



Upload e  
download  
de arquivo

# Upload/Download



Upload significa o ato de enviar uma informação, geralmente um arquivo, para um computador remoto. Para isso utiliza-se um protocolo de transferência de dados, como o HTTP.

Download significa o ato de fazer cópia de uma informação, geralmente de um arquivo, que se encontra num computador remoto. Este processo também necessita de um protocolo de transferência de dados, como o HTTP.

```

class FormularioUpload extends Component {
  private arquivo: File
  constructor(props) {
    super(props)
    this.enviarArquivo = this.enviarArquivo.bind(this)
    this.receberArquivo = this.receberArquivo.bind(this)
  }

  <input type="file" onChange={(e) => this.receberArquivo(e)} />

  public receberArquivo(evento) {
    this.arquivo = evento.target.files[0]
  }

  public enviarArquivo(evento) {
    evento.preventDefault()
    if(!(this.arquivo === undefined)){
      let carregador = new CarregadorArquivo()
      carregador.carregar(this.arquivo)
      evento.target.reset()
      this.arquivo = undefined
    }
  }
}

```

Upload 

```

export default class CarregadorArquivo implements Carregador {
  carregar(arquivo: File): void {
    let dado = new FormData()
    dado.append('file', arquivo)
    fetch(URI.ENVIAR_ARQUIVO, {
      method: 'POST',
      body: dado
    })
  }
}

public receberArquivo(evento) {
  this.arquivo = evento.target.files[0]
}

public enviarArquivo(evento) {
  evento.preventDefault()
  if(!(this.arquivo === undefined)){
    let carregador = new CarregadorArquivo()
    carregador.carregar(this.arquivo)
    evento.target.reset()
    this.arquivo = undefined
  }
}

```

O objeto do tipo FormData encapsula o arquivo (binário), fornecido pelo usuário, para envio ao servidor.

Upload 

```
<a target={"_blank"} rel="noreferrer" href={arquivo['links'][0]['href']}
```

```
@GetMapping("/arquivo/{id}")
public ResponseEntity<Resource> obterArquivo(@PathVariable long id) {
    Resource recurso = armazem.obterArquivoComoRecurso(id);
    if (recurso == null) {
        ResponseEntity<Resource> resposta = new ResponseEntity<>(HttpStatus.NOT_FOUND);
        return resposta;
    } else {
        Arquivo arquivo = armazem.obterArquivo(id);
        MediaType tipoArquivo = MediaType.parseMediaType(arquivo.getTipo());
        ResponseEntity<Resource> resposta = ResponseEntity.ok().contentType(tipoArquivo).body(recurso);
        return resposta;
    }
}
```

A implementação do download depende muito da tecnologia back-end utilizada, contudo, existe um modo comumente aplicado.

O modo consiste em enviar à aplicação front-end informações, um json, contendo os endereços para acesso do recurso desejado. Assim, basta o usuário acessar o endereço para obter o recurso direto do servidor, onde está armazenado.

Download 

# Upload/Download



<https://github.com/gerson-pn/react-file-upload>

