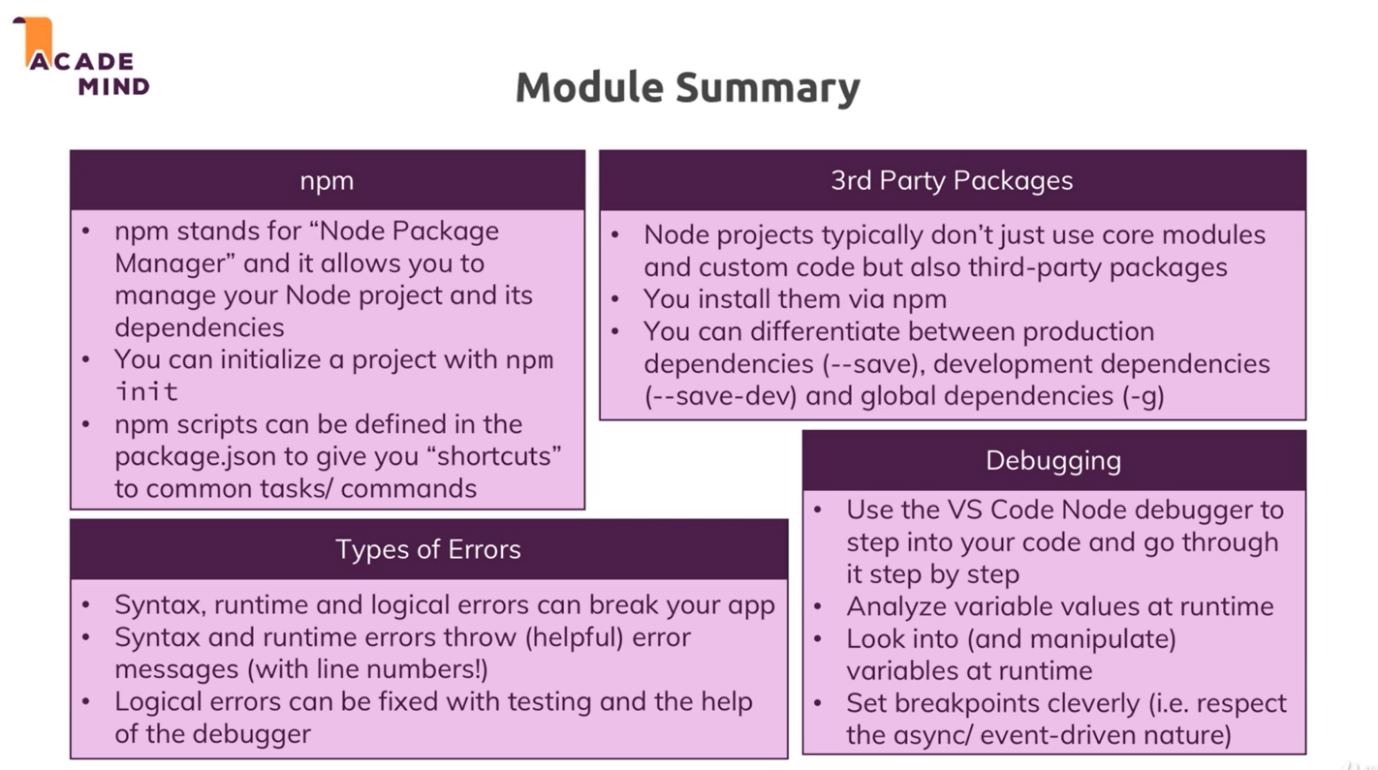
**Debugging & Easier Development**

**Summary**

1. **Understanding NPM scripts**a. Till now we run node app.js command to run your application with that file.  
   b. we use npm to write scripts to perform some task. It is also used to install 3rd party packages. (we will use npm to initialize your project and to add some feature to it)  
   c. to initialize npm in your project run command in your node application run: npm init. this will add **package.json** file in your project.  
   d. package.json is a configuration file of your project, in this configuration file we add some scripts by added in script section of file.  
   e. let add, “start” : “node app.js” : start is a special script, due to this if we share project to another person if he/she want to run application he/she want to run npm start command and they no need to find entry point of the application.  
   f. let add, “start-server” : “node app.js” , now we try to run npm start-server it will give an error(this is not know command), this work for npm start as start is special script, to run this you have yo write npm run start-server
2. **Installing 3rd party packages**a. all 3rd party packages are available in npm repository and we can install & manage by nmp.  
   b. reason to add 3rd party packages is we want to use some predefine modules which are not present in core modules to node .  
   c. Till now, we have change any file in project we have kill server and start it again to apply changes. This can be fix by installing 3rd part package called as **nodemon**. We can install it by using command in project: npm install nodemon –save-dev.  
   d. there are 2 types of dependency: 1. For development (which will speed-up development & not use in project) and 2. For production (used in project running on server)  
   to install packages in development:   
   npm install nodemon --save-dev  
   to install packages in production:   
   npm install nodemon --save  
   to install packages globally:   
   npm install nodemon -g  
     
   e. this will add a node\_modules folder and package-lock.json file in your project & will add dependencies in package.json file.   
   node\_modules will contain all packages.   
   package-lock.json will contain information of dependencies.  
   f. we can delete node\_modules to free-up space. Can re-install by writing command: npm install which will install all dependencies of your project.
3. **Using Nodemon for Auto-restarts**a. to-do so, we have change start script of package.json by  
   “start” : “nodemon app.js”.  
   b.if we try to run nodemon app.js in your terminal it throw an error reason we had installed nodemon locally not globally.  
   c. now if we run npm start, we can see auto-restart while we save file.
4. **Understanding different Error types**Syntax Error, Run-time Error, Logical Error
5. **Finding & Fixing syntax error**this type of error will crash your application and display error in console.
6. **Dealing with Run-time error**this will not crash your application, but when error occurred will display it on console and show this site is not working on web.
7. **Logical error**using node.js debugger. Run application in debugging mode in vs code.  
   and using breakpoint we can debug.