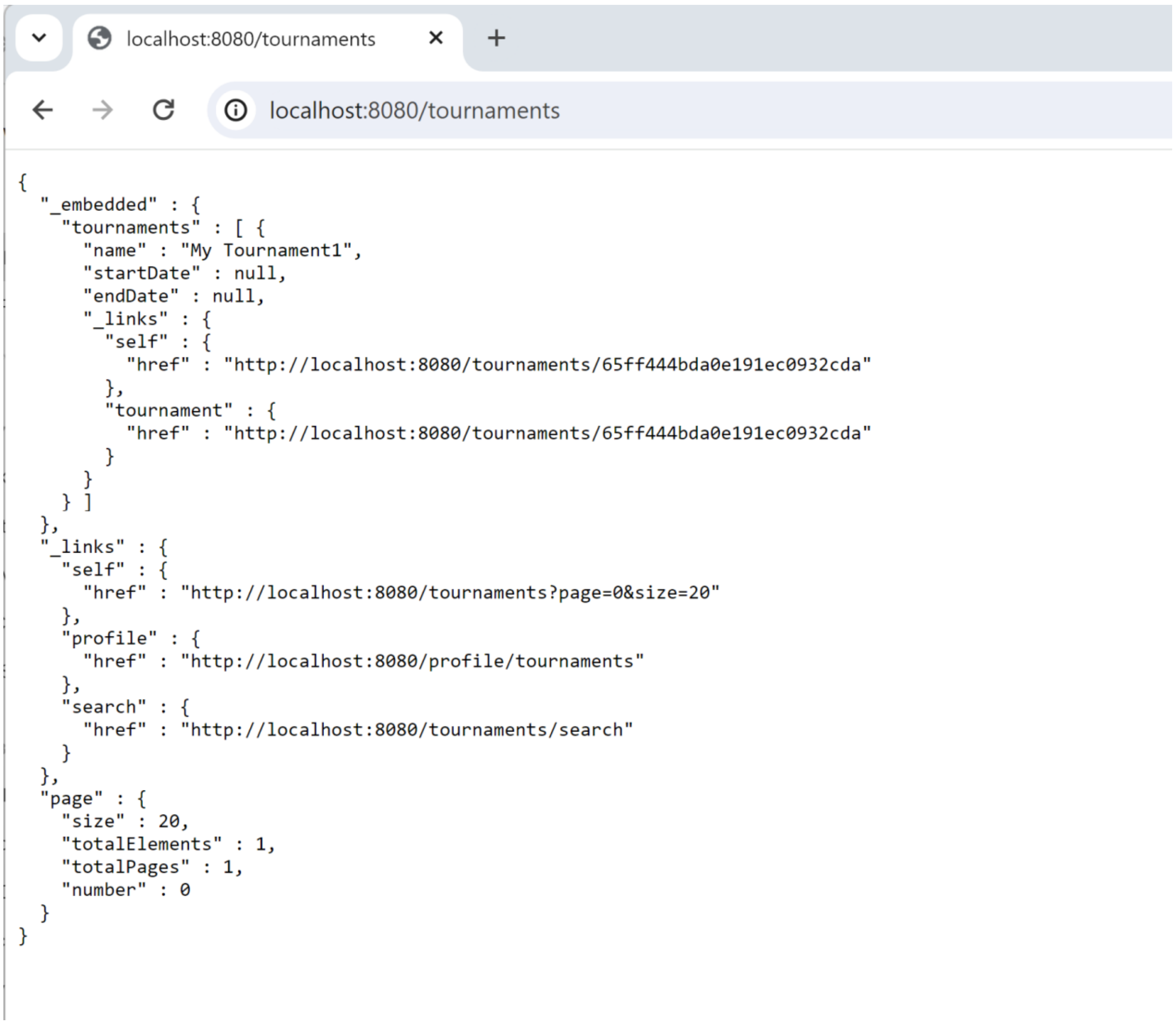
Adding default CRUD Rest Api (not recommended, mostly for debug)

Edit repository class, add

```
import
org.springframework.data.rest.core.annotation.RepositoryRestResource;

@RepositoryRestResource(path = "tournaments")
```

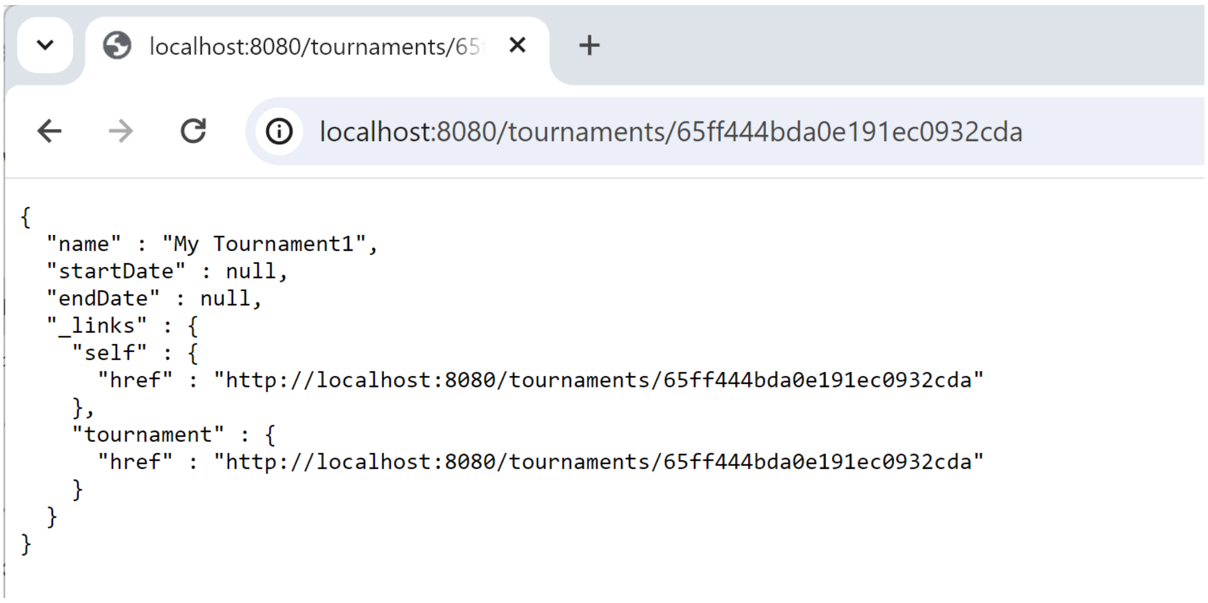
Relaunch server

Test find all “GET /tournaments” in Browser



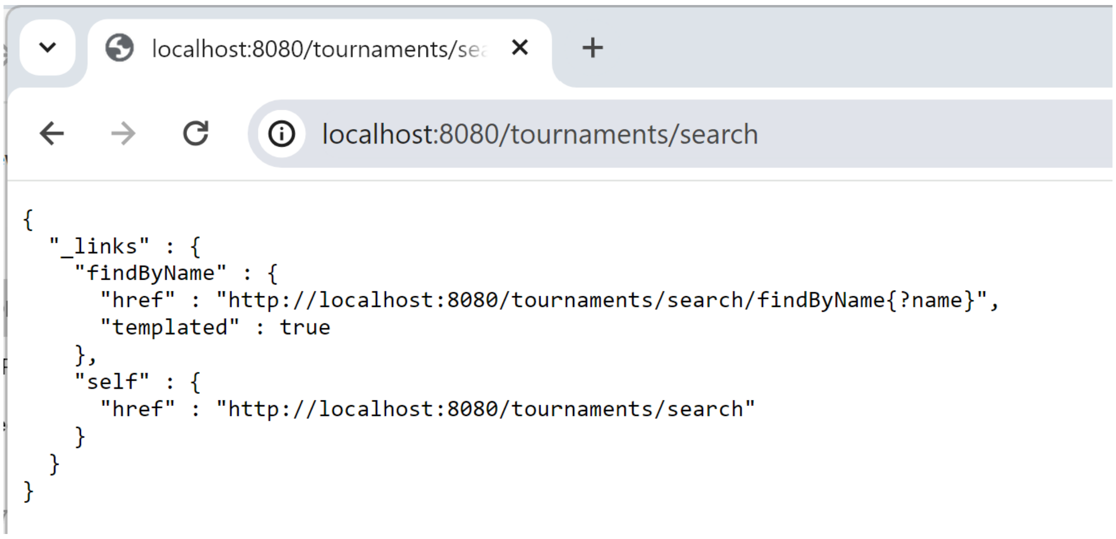
```
{
  "_embedded" : {
    "tournaments" : [ {
      "name" : "My Tournament1",
      "startDate" : null,
      "endDate" : null,
      "_links" : {
        "self" : {
          "href" : "http://localhost:8080/tournaments/65ff444bda0e191ec0932cda"
        },
        "tournament" : {
          "href" : "http://localhost:8080/tournaments/65ff444bda0e191ec0932cda"
        }
      }
    } ]
  },
  "_links" : {
    "self" : {
      "href" : "http://localhost:8080/tournaments?page=0&size=20"
    },
    "profile" : {
      "href" : "http://localhost:8080/profile/tournaments"
    },
    "search" : {
      "href" : "http://localhost:8080/tournaments/search"
    }
  },
  "page" : {
    "size" : 20,
    "totalElements" : 1,
    "totalPages" : 1,
    "number" : 0
  }
}
```

Test find By Id : “GET /tournaments/65ff444bda0e191ec0932cda“


localhost:8080/tournaments/65
localhost:8080/tournaments/65ff444bda0e191ec0932cda

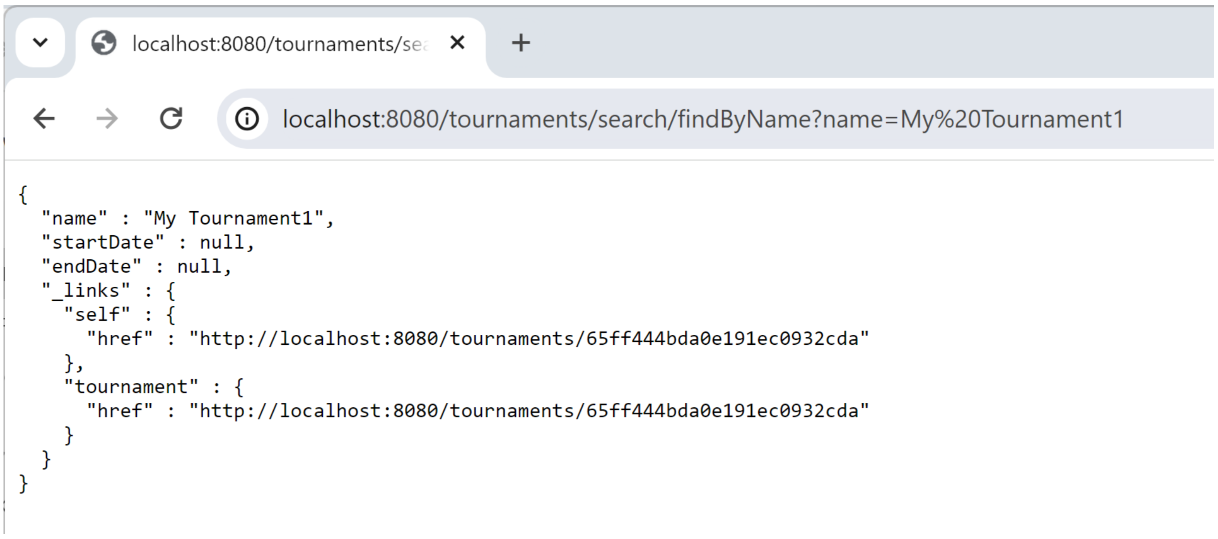
```
{
  "name" : "My Tournament1",
  "startDate" : null,
  "endDate" : null,
  "_links" : {
    "self" : {
      "href" : "http://localhost:8080/tournaments/65ff444bda0e191ec0932cda"
    },
    "tournament" : {
      "href" : "http://localhost:8080/tournaments/65ff444bda0e191ec0932cda"
    }
  }
}
```

All Searches from repository queries


localhost:8080/tournaments/se
localhost:8080/tournaments/search

```
{
  "_links" : {
    "findByName" : {
      "href" : "http://localhost:8080/tournaments/search/findByName{?name}",
      "templated" : true
    },
    "self" : {
      "href" : "http://localhost:8080/tournaments/search"
    }
  }
}
```

Search by name


localhost:8080/tournaments/se
localhost:8080/tournaments/search/findByName?name=My%20Tournament1

```
{
  "name" : "My Tournament1",
  "startDate" : null,
  "endDate" : null,
  "_links" : {
    "self" : {
      "href" : "http://localhost:8080/tournaments/65ff444bda0e191ec0932cda"
    },
    "tournament" : {
      "href" : "http://localhost:8080/tournaments/65ff444bda0e191ec0932cda"
    }
  }
}
```

Custom Rest API (business API code)

Internal rules for code:

1/ all rest controller methods should have 1 line of code, by delegating to a @Service class

2/ PUT and POST rest controller methods should have 1 @RequestBody param using a “RequestDTO” class, and return a “ResponseDTO” class

3/ all service methods should consist of 3 steps:

```
// step 1/3: unmarshall, check inputs
…

// step 2/3: business code
…

// step 3/3: marshall output (DTO, not internal entity)
…
```

Create a Rest controller

```
package com.example.demo.rest;

import com.example.demo.rest.dto.TournamentCreateRequestDTO;
import com.example.demo.rest.dto.TournamentCreateResponseDTO;
import com.example.demo.service.TournamentService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
@RequestMapping("/api/tournament")
public class TournamentRestController {

    @Autowired
    private TournamentService tournamentService;

    @PostMapping()
    public TournamentCreateResponseDTO createTournament(
            @RequestBody TournamentCreateRequestDTO req) {
        return tournamentService.createTournament(req);
    }
}
```

Create the corresponding Service class

```
package com.example.demo.service;

import com.example.demo.domain.Tournament;
import com.example.demo.repo.TournamentRepo;
import com.example.demo.rest.dto.TournamentCreateRequestDTO;
```

```java
import com.example.demo.rest.dto.TournamentCreateResponseDTO;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import org.springframework.transaction.annotation.Transactional;

import java.time.LocalDate;
import java.time.LocalDateTime;
import java.util.Objects;

@Service
@Transactional
public class TournamentService {

    @Autowired
    private TournamentRepo tournamentRepo;

    public TournamentCreateResponseDTO createTournament(
            TournamentCreateRequestDTO req) {
        // step 1/3: unmarshall, check inputs
        String name = Objects.requireNonNull(req.name);
        if (name.length() < 3) {
            throw new IllegalArgumentException("name too short");
        }
        Tournament alreadyFound = tournamentRepo.findByName(name);
        if (alreadyFound != null) {
            throw new IllegalArgumentException("name already used");
        }

        // step 2/3: business code
        // OK, create
        Tournament entity = new Tournament();
        entity.name = name;
        entity.createdAt = LocalDateTime.now();
        entity.createdBy = "<<currentUser>>"; // security not impl yet
        entity = tournamentRepo.save(entity);

        // step 3/3: marshall output (DTO, not internal entity)
        TournamentCreateResponseDTO res = toDto(entity);
        return res;
    }

    private TournamentCreateResponseDTO toDto(Tournament src) {
        TournamentCreateResponseDTO res =
                new TournamentCreateResponseDTO(
                        src.id, src.name, src.createdAt, src.createdBy);
        return res;
    }
}
```

Test using http client : Curl

```
$ curl -H "content-type: application/json" http://localhost:8080/api/tournament -d '{"name":"super to urnament"}'
```

```
{"id":"65ff5a96e9a12719c3847643","name":"super tournament","createdDate":"2024-03-23T23:41:26.9612767","createdBy":"<<currentUser>>"}
```

Relaunch test … assume document NOT inserted twice with same name

```
$ curl -H "content-type: application/json" http://localhost:8080/api/tournament -d '{"name":"super to urnament"}'
```

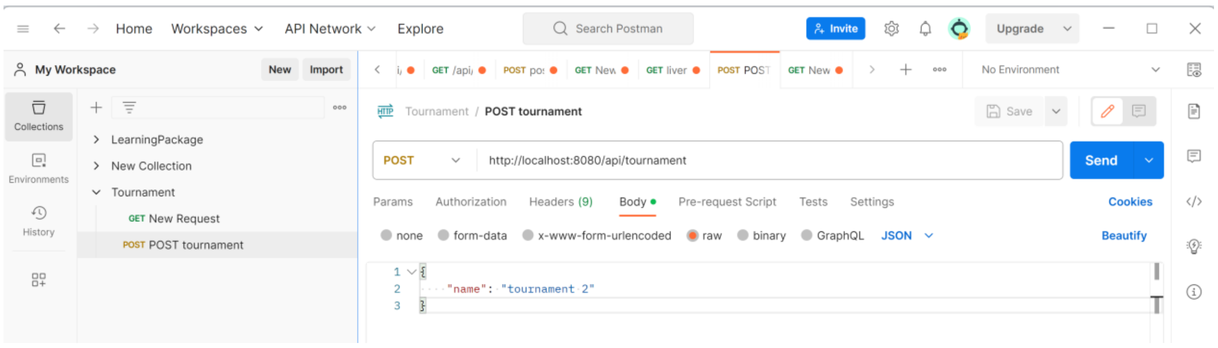
```
{"timestamp":"2024-03-23T22:43:09.970+00:00","status":500,"error":"Internal Server Error","trace":"java.lang.IllegalArgumentException: name already used\r\n\tat com.example.demo.service.TournamentService.createTournament(TournamentService.java:31)\r\n\t at …
```

Test using Postman

Create "Tournament" rest collection

Then "add request"

Edit, to use POST, url, body with type json

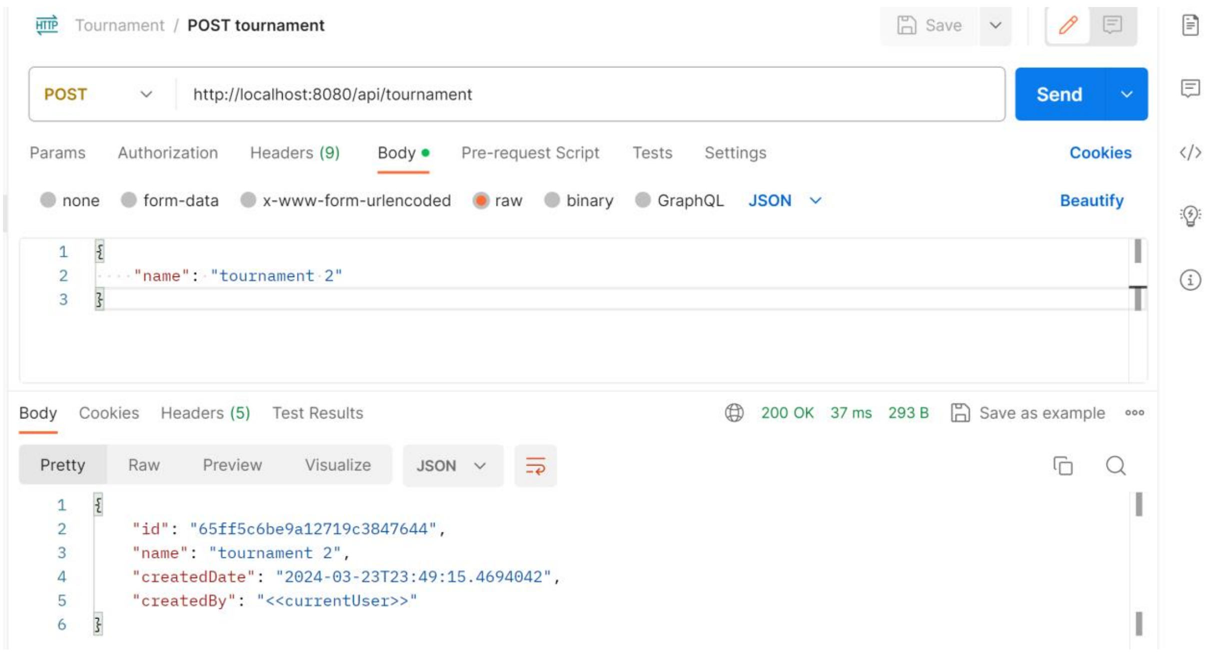


Tournament / POST tournament

POST http://localhost:8080/api/tournament

```
{
    "name": "tournament 2"
}
```

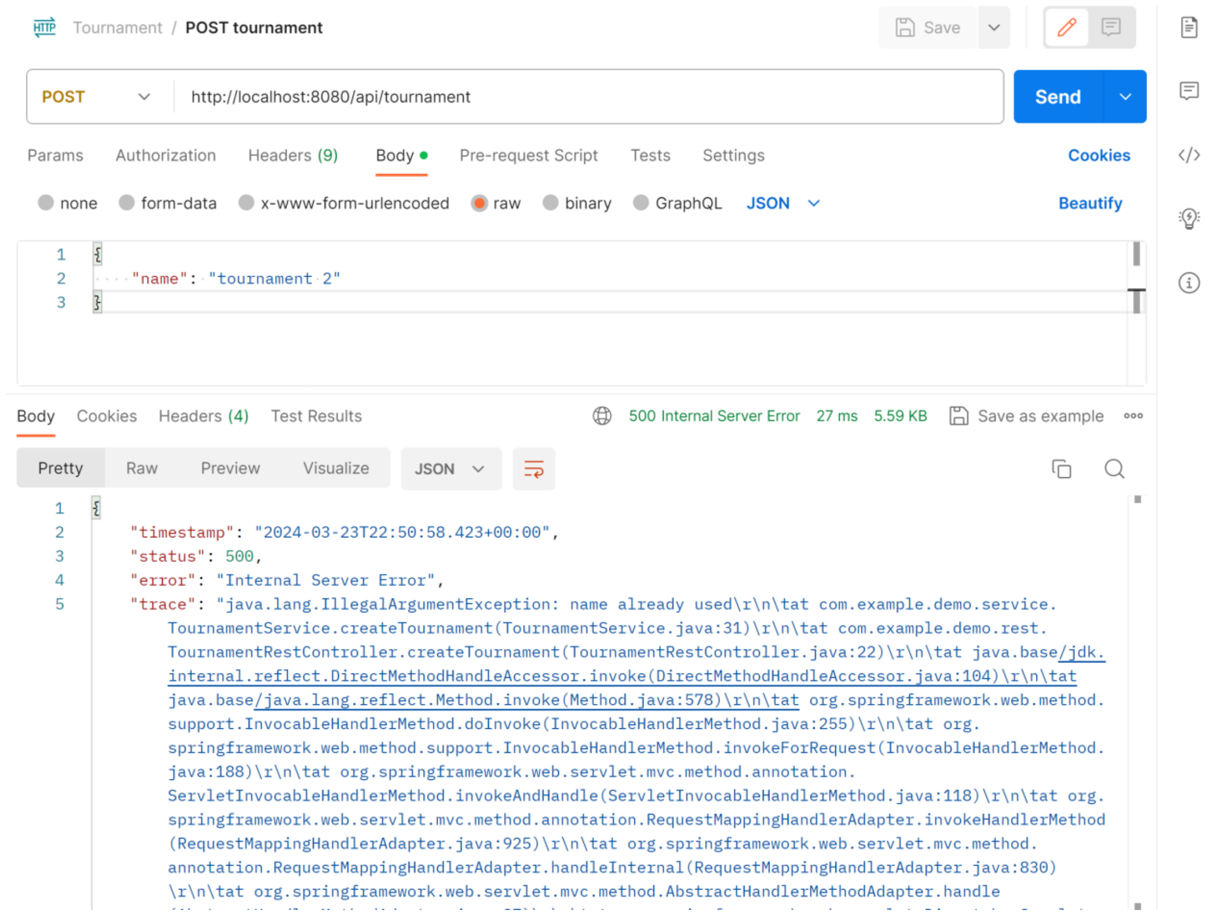
Launch request: click "Send"



```
{
····"name":·"tournament·2"
}
```

```
{
    "id": "65ff5c6be9a12719c3847644",
    "name": "tournament 2",
    "createdDate": "2024-03-23T23:49:15.4694042",
    "createdBy": "<<currentUser>>"
}
```

Relaunch (expecting error… name already used)



```
{
····"name":·"tournament·2"
}
```

```
{
    "timestamp": "2024-03-23T22:50:58.423+00:00",
    "status": 500,
    "error": "Internal Server Error",
    "trace": "java.lang.IllegalArgumentException: name already used\r\n\tat com.example.demo.service.
        TournamentService.createTournament(TournamentService.java:31)\r\n\tat com.example.demo.rest.
        TournamentRestController.createTournament(TournamentRestController.java:22)\r\n\tat java.base/jdk.
        internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:104)\r\n\tat
        java.base/java.lang.reflect.Method.invoke(Method.java:578)\r\n\tat org.springframework.web.method.
        support.InvocableHandlerMethod.doInvoke(InvocableHandlerMethod.java:255)\r\n\tat org.
        springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.
        java:188)\r\n\tat org.springframework.web.servlet.mvc.method.annotation.
        ServletInvocableHandlerMethod.invokeAndHandle(ServletInvocableHandlerMethod.java:118)\r\n\tat org.
        springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.invokeHandlerMethod
        (RequestMappingHandlerAdapter.java:925)\r\n\tat org.springframework.web.servlet.mvc.method.
        annotation.RequestMappingHandlerAdapter.handleInternal(RequestMappingHandlerAdapter.java:830)
        \r\n\tat org.springframework.web.servlet.mvc.method.AbstractHandlerMethodAdapter.handle
```

Adding support for OpenAPI (Swagger)

Step 1: edit pom.xml, add

```
<dependency>
    <groupId>org.springdoc</groupId>
    <artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>
    <version>2.1.0</version>
</dependency>
```

WARN …. This used to work with springboot 2.*, but not on 3.*

```
<!--
<dependency>
    <groupId>org.springdoc</groupId>
    <artifactId>springdoc-openapi-ui</artifactId>
    <version>1.7.0</version>
</dependency>
-->
```

Step 2:

Edit main application class, add

```
@OpenAPIDefinition(
       info = @Info(title = "Tournament App API", version = "1.0",
             description = "Rest API using OpenAPI 3 for a tutorial
Tournament application"))
```

Step 3:

Edit your Rest Controller, add annotation @OpenAPIDefinition to class

```
import io.swagger.v3.oas.annotations.OpenAPIDefinition;

@OpenAPIDefinition(
        // tags = { Tag("Tournament") }
)
```

And optionally add annotation @Operation to methods:

```
import io.swagger.v3.oas.annotations.Operation;
import io.swagger.v3.oas.annotations.responses.ApiResponse;
import io.swagger.v3.oas.annotations.responses.ApiResponses;
```

```
@Operation(summary = "Create a new (unique by name) tournament")
@ApiResponses(value = {
        @ApiResponse(responseCode = "200", description = "Successful
operation"),
        @ApiResponse(responseCode = "500", description = "name already
used"),
})
```

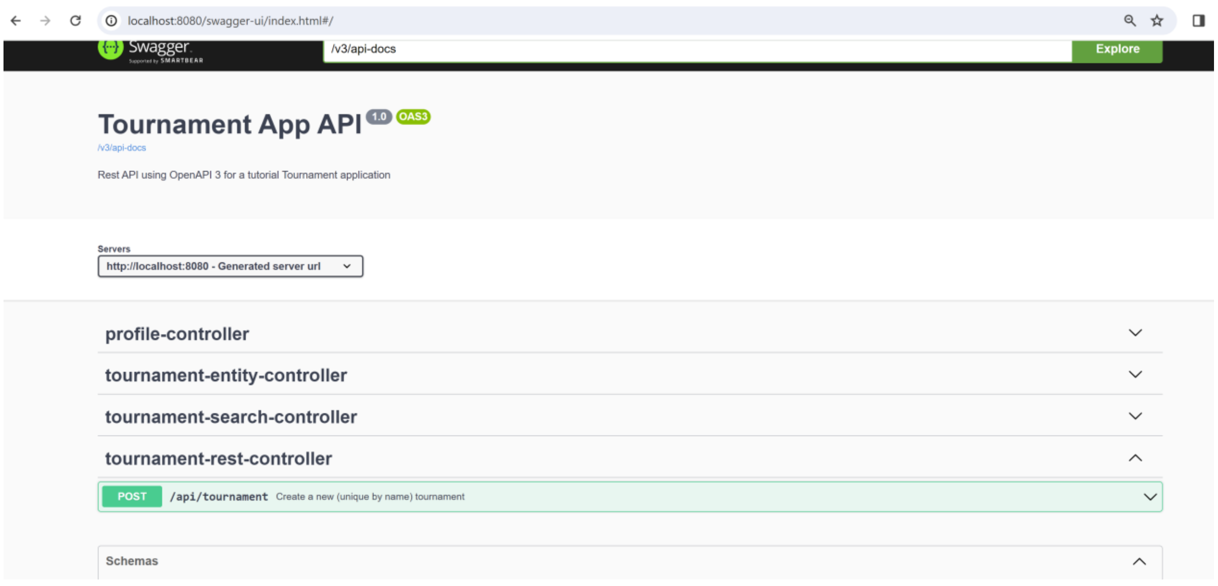
Helper static Html page

Add file “src/main/resources/static/index.html”

```
<html>
<body>

<H1>Test Tournament AApp (Springboot, Rest, MongoDB, OpenAPI)</H1>

<A href="/swagger-ui.html">/swagger-ui.html</A>
<br/>
<A href="/v3/api-docs">/v3/api-docs</A>
<br/>

</body>
```

Relaunch, open http://localhost:8080/swagger-ui.html

(This is redirected to http://localhost:8080/swagger-ui/index.html)



localhost:8080/swagger-ui/index.html#/

/v3/api-docs

Tournament App API 1.0 OAS3

/v3/api-docs

Rest API using OpenAPI 3 for a tutorial Tournament application

Servers

http://localhost:8080 - Generated server url

profile-controller

tournament-entity-controller

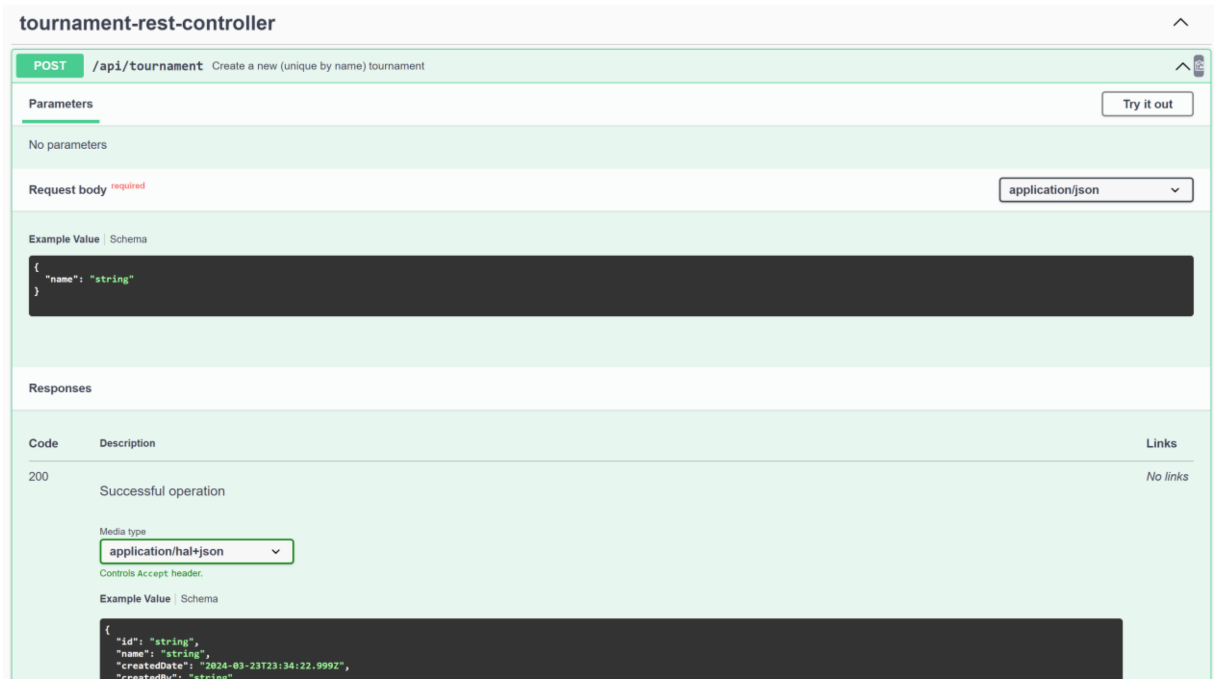
tournament-search-controller

tournament-rest-controller

POST /api/tournament Create a new (unique by name) tournament

Schemas

Expand custom method “POST /api/tournament”


tournament-rest-controller

POST `/api/tournament` Create a new (unique by name) tournament

Parameters — Try it out

No parameters

Request body required — application/json

Example Value | Schema

```
{
  "name": "string"
}
```

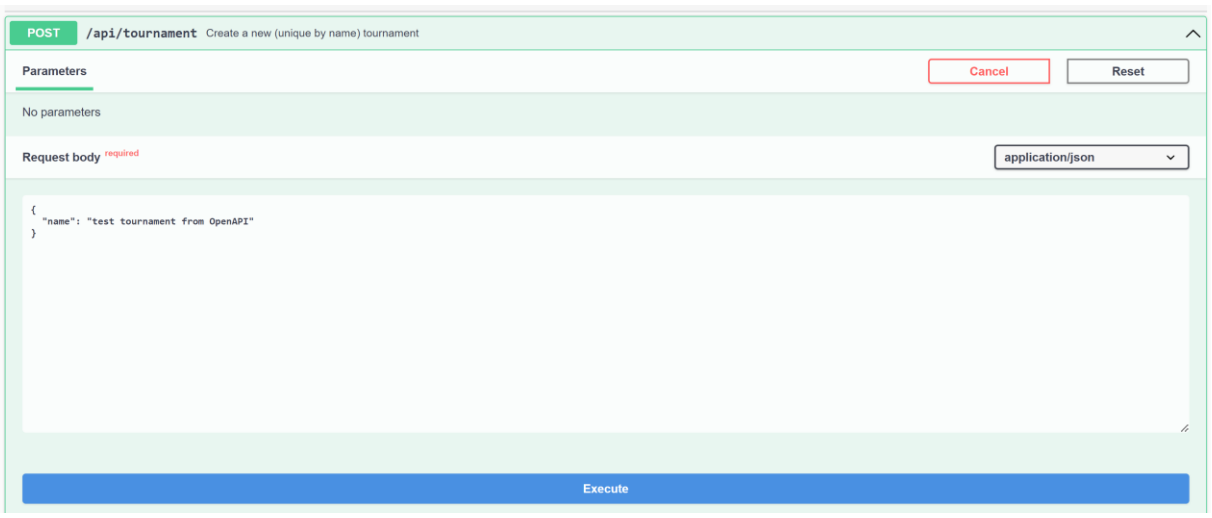
Responses

Code	Description	Links
200	Successful operation Media type: application/hal+json Controls Accept header.	*No links*

Example Value | Schema

```
{
  "id": "string",
  "name": "string",
  "createdDate": "2024-03-23T23:34:22.999Z",
  "createdBy": "string"
```

Then click on “Try it out”, and fill request body


POST `/api/tournament` Create a new (unique by name) tournament

Parameters — Cancel — Reset

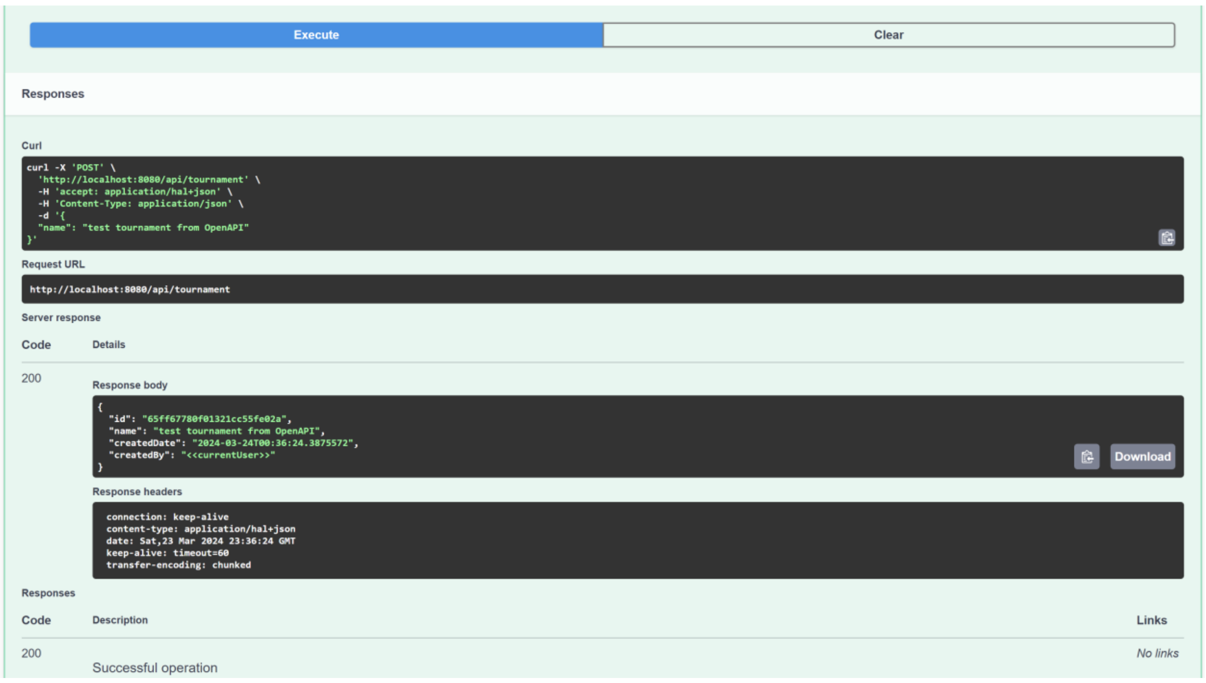
No parameters

Request body required — application/json

```
{
  "name": "test tournament from OpenAPI"
}
```

Execute

Click on “Execute”



Responses

Curl

```
curl -X 'POST' \
  'http://localhost:8080/api/tournament' \
  -H 'accept: application/hal+json' \
  -H 'Content-Type: application/json' \
  -d '{
  "name": "test tournament from OpenAPI"
}'
```

Request URL

```
http://localhost:8080/api/tournament
```

Server response

Code Details

200

Response body

```
{
  "id": "65ff67780f01321cc55fe02a",
  "name": "test tournament from OpenAPI",
  "createdDate": "2024-03-24T00:36:24.3875572",
  "createdBy": "<<currentUser>>"
}
```

Response headers

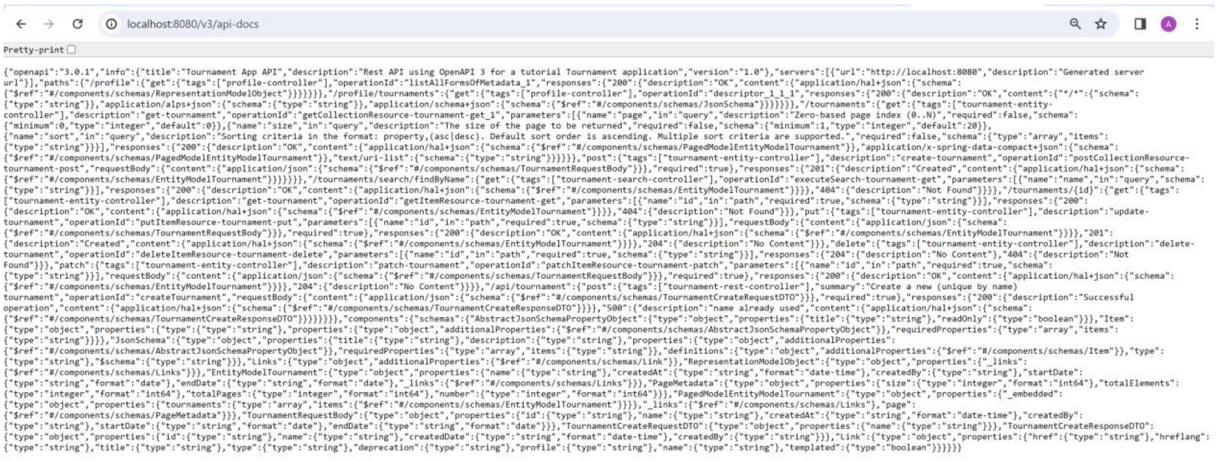
```
 connection: keep-alive
 content-type: application/hal+json
 date: Sat,23 Mar 2024 23:36:24 GMT
 keep-alive: timeout=60
 transfer-encoding: chunked
```

Responses

Code	Description	Links
200	Successful operation	*No links*

JSON open-api doc

Open http://localhost:8080/v3/api-docs



```
{"openapi":"3.0.1","info":{"title":"Tournament App API","description":"Rest API using OpenAPI 3 for a tutorial Tournament application","version":"1.0"},"servers":[{"url":"http://localhost:8080","description":"Generated server
url"}],"paths":{"/profile":{"get":{"tags":["profile-controller"],"operationId":"listAllFormsOfMetadata_1","responses":{"200":{"description":"OK","content":{"application/hal+json":{"schema":
{"$ref":"#/components/schemas/RepresentationModelObject"}}}}}}},"/profile/tournaments":{"get":{"tags":["profile-controller"],"operationId":"descriptor_1_1_1","responses":{"200":{"description":"OK","content":{"*/*":{"schema":
{"type":"string"}},"application/alps+json":{"schema":{"type":"string"}},"application/schema+json":{"schema":{"$ref":"#/components/schemas/JsonSchema"}}}}}}},"/tournaments":{"get":{"tags":["tournament-entity-
controller"],"description":"get-tournament","operationId":"getCollectionResource-tournament-get_1","parameters":[{"name":"page","in":"query","description":"Zero-based page index (0..N)","required":false,"schema":
{"minimum":0,"type":"integer","default":0}},{"name":"size","in":"query","description":"The size of the page to be returned","required":false,"schema":{"minimum":1,"type":"integer","default":20}},
{"name":"sort","in":"query","description":"Sorting criteria in the format: property,(asc|desc). Default sort order is ascending. Multiple sort criteria are supported.","required":false,"schema":{"type":"array","items":
{"type":"string"}}}],"responses":{"200":{"description":"OK","content":{"application/hal+json":{"schema":{"$ref":"#/components/schemas/PagedModelEntityModelTournament"}},"application/x-spring-data-compact+json":{"schema":
{"$ref":"#/components/schemas/PagedModelEntityModelTournament"}},"text/uri-list":{"schema":{"type":"string"}}}}}},"post":{"tags":["tournament-entity-controller"],"description":"create-tournament","operationId":"postCollectionResource-
tournament-post","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/TournamentRequestBody"}}},"required":true},"responses":{"201":{"description":"Created","content":{"application/hal+json":{"schema":
{"$ref":"#/components/schemas/EntityModelTournament"}}}}}}},"/tournaments/search/findByName":{"get":{"tags":["tournament-search-controller"],"operationId":"executeSearch-tournament-get","parameters":[{"name":"name","in":"query","schema":
{"type":"string"}}],"responses":{"200":{"description":"OK","content":{"application/hal+json":{"schema":{"$ref":"#/components/schemas/EntityModelTournament"}}}},"404":{"description":"Not Found"}}}},"/tournaments/{id}":{"get":{"tags":
["tournament-entity-controller"],"description":"get-tournament","operationId":"getItemResource-tournament-get","parameters":[{"name":"id","in":"path","required":true,"schema":{"type":"string"}}],"responses":{"200":
{"description":"OK","content":{"application/hal+json":{"schema":{"$ref":"#/components/schemas/EntityModelTournament"}}}},"404":{"description":"Not Found"}}},"put":{"tags":["tournament-entity-controller"],"description":"update-
tournament","operationId":"putItemResource-tournament-put","parameters":[{"name":"id","in":"path","required":true,"schema":{"type":"string"}}],"requestBody":{"content":{"application/json":{"schema":
{"$ref":"#/components/schemas/TournamentRequestBody"}}},"required":true},"responses":{"200":{"description":"OK","content":{"application/hal+json":{"schema":{"$ref":"#/components/schemas/EntityModelTournament"}}}},"201":
{"description":"Created","content":{"application/hal+json":{"schema":{"$ref":"#/components/schemas/EntityModelTournament"}}}},"204":{"description":"No Content"}}},"delete":{"tags":["tournament-entity-controller"],"description":"delete-
tournament","operationId":"deleteItemResource-tournament-delete","parameters":[{"name":"id","in":"path","required":true,"schema":{"type":"string"}}],"responses":{"204":{"description":"No Content"},"404":{"description":"Not
Found"}}},"patch":{"tags":["tournament-entity-controller"],"description":"patch-tournament","operationId":"patchItemResource-tournament-patch","parameters":[{"name":"id","in":"path","required":true,"schema":
{"type":"string"}}],"requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/TournamentRequestBody"}}},"required":true},"responses":{"200":{"description":"OK","content":{"application/hal+json":{"schema":
{"$ref":"#/components/schemas/EntityModelTournament"}}}},"204":{"description":"No Content"}}}},"/api/tournament":{"post":{"tags":["tournament-rest-controller"],"summary":"Create a new (unique by name)
tournament","operationId":"createTournament","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/TournamentCreateRequestDTO"}}},"required":true},"responses":{"200":{"description":"Successful
operation","content":{"application/hal+json":{"schema":{"$ref":"#/components/schemas/TournamentCreateResponseDTO"}}}},"500":{"description":"name already used","content":{"application/hal+json":{"schema":
{"$ref":"#/components/schemas/TournamentCreateResponseDTO"}}}}}}}},"components":{"schemas":{"AbstractJsonSchemaPropertyObject":{"type":"object","properties":{"title":{"type":"string"},"readOnly":{"type":"boolean"}}},"Item":
{"type":"object","properties":{"type":{"type":"string"},"properties":{"type":"object","additionalProperties":{"$ref":"#/components/schemas/AbstractJsonSchemaPropertyObject"}},"requiredProperties":{"type":"array","items":
{"type":"string"}}}},"JsonSchema":{"type":"object","properties":{"title":{"type":"string"},"description":{"type":"string"},"properties":{"type":"object","additionalProperties":
{"$ref":"#/components/schemas/AbstractJsonSchemaPropertyObject"}},"requiredProperties":{"type":"array","items":{"type":"string"}},"definitions":{"type":"object","additionalProperties":{"$ref":"#/components/schemas/Item"}},"type":
{"type":"string"},"$schema":{"type":"string"}}},"Links":{"type":"object","additionalProperties":{"$ref":"#/components/schemas/Link"}},"RepresentationModelObject":{"type":"object","properties":{"_links":
{"$ref":"#/components/schemas/Links"}}},"EntityModelTournament":{"type":"object","properties":{"name":{"type":"string"},"createdAt":{"type":"string","format":"date-time"},"createdBy":{"type":"string"},"startDate":
{"type":"string","format":"date"},"endDate":{"type":"string","format":"date"},"_links":{"$ref":"#/components/schemas/Links"}}},"PageMetadata":{"type":"object","properties":{"size":{"type":"integer","format":"int64"},"totalElements":
{"type":"integer","format":"int64"},"totalPages":{"type":"integer","format":"int64"},"number":{"type":"integer","format":"int64"}}},"PagedModelEntityModelTournament":{"type":"object","properties":{"_embedded":
{"type":"object","properties":{"tournaments":{"type":"array","items":{"$ref":"#/components/schemas/EntityModelTournament"}}}},"_links":{"$ref":"#/components/schemas/Links"},"page":
{"$ref":"#/components/schemas/PageMetadata"}}},"TournamentRequestBody":{"type":"object","properties":{"id":{"type":"string"},"name":{"type":"string"},"createdAt":{"type":"string","format":"date-time"},"createdBy":
{"type":"string"},"startDate":{"type":"string","format":"date"},"endDate":{"type":"string","format":"date"}}},"TournamentCreateRequestDTO":{"type":"object","properties":{"name":{"type":"string"}}},"TournamentCreateResponseDTO":
{"type":"object","properties":{"id":{"type":"string"},"name":{"type":"string"},"createdDate":{"type":"string","format":"date-time"},"createdBy":{"type":"string"}}},"Link":{"type":"object","properties":{"href":{"type":"string"},"hreflang":
{"type":"string"},"title":{"type":"string"},"type":{"type":"string"},"deprecation":{"type":"string"},"profile":{"type":"string"},"name":{"type":"string"},"templated":{"type":"boolean"}}}}}}
```

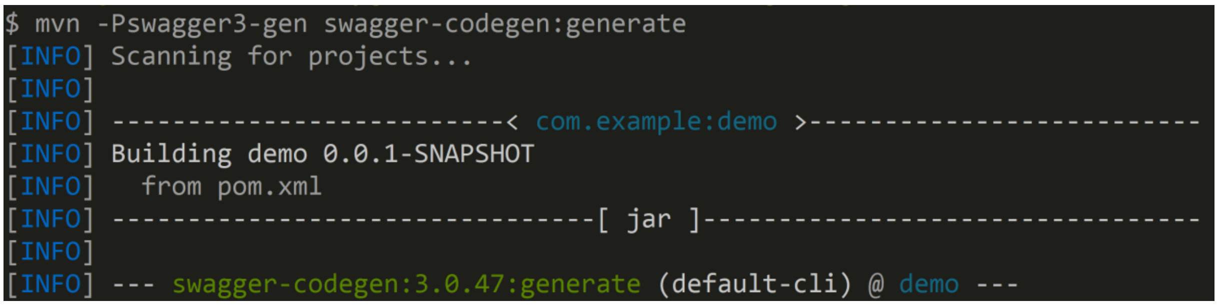
Generate Client code from OpenAPI for Angular

Edit pom.xml, add

```
<profiles>
    <profile>
       <id>swagger3-gen</id>
       <build>
          <plugins>
             <plugin>
                <groupId>io.swagger.codegen.v3</groupId>
                <artifactId>swagger-codegen-maven-plugin</artifactId>
                <version>3.0.47</version>
                <configuration>
                   <inputSpec>http://localhost:8080/v3/api-docs</inputSpec>
                   <language>typescript-angular</language>
                   <output>${basedir}/target/generated-typescript-
angular7</output>
                   <configOptions>
                      <ngVersion>17.0.0</ngVersion>
                   </configOptions>
                </configuration>
                <executions>
                   <execution>
                      <id>generate-swagger-typescript-angular-7</id>
                      <phase>generate-sources</phase>
                      <goals>
                         <goal>generate</goal>
                      </goals>
                   </execution>
                </executions>
             </plugin>
          </plugins>
       </build>
    </profile>

</profiles>
```

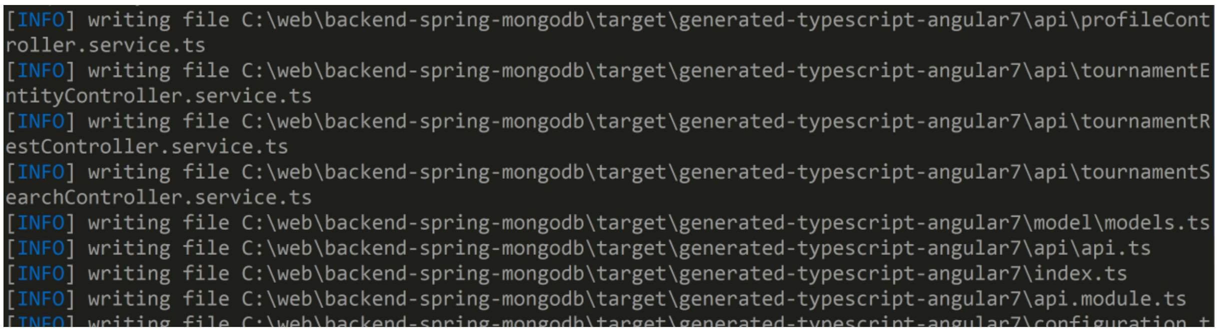
Then execute in terminal

```
mvn -Pswagger3-gen swagger-codegen:generate
```



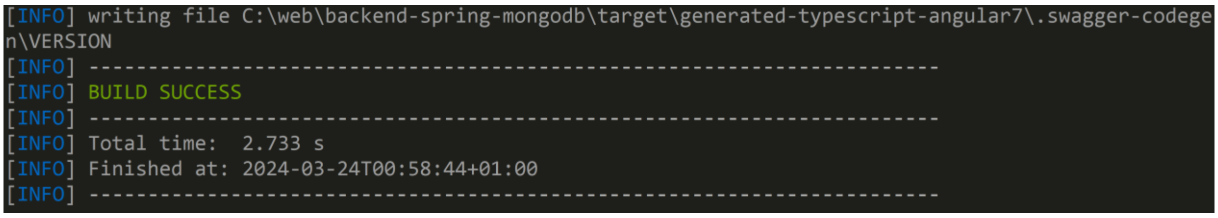
```
$ mvn -Pswagger3-gen swagger-codegen:generate
[INFO] Scanning for projects...
[INFO]
[INFO] --------------------------< com.example:demo >--------------------------
[INFO] Building demo 0.0.1-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- swagger-codegen:3.0.47:generate (default-cli) @ demo ---
```

…



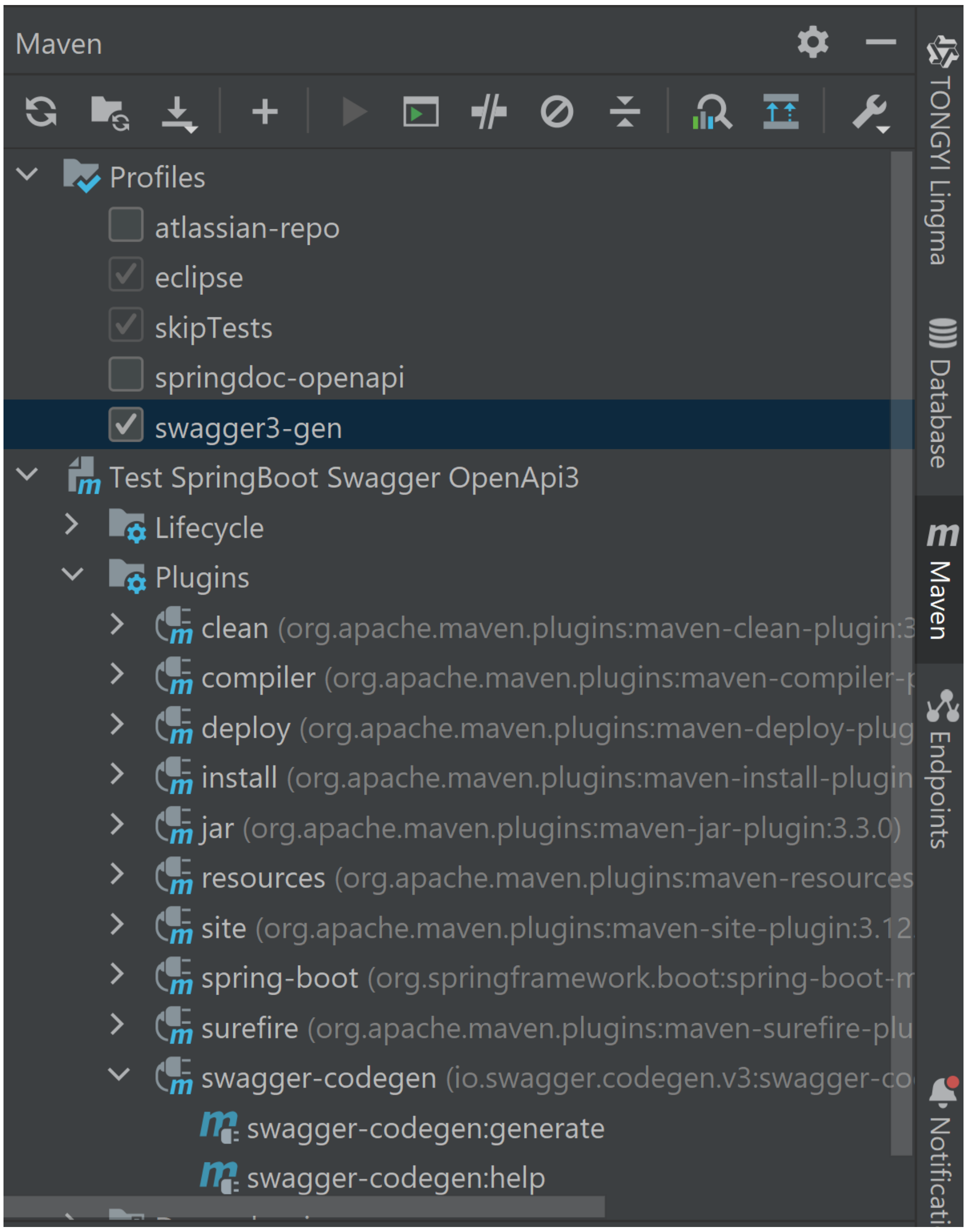
```
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\api\profileCont
roller.service.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\api\tournamentE
ntityController.service.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\api\tournamentR
estController.service.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\api\tournamentS
earchController.service.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\model\models.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\api\api.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\index.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\api.module.ts
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\configuration.t
```

…



```
[INFO] writing file C:\web\backend-spring-mongodb\target\generated-typescript-angular7\.swagger-codege
n\VERSION
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.733 s
[INFO] Finished at: 2024-03-24T00:58:44+01:00
[INFO] ------------------------------------------------------------------------
```

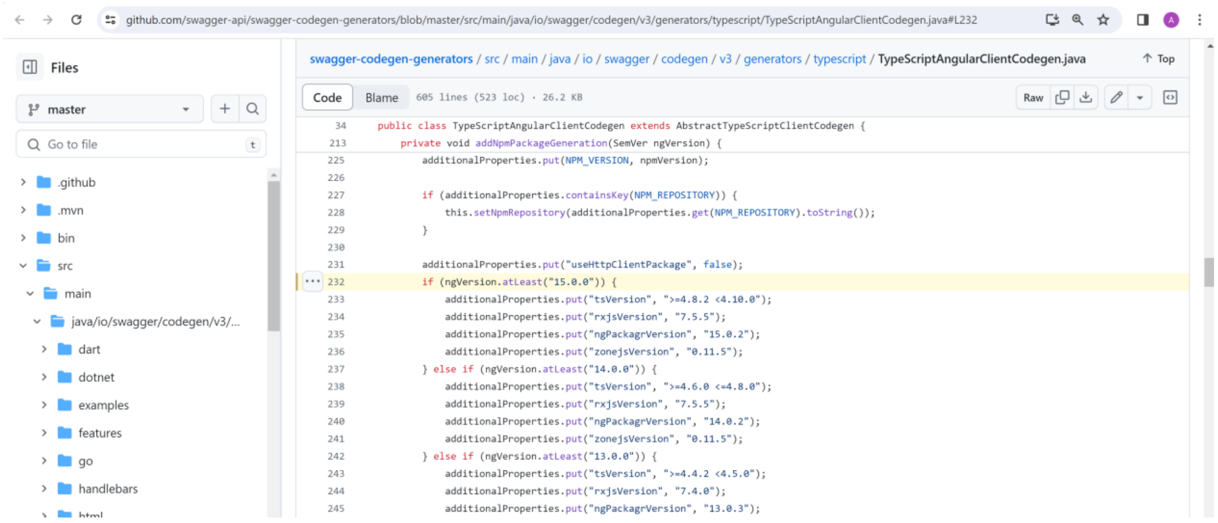
Alternatively, excute maven goal from IntelliJ



```
Profiles
    ☐ atlassian-repo
    ☑ eclipse
    ☑ skipTests
    ☐ springdoc-openapi
    ☑ swagger3-gen
Test SpringBoot Swagger OpenApi3
    Lifecycle
    Plugins
        clean (org.apache.maven.plugins:maven-clean-plugin:3
        compiler (org.apache.maven.plugins:maven-compiler-p
        deploy (org.apache.maven.plugins:maven-deploy-plug
        install (org.apache.maven.plugins:maven-install-plugin
        jar (org.apache.maven.plugins:maven-jar-plugin:3.3.0)
        resources (org.apache.maven.plugins:maven-resources
        site (org.apache.maven.plugins:maven-site-plugin:3.12
        spring-boot (org.springframework.boot:spring-boot-m
        surefire (org.apache.maven.plugins:maven-surefire-plu
        swagger-codegen (io.swagger.codegen.v3:swagger-co
            swagger-codegen:generate
            swagger-codegen:help
```

Browsing source code of plugin for supported version of Angular

https://github.com/swagger-api/swagger-codegen-generators/blob/master/src/main/java/io/swagger/codegen/v3/generators/typescript/TypeScriptAngularClientCodegen.java#L232



github.com/swagger-api/swagger-codegen-generators/blob/master/src/main/java/io/swagger/codegen/v3/generators/typescript/TypeScriptAngularClientCodegen.java#L232
Files
master
Go to file
.github
.mvn
bin
src
main
java/io/swagger/codegen/v3/...
dart
dotnet
examples
features
go
handlebars
swagger-codegen-generators / src / main / java / io / swagger / codegen / v3 / generators / typescript / TypeScriptAngularClientCodegen.java
Top
Code
Blame
605 lines (523 loc) · 26.2 KB
Raw

```java
public class TypeScriptAngularClientCodegen extends AbstractTypeScriptClientCodegen {
    private void addNpmPackageGeneration(SemVer ngVersion) {
        additionalProperties.put(NPM_VERSION, npmVersion);

        if (additionalProperties.containsKey(NPM_REPOSITORY)) {
            this.setNpmRepository(additionalProperties.get(NPM_REPOSITORY).toString());
        }

        additionalProperties.put("useHttpClientPackage", false);
        if (ngVersion.atLeast("15.0.0")) {
            additionalProperties.put("tsVersion", ">=4.8.2 <4.10.0");
            additionalProperties.put("rxjsVersion", "7.5.5");
            additionalProperties.put("ngPackagrVersion", "15.0.2");
            additionalProperties.put("zonejsVersion", "0.11.5");
        } else if (ngVersion.atLeast("14.0.0")) {
            additionalProperties.put("tsVersion", ">=4.6.0 <=4.8.0");
            additionalProperties.put("rxjsVersion", "7.5.5");
            additionalProperties.put("ngPackagrVersion", "14.0.2");
            additionalProperties.put("zonejsVersion", "0.11.5");
        } else if (ngVersion.atLeast("13.0.0")) {
            additionalProperties.put("tsVersion", ">=4.4.2 <4.5.0");
            additionalProperties.put("rxjsVersion", "7.4.0");
            additionalProperties.put("ngPackagrVersion", "13.0.3");
```