

Curso Técnico Superior Profissional em: Tecnologias e Programação de Sistemas de Informação

1.º Ano/ 2.º Semestre

Unidade Curricular: Desenvolvimento Web - Back-End

Docente: David Jardim

FICHA DE TRABALHO 5

Exercícios:

1. Crie uma pasta para a aula 5
 - a. Crie um ficheiro *server.js*
 - b. Instale a framework Express: <https://expressjs.com>
 - i. <https://expressjs.com/en/starter/installing.html>
 - c. Instale a aplicação Postman: <https://www.getpostman.com>
2. Após a instalação da framework Express efetue o exemplo seguinte:
 - a. <https://expressjs.com/en/starter/hello-world.html>
 - b. Utilize o Postman para testar o pedido GET
3. API REST
 - a. Crie um ficheiro *persons.json*
 - b. Adicione várias “pessoas” ao ficheiro tal como na aula passada com os seguintes atributos:
 - i. Id
 - ii. First Name
 - iii. Last Name
 - iv. Gender
 - v. Age
 - vi. Profession
 - c. Importe o file system do Node.js (<https://nodejs.org/api/fs.html>)
 - d. No *server.js* crie uma função para ler o ficheiro de forma síncrona
 - e. Tendo em conta a tabela 1, implemente os seguintes endpoints no *server.js*:
 - i. Listar todas as pessoas existentes no ficheiro e devolver na resposta
 - ii. Adicionar uma nova pessoa ao ficheiro, o ID deve ser gerado automaticamente tendo em conta o número de pessoas existentes. A nova lista deve ser devolvida na resposta.
 - iii. Apagar uma pessoa do ficheiro pelo seu ID. A nova lista deve ser devolvida na resposta.
 - iv. Selecionar apenas uma pessoa pelo seu ID e devolver essa mesma pessoa na resposta.
 - v. Editar os detalhes de uma determinada pessoa pelo seu ID e devolver essa mesma pessoa na resposta.

URI	Método HTTP	Body do POST	Resultado
/users	GET	empty	Show list of all the persons.
/users	POST	JSON String	Add details of new person.
/users/:id	DELETE	empty	Delete an existing person.
/users/:id	GET	empty	Show details of a person.
/users/:id	PUT	JSON String	Update details of a person.

Tabela 1 - Métodos a implementar

```
{
  "person1" : {
    "firstname" : "Pedro",
    "lastname" : "Lopes",
    "profession" : "teacher",
    "age" : "25",
    "id": 1
  },
  "person2" : {
    "firstname" : "André",
    "lastname" : "Faria",
    "profession" : "librarian",
    "age" : "35",
    "id": 2
  },
  "person3" : {
    "firstname" : "Ana",
    "lastname" : "Barros",
    "profession" : "clerk",
    "age" : "76",
    "id": 3
  }
}
```

Figura 1- Exemplo do ficheiro JSON