

Módulos

No node, cada arquivo javascript é considerado um módulo. Nos módulos você irá guardar objetos, funções, variáveis, diversas coisas que fazem sentido pra sua aplicação.

Quando você cria um módulo no node com uma função, ela irá ficar privada para aquele módulo, caso você deseje utilizar essa função em outro módulo, você terá que usar o método `module.exports`; **Sempre sendo a última diretiva!**

Exemplo: `user.js`

```
const getUserById => id => {  
  async buscando no banco...  
  return user;  
};  
  
module.exports = { getUserById };
```

Agora no meu arquivo `index.js` eu quero utilizar essa função existente em `user.js`, então no início do meu `index` eu irei importar o meu módulo `user.js`

```
const user = require('./user.js');
```

Event Loop

O event loop é o coração do node.js. Falamos que o node é single thread (que é o event loop). Quando chega uma requisição para o servidor é disparado um evento (o nodejs é baseado em eventos), quando essa requisição chega ela entra em uma fila de processo, se ela for a primeira da fila ela vai direto para o event loop e ele faz uma pergunta (NÃO TÃO SIMPLES ASSIM, MAS PARA ENTENDIMENTO BASTA)

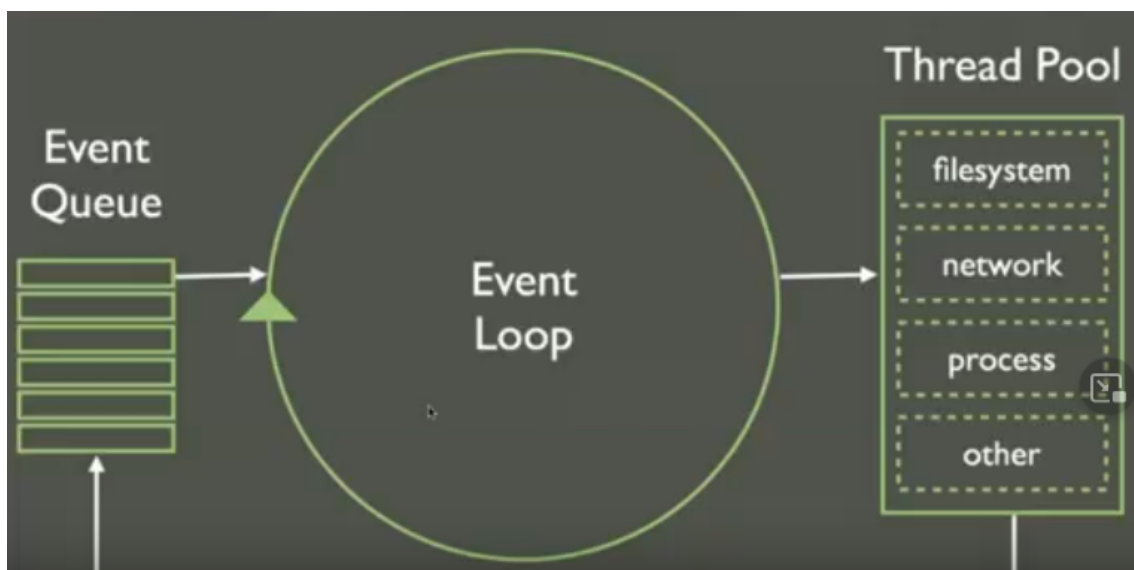
Isso é rápido para eu processar?

Caso ele fale que sim, ele processa os dados e retorna pro usuário o que foi pedido na requisição.

Caso ele fale que não, ai ele passa para uma pool de threads em background. **Ai cai a casa de quem fala que node é single thread**

Thread Pool

- Filesystem
- process
- network
- other



Características

- O motor do javascript V8 é single thread
- libuv (C++) Multi thread, essa é a pool thread que fica em background no node