

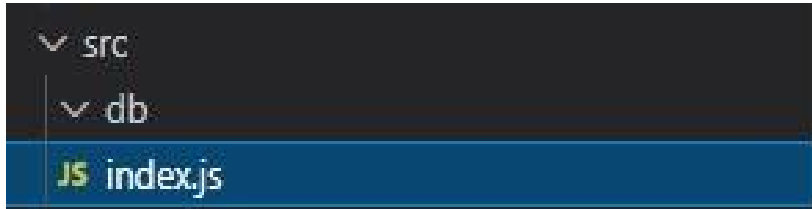


## AULA 08 - Vídeo 02

### Introdução

Retomando da última aula, o banco de dados é acessado constantemente e em caso de mudança de algum parâmetro podemos mudar direto do **.env**, assim deixando o código mais limpo. Nesta aula iremos organizar parte do código, e começamos criando uma pasta no “**src**” e por conseguinte um arquivo.

Exemplo:



Iniciaremos colocando toda parte de conexão do **.env** no Banco de Dados no arquivo criado.

```
require("dotenv").config() //import
const currentEnv = process.env

db.connect (

  `${currentEnv.DB_PROTOCOL}://${currentEnv.DB_USER}:${currentEnv.DB_PASSWORD}@${currentEnv.DB_HOSTY}/${currentEnv.DB_NAME}?${currentEnv.DB_OPTIONS}`

)
```

Uma vez feita a conexão inicial, devemos exportá-la para utilização. Primeiro colocamos o “**db.connect**” numa variável, depois criamos um **module exports** que irá conectar e pegar

a instância do banco de dados, assim deixando o acesso livre, porém a alteração fica dada somente para este arquivo.

```
const mongoose = require('mongoose')
require("dotenv").config() //import
const currentEnv = process.env

const URI = `
  ${currentEnv.DB_PROTOCOL}://${currentEnv.DB_USER}:${currentEnv.DB_PASSWORD}@${currentEnv.DB_HOSTY}/${currentEnv.DB_NAME}?${currentEnv.DB_OPTIONS}`

let db

module.exports = (
  connect: async () => {
    db = await mongoose.connect(URI)
    console.log("Conexao com o banco de dados estabelecida com sucesso!")
  },
  getDB: () => {
    return db
  }
)
```

Depois fazemos o tratamento no arquivo server com o “db valendo db” e o conectamos ao banco.