Get all the published frontend and backend courses,

sort them by their price in a descending order,

pick only their name and author,

and display them.

Solution

```javascript
const mongoose = require('mongoose');

mongoose.connect('mongodb://localhost/mongo-exercises');

const courseSchema = new mongoose.Schema({
  name: String,
  author: String,
  tags: [ String ],
  date: Date,
  isPublished: Boolean,
  price: Number
});

const Course = mongoose.model('Course', courseSchema);

async function getCourses() {
  return await Course
    .find({ isPublished: true, tags: { $in: ['frontend','backend'] } })
    .sort({ price: -1 }) // or .sort('price') // for descending .sort('-price')
```

TERMINAL

```
rser, pass option { useNewUrlParser: true } to MongoClient.connect.
(node:5086) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future versi
on. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[ { _id: 5a68fdd7bee8ea64649c2777,
    name: 'Node.js Course',
    author: 'Mosh' },
  { _id: 5a6900fff467be65019a9001,
    name: 'Angular Course',
    author: 'Mosh' },
  { _id: 5a68fde3f09ad7646ddec17e,
    name: 'ASP.NET MVC Course',
    author: 'Mosh' },
  { _id: 5a68fe2142ae6a6482c4c9cb,
    name: 'Node.js Course by Jack',
    author: 'Jack' },
  { _id: 5a68fdc3615eda645bc6bdec,
    name: 'Express.js Course',
    author: 'Mosh' } ]
```

```js
const Course = mongoose.model('Course', courseSchema);

async function getCourses() {
  return await Course
  .find({ isPublished: true, tags: { $in: ['frontend','backend'] } })
  .sort({ price: -1 }) // or .sort('price') // for descending .sort('-price')
  .select({ name: 1, author: 1 }); // or .select('name author')
}

async function run() {
  const courses = await getCourses();
  console.log(courses);
}

run();
```

index.js - Applications - Visual Studio Code

File   Edit   Selection   View   Go   Run   Terminal   Help

EXPLORER

JS index.js ✕

15. Exercise 2 › mongo-demo › JS index.js › ⬡ getCourses

OPEN EDITORS
- ✕ JS index.js   15. Exercise 2/mongo-demo

APPLICATIONS
- › 01. Introducing MongoDB
- › 02. Installing MongoDB On Mac
- › 03. Installing MongoDB On Windows
- › 04. Connecting To MongoDB
- › 05. Schemas
- › 06. Models
- › 07. Saving A Document
- › 08. Querying Documents
- › 09. Comparison Query Operators
- › 10. Logical Query Operators
- › 11. Regular Expressions
- › 12. Counting
- › 13. Pagination
- › 14. Exercise 1
- ⌄ 15. Exercise 2
  - ⌄ mongo-demo
    - › node_modules
    - JS index.js
    - {} package.json
  - ⓘ README.md
- › 16. Exercise 3
- › 17. Updating Documents - Query First
- › 18. Updating A Document - Update First
- › 19. Removing Documents
- › 20. CRUD Operations With Mongoose And Mongo...
- ⓘ README.md

```js
15
16  async function getCourses() {
17    return await Course
18      .find({ isPublished: true, tags: { $in: ['frontend','backend'] } })
19      .sort({ price: -1 }) // or .sort('price') // for descending .sort('-price')
20      .select('name author price')//.select({ name: 1, author: 1, price: 1 })
21  }
22
23  async function run() {
24    const courses = await getCourses();
25    console.log(courses);
26  }
27
28  run();
```

DEBUG CONSOLE   PROBLEMS   OUTPUT   **TERMINAL**   2: node ▾

```
dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/07. CRUD Operations Using MongoDB/Applications/15. Exercise 2/mongo-demo$ node i
ndex.js
(node:5131) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new pa
rser, pass option { useNewUrlParser: true } to MongoClient.connect.
(node:5131) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future versi
on. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[ { _id: 5a68fdd7bee8ea64649c2777,
    name: 'Node.js Course',
    author: 'Mosh',
    price: 20 },
  { _id: 5a6900fff467be65019a9001,
    name: 'Angular Course',
    author: 'Mosh',
    price: 15 },
  { _id: 5a68fde3f09ad7646ddec17e,
    name: 'ASP.NET MVC Course',
    author: 'Mosh',
    price: 15 },
  { _id: 5a68fe2142ae6a6482c4c9cb,
    name: 'Node.js Course by Jack',
    author: 'Jack',
    price: 12 },
  { _id: 5a68fdc3615eda645bc6bdec,
    name: 'Express.js Course',
    author: 'Mosh',
    price: 10 } ]
```

OUTLINE

TIMELINE

File   Edit   Selection   View   Go   Run   Terminal   Help

EXPLORER

OPEN EDITORS
  × JS index.js 15. Exercise 2/mongo-demo
APPLICATIONS
  > 01. Introducing MongoDB
  > 02. Installing MongoDB On Mac
  > 03. Installing MongoDB On Windows
  > 04. Connecting To MongoDB
  > 05. Schemas
  > 06. Models
  > 07. Saving A Document
  > 08. Querying Documents
  > 09. Comparison Query Operators
  > 10. Logical Query Operators
  > 11. Regular Expressions
  > 12. Counting
  > 13. Pagination
  > 14. Exercise 1
  ∨ 15. Exercise 2
    ∨ mongo-demo
      > node_modules
      JS index.js
      {} package.json
    README.md
  > 16. Exercise 3
  > 17. Updating Documents - Query First
  > 18. Updating A Document - Update First
  > 19. Removing Documents
  > 20. CRUD Operations With Mongoose And Mongo...
    README.md

15. Exercise 2 > mongo-demo > JS index.js > ⬡ run

```javascript
16   async function getCourses() {
17     return await Course
18     //.find({ isPublished: true, tags: { $in: ['frontend','backend'] } })
19       .find({ isPublished: true})
20       .or([ { tags: 'frontend' }, { tags: 'backend' } ])
21       .sort({ price: -1 }) // or .sort('price') // for descending .sort('-price')
22       .select('name author price')//.select({ name: 1, author: 1, price: 1 })
23   }
24
25   async function run() {
26     const courses = await getCourses();
27     console.log(courses);
28   }
```

DEBUG CONSOLE   PROBLEMS   OUTPUT   TERMINAL   2: node ▾

```
dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/07. CRUD Operations Using MongoDB/Applications/15. Exercise 2/mongo-demo$ node i
ndex.js
(node:5181) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new pa
rser, pass option { useNewUrlParser: true } to MongoClient.connect.
(node:5181) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future versi
on. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[ { _id: 5a68fdd7bee8ea64649c2777,
    name: 'Node.js Course',
    author: 'Mosh',
    price: 20 },
  { _id: 5a6900fff467be65019a9001,
    name: 'Angular Course',
    author: 'Mosh',
    price: 15 },
  { _id: 5a68fde3f09ad7646ddec17e,
    name: 'ASP.NET MVC Course',
    author: 'Mosh',
    price: 15 },
  { _id: 5a68fe2142ae6a6482c4c9cb,
    name: 'Node.js Course by Jack',
    author: 'Jack',
    price: 12 },
  { _id: 5a68fdc3615eda645bc6bdec,
    name: 'Express.js Course',
    author: 'Mosh',
    price: 10 } ]
```