* O que é o cliente? São os dispositivos que fazem pedidos de serviços ou recursos a um servidor. O cliente geralmente acessa esses serviços e recursos através de um navegador web. O cliente gera requisição e o servidor a resposta.
* O que é o servidor? É o computador que fornece os serviços e recursos aos quais os clientes têm acesso. Programa no servidor que processa requisições e envia responses
* Request: é a requisição que realizamos através do browser (o cliente) para um servidor.
  + Request -> Headers-> URL/Dominio->DNS(sincronizando o nome do domínio com o ip da máquina servidor
  + Headers - > method: get, post, delete, put
  + Request -> body: as informações usadas com post são mascaradas, viajam com mais segurança
* Response: o servidor recebe o nosso request, processa-o e envia como resultado uma resposta ao cliente.
* FRONT-END: é tudo aquilo que acontece no lado do cliente. HTML, CSS, JS.
* BACK-END: é tudo o que acontece no lado do servidor. MySQL, PHP, JAVA
* Node é uma plataforma Js para o Back-End, não é uma linguagem. Ele atua no servidor.
* NPM: Node Package Manager: grande biblioteca de soluções para o Node.
  + Essas bibliotecas podem ser instaladas no meu computados globalmente, ou localmente, sendo disponível apenas em um projeto especifico.
  + Npm init: inicializa um projeto node -> cria um arquivo package.json que guarda as configurações do projeto
* Módulo do node: um bloco de código reutilizável, uma unidade de funcionalidade cuja existência não altera incidentalmente o comportamento de outros blocos de código.
  + Request(nomeDoModulo): retorna as funcionalidades do módulo dentro de uma variável. Um objeto literal é criado dentro dessa variável
* HTTP: Hyper text transfer protocol. Protocolo que administra as transações web entre cliente-servidor
* URI: endereços de web, composto pelo URL(que indica onde se encontra o recurso que queremos obter e começa sempre com um protocolo, o http) e pelo URN(o nome exato de recurso uniforme)
  + Get: para solicitar e recuperar somente dados (quando digitamos o endereço no navegador)
  + Post: envia dados para o servidor, é mais seguro que o get
  + Put: envie dados, substitua todas as informações de um recurso
  + Patch: envia dados, modifica informações parcial de um recurso
  + Delete: envia uma solicitação para eliminar um recurso do servidor