

Universidade Federal de Pernambuco

Centro de Informática - CIn

Cursos para Concursos



Documentação Grupo 3

Gerenciamento de dados e Informação - 2021.2

José Danilo Silva do Carmo - jdsc

Humberto Lima de Alcântara Fonseca Filho - hlaff

Emanuel Vinícius Tavares dos Santos Gomes- evtsg

Sérgio Leonardo Barreto Paixão - slbp

Antonio Bento de Souza Neto - absn3

Descrição da aplicação

O propósito é criar um banco de dados MongoDB para a aplicação do tema **Cursos para Concursos**, podendo criar, remover, modificar e ler dados. No mundo real, um curso de determinada matéria e com foco em algum(ns) concurso(s) é ofertado e o aluno pode escolher cadastrar-se nele, para então ter as aulas com os professores daquele curso e, eventualmente, terminar o curso. Tudo isso será armazenado em conjunto com outras informações, sendo muitas delas concernentes aos alunos.

A seguir serão apresentadas as *collections* existentes, bem como os seus dados..

Collections

Courses:

Essa *collection* armazena os dados que cada curso ofertado contém.

Ela possui os seguintes dados:

- **name:** o nome do curso
- **description:** uma descrição a respeito do propósito do curso
- **durationInMonths:** duração do curso, medida em meses
- **URL:** *link* para o site do curso, caso exista. Um curso será feito no formato EAD se e somente se tiver URL
- **price:** preço do curso
- **address:** endereço de onde ocorrerá as aulas do curso, caso exista.

Um curso será presencial se e somente se tiver *address*

- street: rua
- city: cidade
- state: estado
- **teacherList[]:** *array* contendo os professores responsáveis por dar aula no curso

- name: nome do professor
- wage: salário do professor
- **classesList[]**: *array* contendo as aulas que serão ministradas no curso
 - teacher: nome do professor responsável pela aula
 - className: nome da aula. Descreve o assunto que será abordado nela
 - Q&A[]: questões relacionadas ao assunto abordado na aula
 - question: Pode ser um identificador (como “questão 1”) ou o próprio enunciado da questão
 - answer: a resposta para a questão
- **targetContests[]**: os concursos que esse curso tem como foco
- **assistantList[]**: *array* contendo os monitores do curso
 - name: nome do monitor
 - wage: salário do monitor
 - availabilityList[]: *array* contendo os horários de disponibilidade do monitor
 - day: dia de disponibilidade
 - startingHour: o horário a partir do qual o monitor está disponível
 - durationInMinutes: por quanto tempo após o horário inicial o monitor ficará disponível

Students:

Essa *collection* contém os dados concernentes aos aluno matriculados em algum dos cursos disponibilizados

- **name**: nome do aluno
- **email**: e-mail do aluno
- **birth_date**: data de nascimento do aluno
- **password**: senha do aluno
- **cpf**: cpf do aluno
- **courseList[]**: *array* contendo os cursos nos quais o aluno está matriculado

- name: nome do curso
- progress: progresso, em porcentagem, que o aluno concluiu do curso
- start_date: data na qual o aluno começou as aulas no curso referido

Scripts

```
// Entra no banco de dados "aguiaConcursos"
use aguiaConcursos;

// Cria uma collection "estudantes" e armazena os dados dos estudante
db.estudantes.insertMany([
  {
    "name": "José Roberto da Silva",
    "email": "joseRS@email.com",
    "birthDate": new Date("06/01/1984"),
    "password": "senha123",
    "courseList": [
      {
        "name": "course1",
        "progress": "40%",
        "startDate": new Date("08/14/2019")
      },
      {
        "name": "course2",
        "progress": "60%",
        "startDate": new Date("01/03/2020")
      },
      {
        "name": "course3",
        "progress": "100%",
        "startDate": new Date("02/24/2019")
      }
    ]
  },
  {
    "cpf": "11122233304"
  }
])
```

```
},
{
  "name": "Marcelo da Silva Santos",
  "email": "marceloSS@email.com",
  "birthDate": new Date("06/01/1990"),
  "password": "senha",
  "courseList": [
    {
      "name": "course3",
      "progress": "40%",
      "startDate": new Date("03/05/2021")
    },
    {
      "name": "course2",
      "progress": "60%",
      "startDate": new Date("09/01/2020")
    }
  ],
  "cpf": "12647532408"
},
{
  "name": "Maria Júlia Alves",
  "email": "majuAlves@email.com",
  "birthDate": new Date("08/13/1999"),
  "password": "pass",
  "courseList": [],
  "cpf": "01245682400"
},
{
  "name": "Calebe Tiago Fernando Fernandes",
  "email": "calebetiagofernandes@crsilvadesign.com",
  "birthDate": new Date("01/02/1978"),
  "password": "ANqLUEcr3E",
  "courseList": [
    {
      "name": "course1",
      "progress": "100%",
```

```
        "startDate": new Date("08/08/2021")
    },
    {
        "name": "course2",
        "progress": "20%",
        "startDate": new Date("11/07/2020")
    }
],
"cpf": "23863102304"
},
{
    "name": "Lucas Gabriel da Rocha",
    "email": "LucasGR@email.com",
    "birthDate": new Date("12/11/2000"),
    "password": "senha",
    "courseList": [
        {
            "name": "course2",
            "progress": "20%",
            "startDate": new Date("04/22/2019")
        },
        {
            "name": "course1",
            "progress": "90%",
            "startDate": new Date("07/30/2020")
        }
    ],
    "cpf": "10024547806"
},
{
    "name": "Fabyo Henrique de Melo",
    "email": "fabyoHM@email.com",
    "birthDate": new Date("02/27/1997"),
    "password": "senha",
    "courseList": [
        {
            "name": "course4",
```

```

        "progress": "40%",
        "startDate": new Date("12/11/2019")
    },
    {
        "name": "course5",
        "progress": "100%",
        "startDate": new Date("03/24/2020")
    }
],
"cpf": "12647532408"
},
{
    "name": "Pedro Leonardo de Sousa",
    "email": "pedroLS@email.com",
    "birthDate": new Date("07/15/2001"),
    "password": "senha",
    "courseList": [
        {
            "name": "course2",
            "progress": "100%",
            "startDate": new Date("08/11/2020")
        },
        {
            "name": "course5",
            "progress": "0%",
            "startDate": new Date("04/24/2022")
        }
    ],
    "cpf": "14725836904"
},
{
    "name": "Matheus Melo Machado",
    "email": "matheusMM@email.com",
    "birthDate": new Date("05/28/2000"),
    "password": "senha",
    "courseList": [
        {

```

```

        "name": "course1",
        "progress": "100%",
        "startDate": new Date("08/14/2019")
    },
    {
        "name": "course4",
        "progress": "100%",
        "startDate": new Date("03/21/2020")
    }
],
"cpf": "15935745608"
},
{
    "name": "Luana Lopes Machado",
    "email": "LuanaLM@email.com",
    "birthDate": new Date("11/20/1999"),
    "password": "senha",
    "courseList": [
        {
            "name": "course2",
            "progress": "100%",
            "startDate": new Date("04/20/2021"),
        },
    ],
    "cpf": "12378954600"
},
{
    "name": "Anna Beatriz Castro",
    "email": "annaBC@email.com",
    "birthDate": new Date("02/27/1989"),
    "password": "2b0dfb4389a743e54fb7706ff6f885a9",
    "courseList": [
        {
            "name": "course3",
            "progress": "40%",
            "startDate": new Date("06/14/2021")
        },
    ],

```



```

        {
            "name": "course4",
            "progress": "10%",
            "startDate": new Date("05/01/2022")
        },
    ],
    "cpf": "96335741032"
}
]);

// Create courses and inset them in the database

db.courses.insertOne({
    "name": "Criptografia de Dados",
    "description": "Aprender conceitos sobre criptografia e saber como
    utilizá-la para proteger dados.",
    "durationInMonths": 5,
    "price": 1095.95,
    "URL": "http://aguiaconcursos.com/criptografia+de+dados",
    "address": {
        "street": "Rua Major João Silveira",
        "city": "Recife",
        "state": "Pernambuco"
    },
    "teacherList": [
        {
            "name": "Pedro Kuber Novalho",
            "wage": 2507.05
        }, {
            "name": "Jorge da Silva Ramos Filho",
            "wage": 8202.90
        }, {
            "name": "Vladmir Costa Burgos",
            "wage": 13200.35
        }
    ],
    "classesList": [

```

```
{
  "className": "Tópico 1",
  "teacher": "Vladmir Costa Burgos",
  "Q&A": [
    {
      "question": "Questão 1",
      "answer": "V"
    }, {
      "question": "Questão 2",
      "answer": "F"
    }, {
      "question": "Questão 3",
      "answer": "Pois a criptografia possui vários ramos,
não apenas um"
    }, {
      "question": "Questão 4",
      "answer": "Depende do tipo de criptografia
utilizado",
    }, {
      "question": "Questão 5",
      "answer": "Chave assimétrica"
    },
  ],
}, {
  "className": "Tópico 2",
  "teacher": "Jorge da Silva Ramos Filho",
  "Q&A": [
    {
      "question": "Questão 1",
      "answer": "F"
    }, {
      "question": "Questão 2",
      "answer": "V",
    }, {
      "question": "Questão 3",
```

```

        "answer": "Criptografia avançada para dados
multivalorados"
    }, {
        "question": "Questão 4",
        "answer": "Utilizando uma função hash"
    },
],
}, {
    "className": "Tópico 3",
    "teacher": "Pedro Kuber Novalho",
    "Q&A": [
        {
            "question": "Questão 1",
            "answer": "F"
        }, {
            "question": "Questão 2",
            "answer": "V",
        }, {
            "question": "Questão 3",
            "answer": "V",
        }, {
            "question": "Questão 4",
            "answer": "V",
        }, {
            "question": "Questão 5",
            "answer": "Sim"
        },
    ],
}
],
"targetContests": ["Polícia Federal de Alagoas", "Polícia
Rodoviária de Alagoas"],
"assistantList": [{
    "name": "Pedro Sávio Costa",
    "wage": 0,

```

```
    "availabilityList": [  
      {  
        "day": "Segunda-feira",  
        "startingHour": "13:00",  
        "durationInMinutes": 30  
      },  
      {  
        "day": "Terça-feira",  
        "startingHour": "17:00",  
        "durationInMinutes": 45  
      },  
      {  
        "day": "Quinta-feira",  
        "startingHour": "12:00",  
        "durationInMinutes": 60  
      }  
    ]  
  },  
  {  
    "name": "Jorge Medeiro Santos",  
    "wage": 350,  
    "availabilityList": [  
      {  
        "day": "Quarta-feira",  
        "startingHour": "15:00",  
        "durationInMinutes": 120  
      },  
      {  
        "day": "Sexta-feira",  
        "startingHour": "12:00",  
        "durationInMinutes": 240  
      },  
      {  
        "day": "Sábado",  
        "startingHour": "08:00",  
        "durationInMinutes": 180  
      }  
    ]  
  }  
]
```

```

    ]
  }
]
});

db.courses.insertOne({
  "name": "Direito Penal",
  "description": "Entender sobre as leis concernentes ao Direito
Penal",
  "durationInMonths": 8,
  "price": 12000.95,
  "URL": "",
  "address": {
    "street": "Rua da Algaroba",
    "city": "São Paulo",
    "state": "SP"
  },
  "teacherList": [
    {
      "name": "João Pedro dos Santos",
      "wage": 5200.95,
    },
    {
      "name": "Robéria Burgos Sá",
      "wage": 9262.62,
    },
    {
      "name": "Haldir Vladniskotv Brakt",
      "wage": 2400.25
    }
  ],
  "classesList": [
    {
      "teacher": "Robéria Burgos Sá",
      "className": "Filosofia existencial a respeito do Código
Penal",
      "Q&A": [

```

```

        {
            "question": "Por que existe o Código Penal?",
            "answer": "Para regir as punições que devem ser
aplicadas áqueles que perturbam a ordem social."
        },
        {
            "question": "Qual a relação entre Direito Penal e
Código Penal?",
            "answer": "O Direito Penal é a aplicação das leis
descritas no Código Penal."
        }
    ]
},
{
    "teacher": "Haldir Vladniskotv Brakt",
    "className": "Direito Penal na prática",
    "Q&A": [
        {
            "question": "Quanto tempo de prisão recebe aquele
que furta um objeto de R$500,00?",
            "answer": "5 meses e 16 dias, com direito à prisão
domiciliar após cumpridos 60% do prazo."
        },
        {
            "question": "Cite dois exemplos de crimes
inafiançáveis.",
            "answer": "Estupro e sequestro."
        }
    ]
},
{
    "teacher": "João Pedro dos Santos",
    "className": "Tópicos avançados em Direito Penal",
    "Q&A": [
        {
            "question": "Questão 1",
            "answer": "V"
        }
    ]
}

```

```

        },
        {
            "question": "Questão 2",
            "answer": "F"
        },
        {
            "question": "Questão 3",
            "answer": "V"
        },
        {
            "question": "Questão 4",
            "answer": "V"
        }
    ]
},
],
"targetContests": [
    "Corpo de Bombeiros Militar de Pernambuco",
    "Polícia Rodoviária Federal de Tocantins"
],
"assistantList": [
    {
        "name": "Jorge Medeiros Sávio",
        "wage": 125.00,
        "availabilityList": [
            {
                "day": "Segunda-feira",
                "startingHour": "10:00",
                "durationInMinutes": 90
            },
            {
                "day": "Quarta-feira",
                "startingHour": "12:00",
                "durationInMinutes": 240
            }
        ]
    }
],
},

```

```

    {
      "name": "Pedro Tenório Ávila",
      "wage": 850.00,
      "availabilityList": [
        {
          "day": "Segunda-feira",
          "startingHour": "11:00",
          "durationInMinutes": 300
        },
        {
          "day": "Terça-feira",
          "startingHour": "12:00",
          "durationInMinutes": 240
        },
        {
          "day": "Quarta-feira",
          "startingHour": "8:00",
          "durationInMinutes": 200
        },
        {
          "day": "Sexta-feira",
          "startingHour": "8:00",
          "durationInMinutes": 200
        },
        {
          "day": "Sábado",
          "startingHour": "10:00",
          "durationInMinutes": 100
        }
      ]
    }
  ]
});

db.courses.save({
  "name": "Curso Polícia Federal",
  "description": "Curso completo para concursos da polícia federal",

```



```
"durationInMonths": 4,
"price": 1500,
"URL": "http://aguiaconcursos.com.br/policia+federal",
"address": null,
"teacherList": [
  {
    "name": "João Paulo Borges Machado",
    "wage": 4300.00
  }, {
    "name": "Veridiano de Sá da Costa",
    "wage": 6300.87
  },
],
"classesList": [
  {
    "teacher": "João Paulo Borges Machado",
    "className": "Definindo as etapas para conseguir
aprovação",
    "Q&A": [
      {
        "question": "Esse curso abrange tudo que é preciso
para aprovação nos concursos da polícia federal?",
        "answer": "Sim"
      }
    ]
  },
  {
    "teacher": "João Paulo Borges Machado",
    "className": "Raciocínio Logico Matemático",
    "Q&A": [
      {
        "question": "Esse curso abrange tudo que é preciso
para aprovação nos concursos da polícia federal?",
        "answer": "Sim"
      }
    ]
  },
],
```

```

    {
        "teacher": "Veridiano de Sá da Costa",
        "className": "Conceito de Internet e Intranet.Conceitos e
modos de utilização de tecnologia e ferramentas",
        "Q&A": [
            {
                "question": "Quais os principais protocolos da
internet?",
                "answer": "HTTP (acessar páginas Web), FTP
(transferir arquivos), SMTP (enviar e-mails), POP3 (recebere-mails),
IMAP4 (receber e-mails).",
            },
            {
                "question": "O que é um protocolo?",
                "answer": "Protocolo é o conjunto de regras que
definem o modo como se dará a comunicação entre dispositivos conectados
em uma rede."
            }
        ]
    }

],
"targetContests": [
    "Polícia Federal",
    "Polícia Rodoviária Federal"
],
"assistantList": [
    {
        "name": "Pedro Henrique",
        "wage": 1000,
        "availabilityList": [
            {
                "day": "Quinta-feira",
                "startingHour": "09:00",
                "durationInMinutes": 60
            },
            {

```

```

        "day": "Sexta-feira",
        "startingHour": "20:00",
        "durationInMinutes": 60
    }
}
]
}
]
});
// [RenameCollection] Renomeia a coleção de estudantes
db.estudantes.renameCollection("students");

// [Where] Verifica se a senha do usuário é igual ao nome
db.students.find({
    $where: function() {
        return hex_md5(this.name) === this.password
    }
}).pretty();

// Remove password field
db.students.update(
    {},
    { $unset: {
        password: 1
    } },
    { multi: true }
);

// [Delete] e [Size] Remove os alunos sem nenhum curso
db.students.deleteMany({
    "courseList": {
        $size: 0
    }
})

// [Update] e [Set] Atualiza o nome do curso dos alunos
db.students.updateMany({

```

```
    "courseList.name": "course1"
  }, {
    $set: {
      "courseList.$.name": "Criptografia de Dados"
    }
  })
```

```
db.students.updateMany({
  "courseList.name": "course2"
}, {
  $set: {
    "courseList.$.name": "Direito Penal"
  }
})
```

// [AddToSet] Adiciona um novo curso ao alunos

```
db.students.updateOne({
  "name": "Marcelo da Silva Santos"
}, {
  $addToSet: {
    "courseList": {
      "name": "Criptografia de Dados",
      "progress": "0%",
      "startDate": new Date()
    }
  }
})
```

// [In] Verifica se o aluno está inscrito em
// algum curso listado e atualiza o nome do curso

```
db.students.updateMany({
  "courseList.name": {
    $in: ["course3", "course4", "course5"]
  }
}, {
  $set: {
    "courseList.$.name": "Curso Polícia Federal"
  }
})
```

```

    }
  })

  // [Find] Devolve todos os alunos que terminaram todos os cursos
  db.students.find({
    "courseList": {
      $not: {
        $elemMatch: {
          "progress": {
            $ne: "100%"
          }
        }
      }
    }
  })

  // [FindOne] Devolve um curso que é híbrido (Online e presencial)
  db.students.findOne({
    "address": {
      $exists: false
    },
    "URL": {
      $ne: ""
    }
  })

  // [Gte] Devolve os cursos que tem preço acima de 10k
  db.courses.find({
    "price": {
      gte: 10000
    }
  })

  // [Exists] Devolve todos os cursos Online
  db.courses.find({
    "address.street": {
      $exists: false
    }
  })

```

```

    }
  })

// [All] Lista todos os cursos que tem monitor disponível em dias úteis
db.courses.find({
  "assistantList.availabilityList.day": {
    $all: [
      "Segunda-feira",
      "Terça-feira",
      "Quarta-feira",
      "Quinta-feira",
      "Sexta-feira"
    ]
  }
})

// [Aggregate] [Sum] [Sort] [Group] [Limit] Ordenar os monitores pelas
horas trabalhadas
db.courses.aggregate([
  $unwind: {
    path: '$assistantList',
  }, {
    $unwind: {
      path: '$assistantList.availabilityList',
    }, {
      $group: {
        _id: '$assistantList.name',
        hours: {
          $sum: '$assistantList.availabilityList.durationInMinutes'
        }
      }
    }, {
      $sort: {
        hours: -1
      }
    }
  ]
})

```

```

}, {
    $limit: 3
}] )

// [Search] [Text] Procura algum curso voltado para Pernambuco
db.courses.createIndex( { targetContests: "text" } )
db.courses.find( { $text: { $search: "Pernambuco" } })

// [Match] [Project] Todos os alunos acima de 18 anos
db.students.aggregate([
  $project: {
    name: 1,
    age: {
      $floor: {
        $divide: [{
          $subtract: [
            "$$NOW",
            "$birthDate"
          ]
        },
        60 * 60 * 24 * 30 * 12 * 1000
      ]
    }
  }
},
{
  $match: {
    age: {
      $gte: 18
    }
  }
}
]).pretty()

// [Function] [Pretty] Devolve uma mensagem do progresso do curso de
cada aluno
db.students.aggregate([

```

```

    $project: {
      name: 1,
      message: {
        $function:
        {
          lang: "js",
          args: ["$name", "$courseList"],
          body: function (name, courseList) {
            let total = 0;
            courseList.forEach(course => {
              if (course.progress === "100%")
                total += 1;
            });
            if (total === 0) {
              return `Ola ${name}. Você não concluiu nenhum
curso.`;
            }
            return `Ola ${name}. Você concluiu ${total}
curso(s).`;
          }
        }
      }
    }
  }
}

}).pretty();

// [Filter] [Cond] O nome do aluno e os cursos que o mesmo não terminou
db.students.aggregate([
  {
    $project: {
      name: "$name",
      course: {
        $filter: {
          input: "$courseList",
          as: "course",
          cond: {
            $ne: ["$$course.progress", "100%"]
          }
        }
      }
    }
  }
])

```



```

        }
    }
}
},
{
    $match: {
        course: {
            $gt: {
                $size: 0
            }
        }
    }
}
}
]).pretty()

// [Count] Quantidade de professores em cada curso
db.courses.aggregate([
    {
        $unwind: {
            path: '$teacherList'
        }
    },
    {
        $group: {
            _id: '$name',
            teacherCount: {
                $count: {}
            }
        }
    }
]);

// [Avg] A média do preço dos cursos
db.courses.aggregate([
    $group: {
        _id: null,
        mediaDosPrecos: {

```

```

        $avg: "$price"
    }
}
}]])

// [Lookup] Une os cursos aos estudantes de acordo com o nome do curso
db.students.aggregate([
    {
        $lookup: {
            from: "courses",
            localField: "courseList.name",
            foreignField: "name",
            as: "Courses"
        }
    }
]);

// [MapReduce] [Max] O preço máximo de cada curso
var mapFunction1 = function() {
    emit(this.name, this.price)
}
var reduceFunction1 = function(key, prices) {
    return Array.sum(prices)
}
db.courses.mapReduce(
    mapFunction1,
    reduceFunction1,
    { out: "map_reduce_function_1" }
)
db.courses.map_reduce_function_1.find()

```