

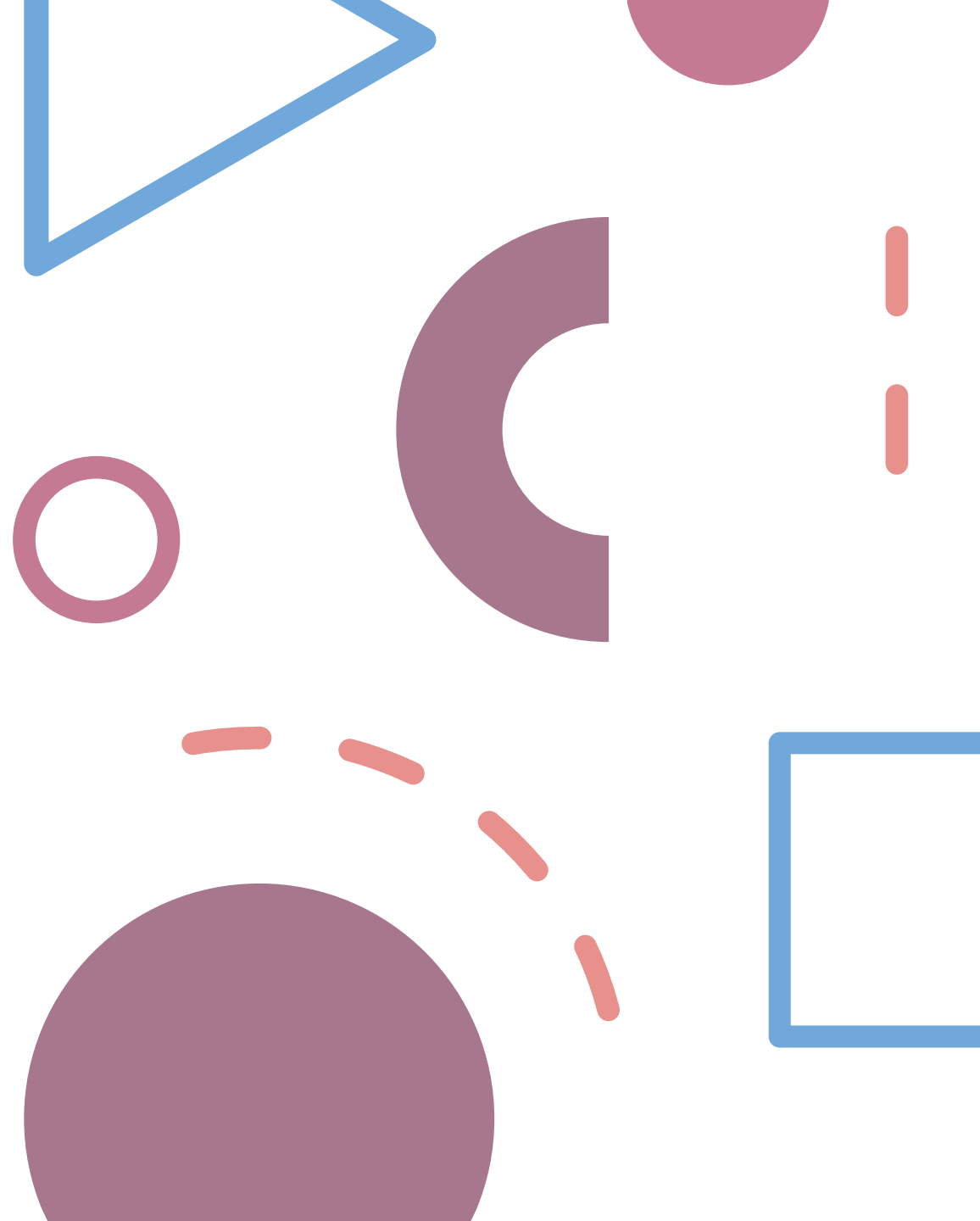
Node js

Av Anjani



Content

- General
 - What is Node js?
 - Why use Node js?
 - Non-blocking I/O model
- Installation
 - Windows
 - Mac
 - Linux
- REPL
- NPM
- Modules
- Basic server
- Frameworks
- Template Engines



What is Node js?

- JavaScript Runtime
- Built on V8 JavaScript Engine (Developed by Chrome)
- Written in C++
- Allows JavaScript on server
- Used for non-CPU intensive apps (APIs, CRUD apps)

Why use Node.js?

- Same language front and back
- Fast, efficient, and scalable
- Event driven
- Non-blocking I/O model
- Third-party packages

Non-blocking I/O model

- Input/Output calls
- Single thread
- Asynchronous
- Support for many connections
- Optimizes scalability

Installation: Windows

- Install .msi file from <https://nodejs.org/en/download/>
- Double-click on installer and go through installation
- Run "node -v" in terminal to confirm installation success

Installation: Mac

- Install .pkg file from <https://nodejs.org/en/download/>
- Double-click on installer and go through installation
- Run "node -v" in terminal to confirm installation success

Installation: Linux

- Ubuntu, Debian, Mint:
 - `sudo apt install nodejs npm`
- CentOS(8 - latest), Fedora, Red Hat(latest):
 - `sudo dnf install nodejs`
- CentOS(6, 7, 8), Red Hat(old):
 - `sudo yum install epel-release`
 - `sudo yum install nodejs`
- Arch, Manjaro:
 - `sudo pacman -S nodejs`
- Run "node -v" in terminal to confirm installation success

REPL

- Read-Eval-Print-Loop
- Specifies environment to run code
- Run "node" in terminal to start REPL

NPM

- Node Package Manager
- Third-party packages
- Saved in "node_modules" folder
- Saved as dependencies in "package.json" file
- Scripts to run on server

Modules

- Core modules (path, fs, http...)
- Third-party packages
- Custom modules (files)
- CommonJS (require({package-name}))
 - Conventions outside web

Basic server

```
const http = require('http');

http.createServer((req, res) => {
  if(req.url === '/' && req.method === 'GET'){
    res.write('HOME');
  } else if(req.url === '/about' && req.method === 'GET'){
    res.write('ABOUT');
  } else if(req.method === 'GET'){
    res.write('404');
  }
  res.end();
}).listen(3000, 'localhost', () => console.log('Server running on port 3000'));
```

Frameworks

- Libraries/tools
- Extend properties/functionality
- Productive, scalable, fast
- Express, Koa, Nest, Adonis...

Template Engines

- HTML Templates
- Inject data
- Partials
- Layouts
- EJS, Jade, Handlebars...