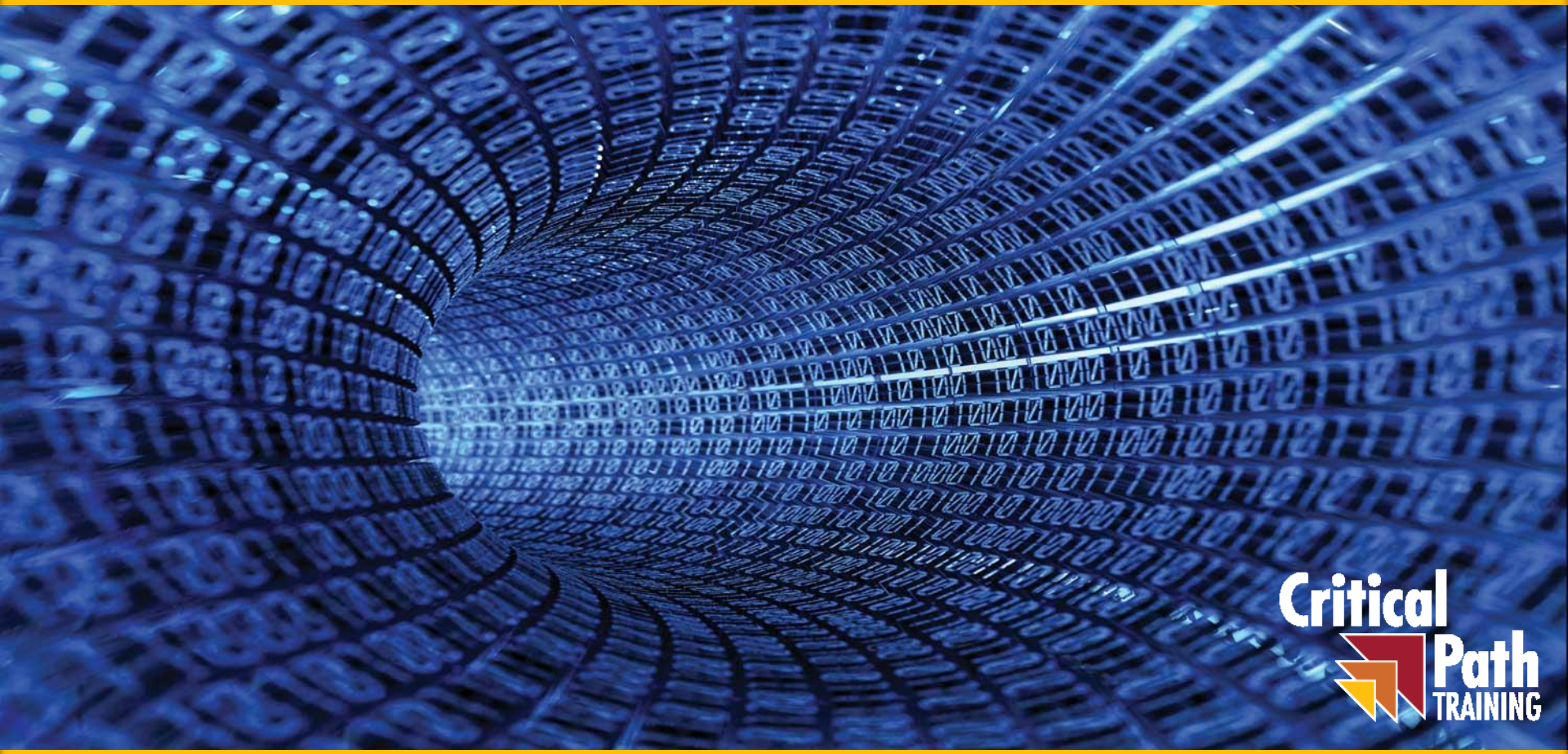
Getting Started with the Power BI Developer Tools



Agenda

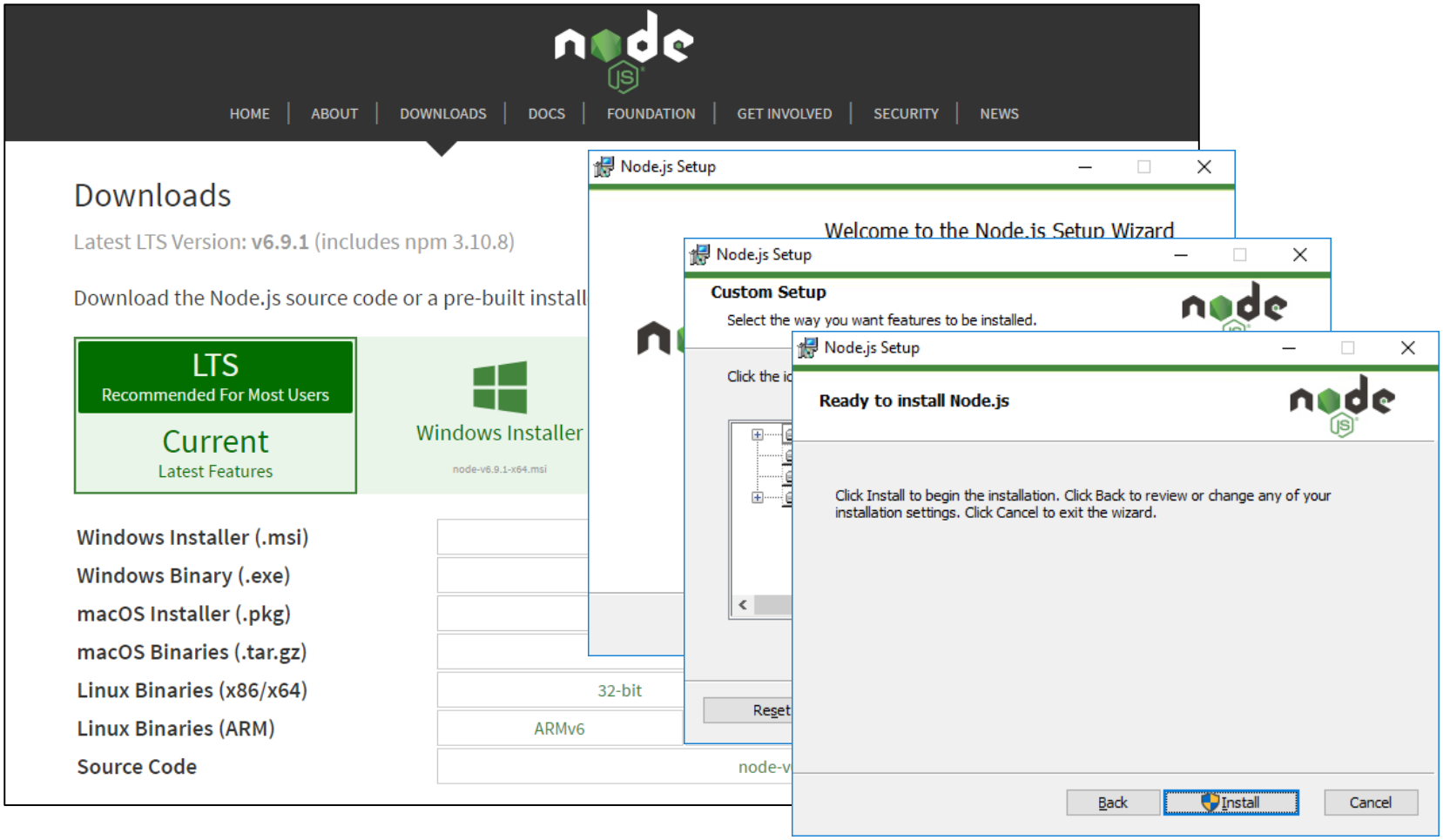
- Custom Visuals in Power BI
- Node.JS and the Cross-platform Toolchain
- Creating Projects with the PBIVIZ CLI
- Custom Visual Project Structure
- Adding Typed Definition Files
- Testing and Debugging a Custom Visual

Install the Power BI Developer Toolchain

- Install Node.JS
  - Installs Node Package Manage (npm)
- Install Power BI visuals CLI tool (pbiviz)
  - Install using Node Package Manager (npm)
- Install Local self-signed certificate
  - Install using Power BI visuals CLI tool (pbiviz)
- Install Visual Studio Code
  - Lightweight Alternative to Visual Studio for Node.js Development

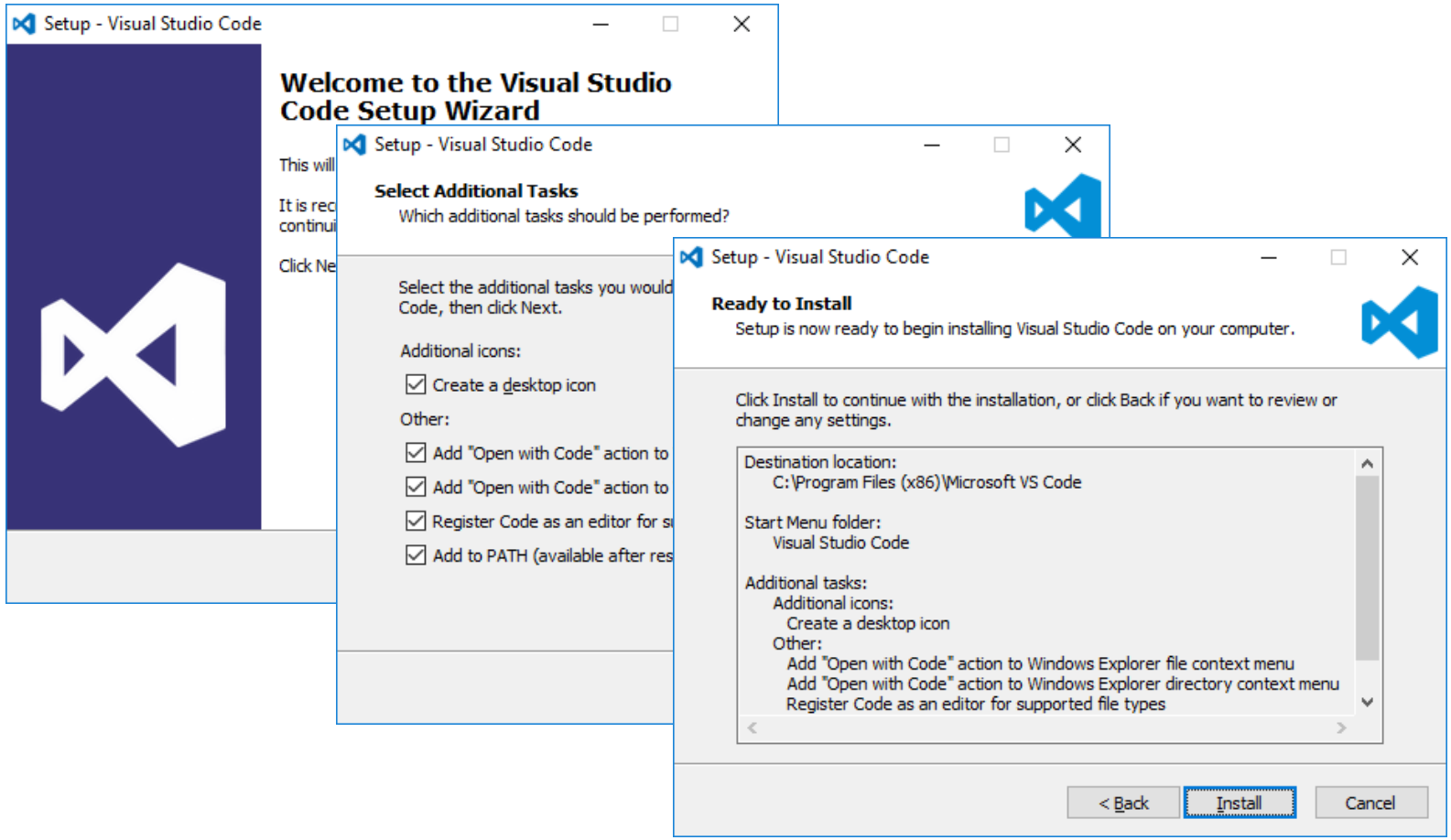
Installing node.js

- https://nodejs.org/en/download/



Install Visual Studio Code

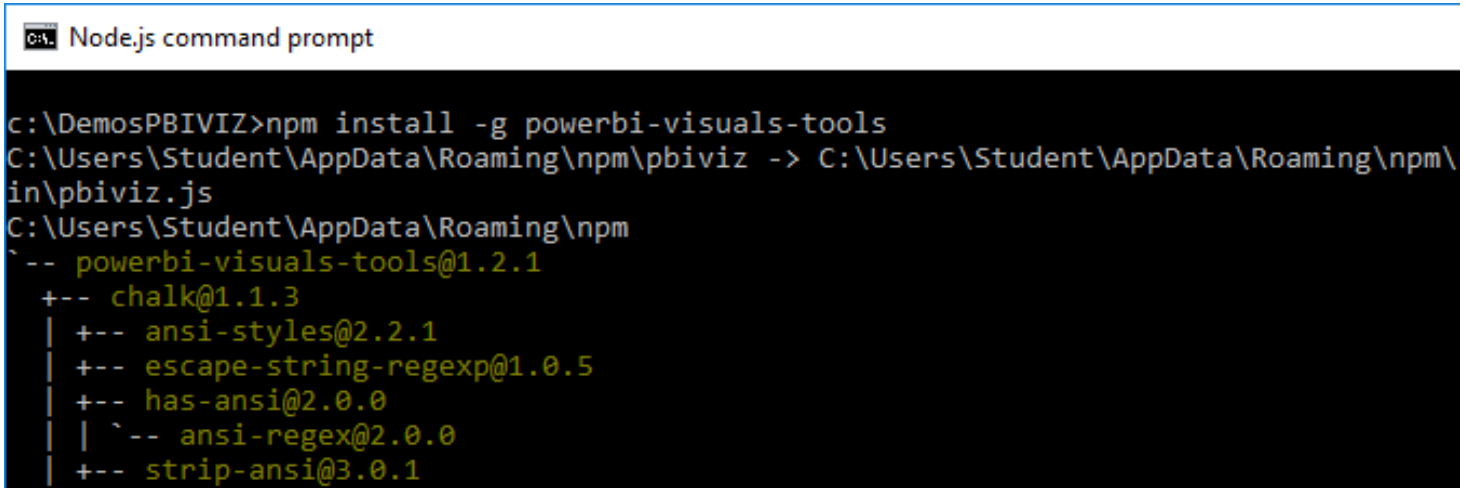
- http://code.visualstudio.com/



Power BI Visual CLI Tool (PBIVIZ)

- What is the Power BI Custom Visual Tool?
  - Command-line utility for cross-platform dev
  - Use it with Visual Studio or Visual Studio Code
  - Requires that you first install node.js
  - Install by running command from node.js command prompt

```
npm install -g powerbi-visuals-tools
```



Power BI Visual Tool Dependencies

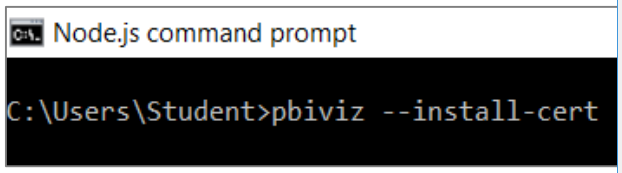
```
- ms@0.7.2 node_modules\powerbi-visuals-tools\node_modules\send\node_modules\ms
C:\Users\Student\AppData\Roaming\npm
`-- powerbi-visuals-tools@1.5.0
  +-- connect@3.6.0
  | +-- debug@2.6.1
  | | `-- ms@0.7.2
  | `-- finalhandler@1.0.0
  +-- fs-extra@0.28.0
  | `-- rimraf@2.6.0
  +-- gulp-powerbi-package-validator@1.0.0
  | +-- eslint@3.16.1
  | | +-- concat-stream@1.6.0
  | | | `-- readable-stream@2.2.3
  | | `-- ignore@3.2.4
  | `-- gulp-stylelint@0.2.0
  |   +-- postcss@5.2.15
  |   `-- stylelint@4.5.1
  |     +-- autoprefixer@6.7.5
  |     | +-- browserslist@1.7.5
  |     | | `-- electron-to-chromium@1.2.3
  |     | `-- caniuse-db@1.0.30000626
  |     `-- stylehacks@2.3.2
  |       `-- postcss-selector-parser@2.2.3
  `-- serve-static@1.11.2
    `-- send@0.14.2
      `-- debug@2.2.0
        `-- ms@0.7.1
```

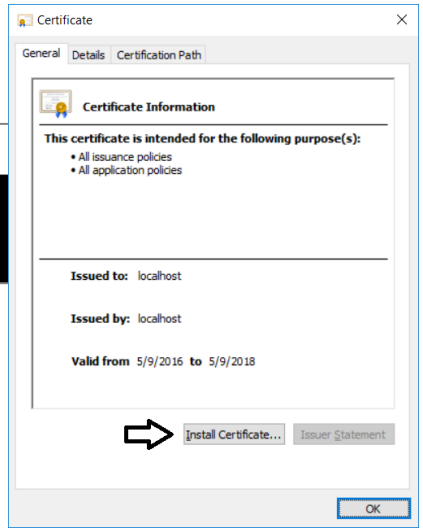
Getting Started with PBIVIZ

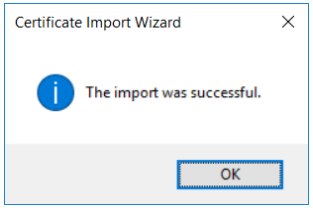
```
Usage: pbiviz [options] [command]


Commands:

  new [name]        Create a new visual
  info              Display info about the current visual
  start             Start the current visual
  package           Package the current visual into a pbiviz file
  update [version]  Updates the api definitions and schemas in the current visual. Changes the version if specified
  help [cmd]        display help for [cmd]

Options:

  -h, --help      output usage information
  -V, --version   output the version number
  --install-cert  Install localhost certificate


:\Demo\PBI\simplebarchart>
```

Installing the Developer Certificate

- Debugging visuals inside PowerBI.com requires SSL
  - PBIVIZ leverages Node.js to provide debugging experience
  - Node.js acts as web service to serve project files through HTTP
  - Node.js debugging session uses http://localhost address
  - Installing certificate enables SSL through https://localhost
  - Installing certificate is a one time operation – not once per project







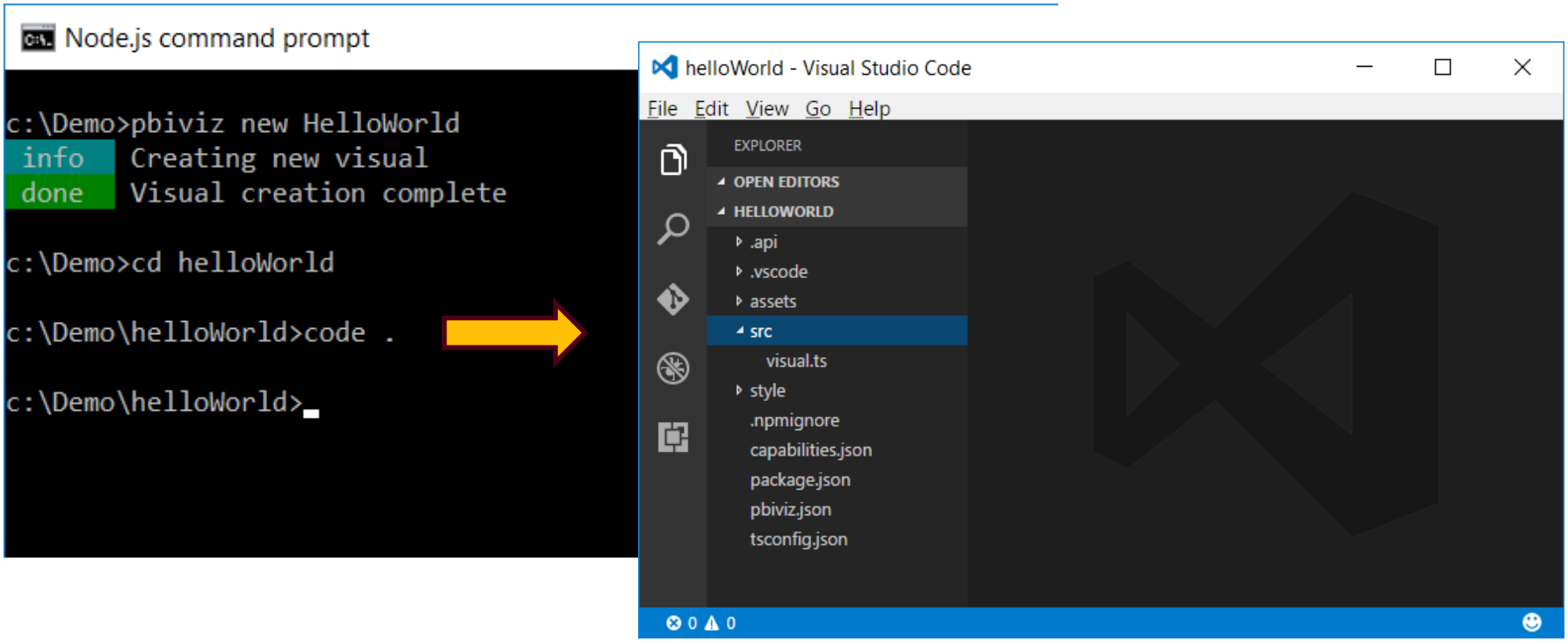
Creating a New Custom Visual Project

- Creating a new project

  ```
  pbiviz new <ProjectName>
  ```

- Open the Project with Visual Studio Code

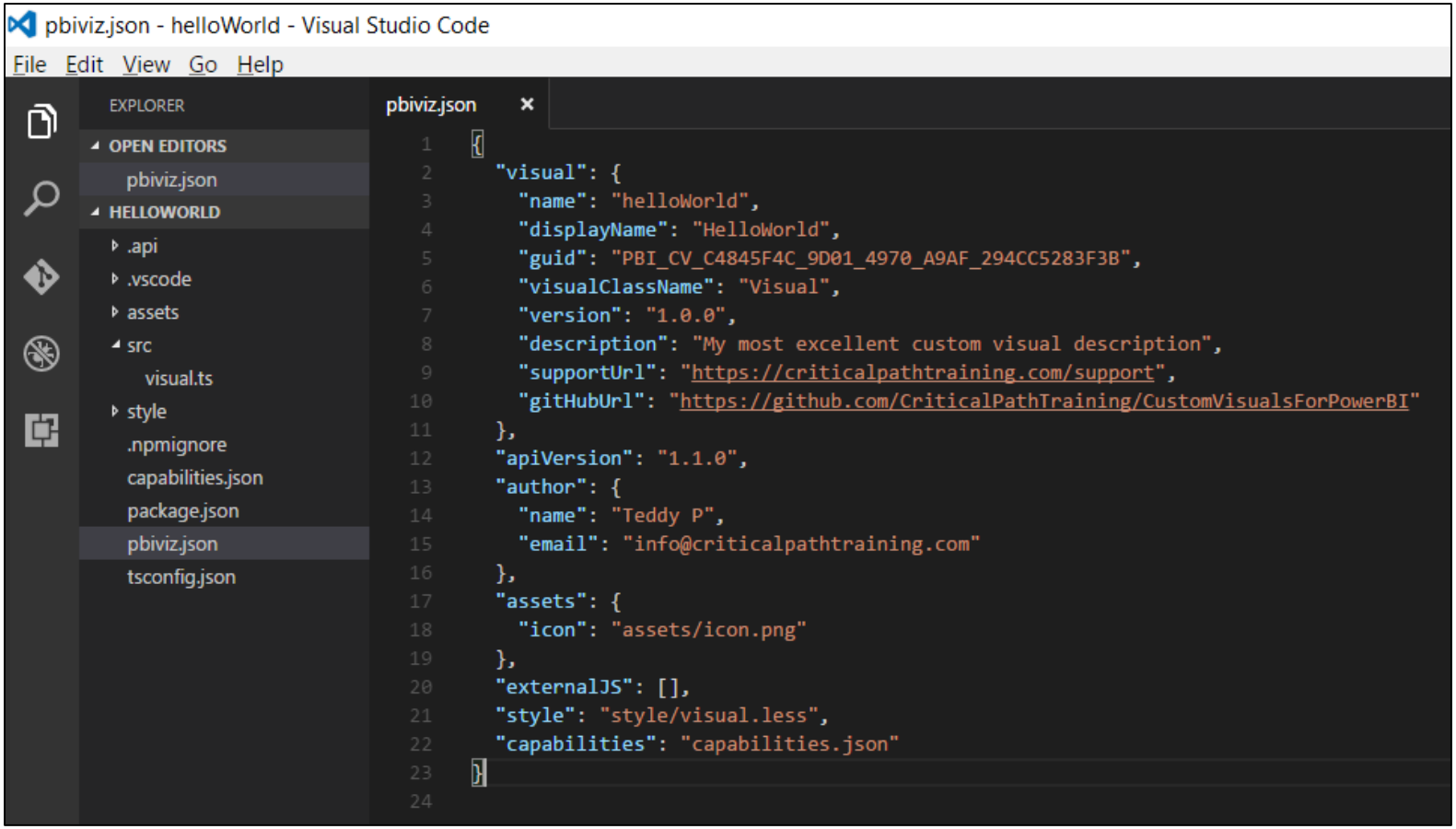
  ```
  code .
  ```



Files in the new project

- gitignore
  - tells git to ignore files that shouldn't be tracked in the repository
- capabilities.json
  - used to define the capabilities of your visual learn more about visual capabilities
- package.json
  - Used by npm to manage modules learn more about npm
- pbiviz.json
  - Main configuration file for your visual
- tsconfig.json
  - Typescript compiler settings learn more about tsconfig

The pbiviz.json File



Folders in the new project

- assets/
  - Used to store visual assets (icon, screenshots, etc)
- dist/
  - when you run pbiviz package the pbiviz file will be generated here
- src/
  - Typescript code for your visual goes here
- style/
  - Less styles for your visual go here

Installing Typings for D3 using npm

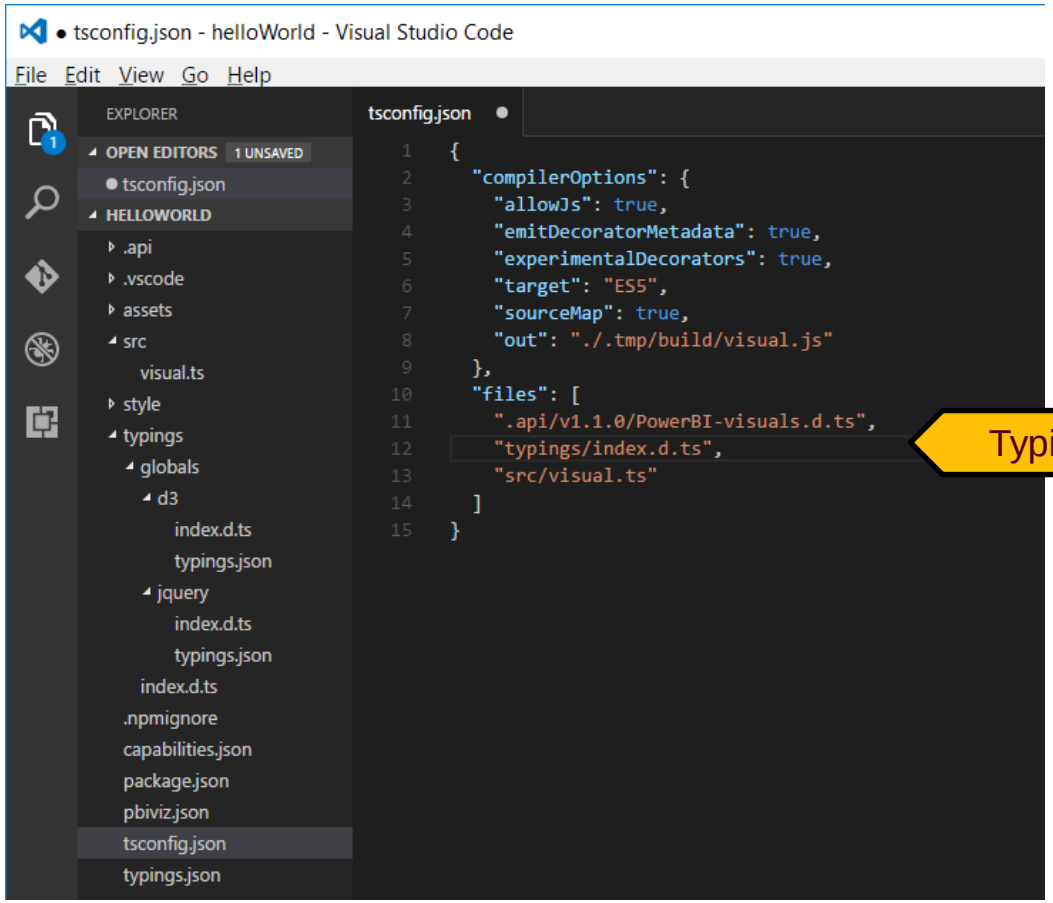
- Installing typings support using Node Package Manage
  - Start by installing global support for typings

    npm install typings –g
  - Install typings for specific JavaScript libraries

    typings install --save --global dt~d3#0.0.0+20160907005744

The tsconfig.json File

- Used to add references to typings files
  - This is what enables Intellisense

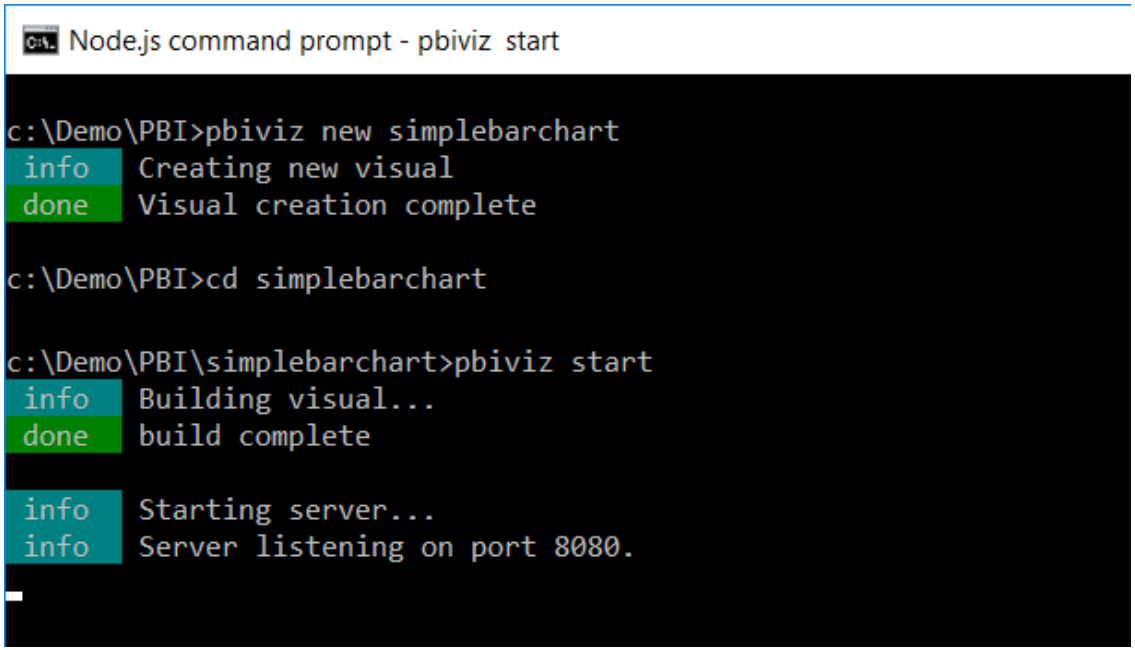


Developing a Custom Visual?

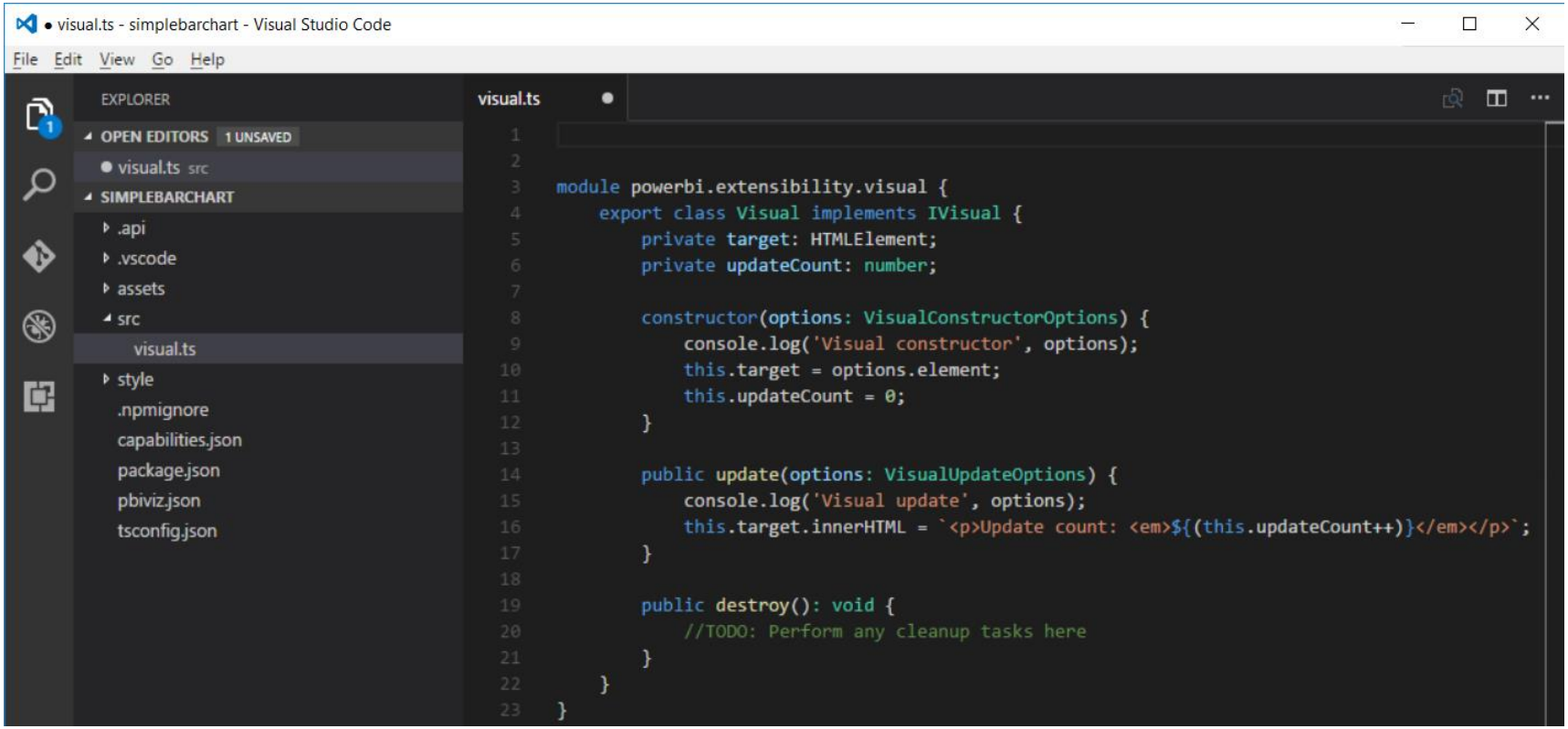
- Create a class that implements IVisual
  - Class wrapped in module with namespace to APIs
  - You code can program again PBI APIs types

```
module powerbi.extensibility.visual {

    export class Visual implements IVisual {

        constructor(options: VisualConstructorOptions) {
            // one-time initializaion code
        }

        public update(options: VisualUpdateOptions) {
            // called when viewport or data changes
        }

        public destroy(): void {
            // add cleanup code here
        }
    }
}
```

Running a Custom Visual Project



Developing with Visual Studio Code



Summary

- ✓ Custom Visuals in Power BI
- ✓ Node.JS and the Cross-platform Toolchain
- ✓ Creating Projects with the PBIVIZ CLI
- ✓ Custom Visual Project Structure
- ✓ Adding Typed Definition Files
- ✓ Testing and Debugging a Custom Visual