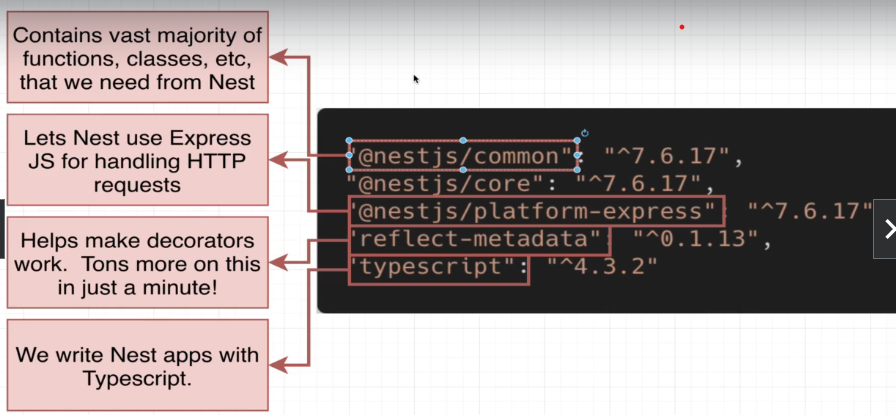
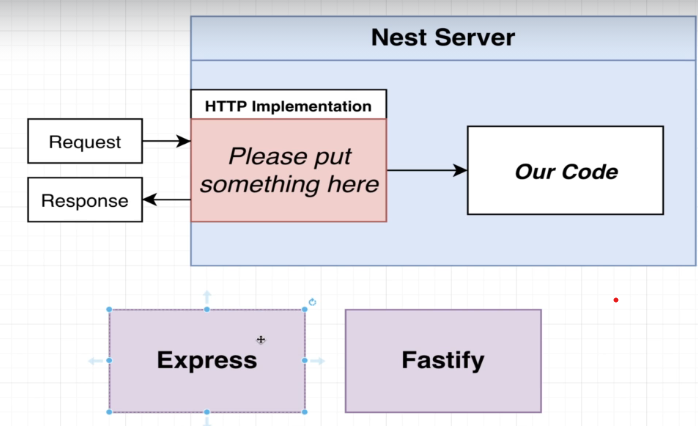
4. Project Setup

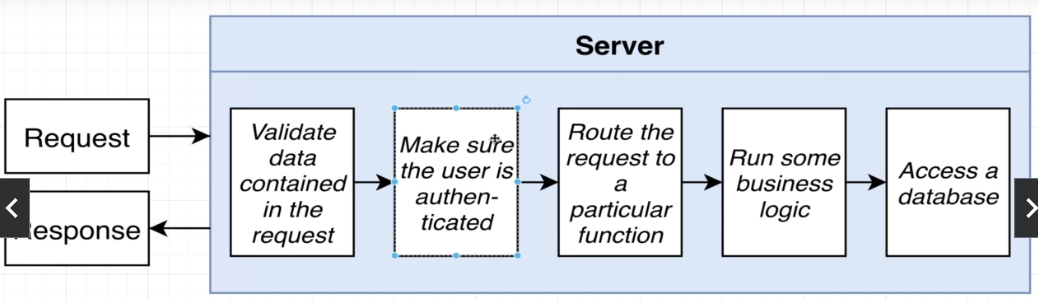
5. TypeScript Configuration

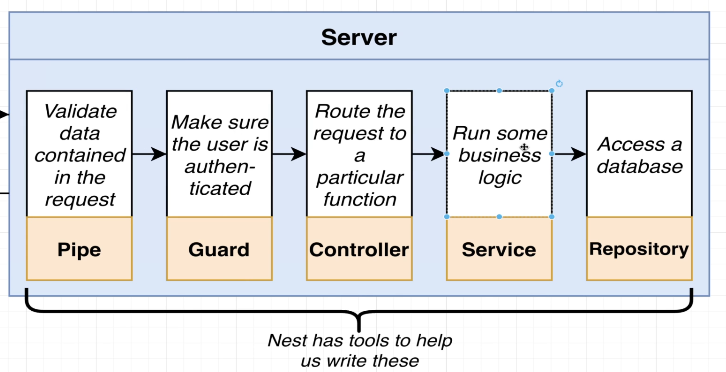
6. Creating a Controller

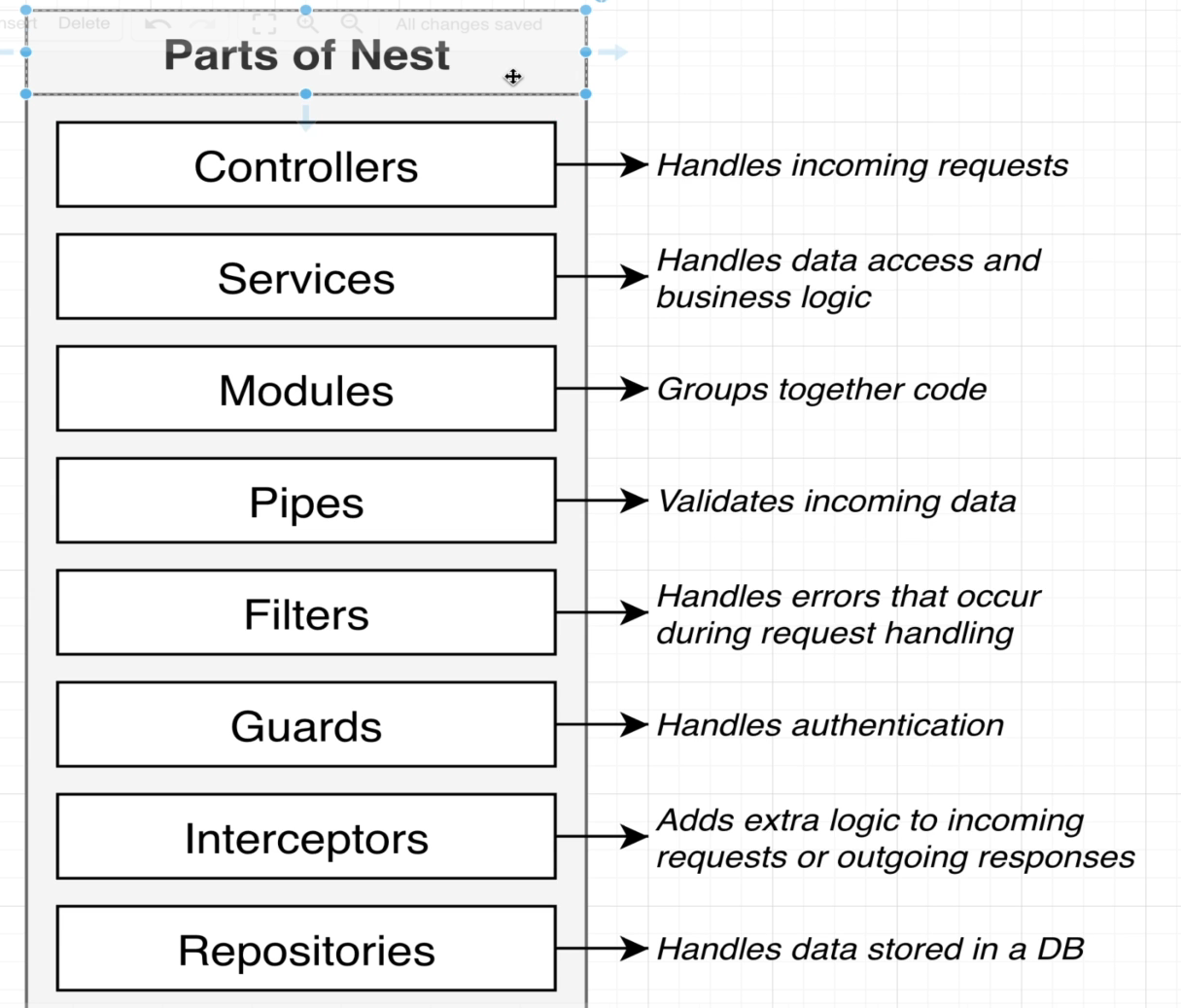


npm i @nestjs/common@7.6.17 @nestjs/core@7.6.17 @nestjs/platform-express@7.6.17 reflect-metadata@0.1.13 typescript@4.3.2





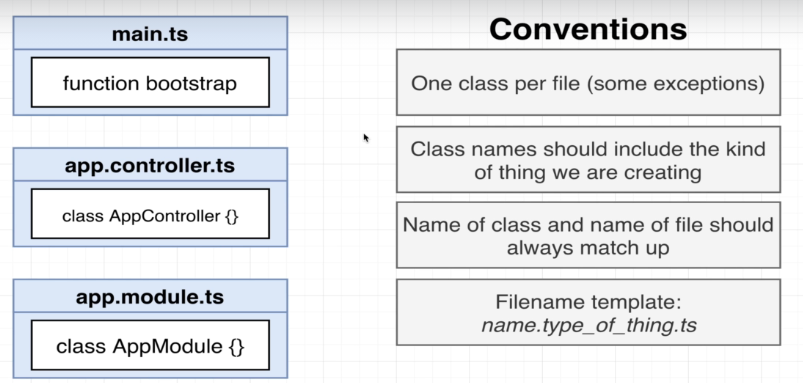




7. Starting Up a Nest App

npx ts-node-dev .\src\main.ts

8. File Naming Conventions

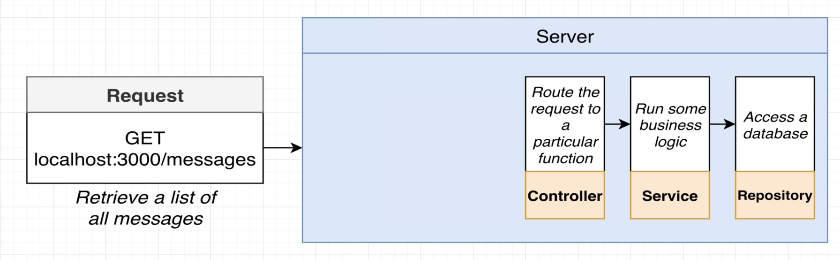


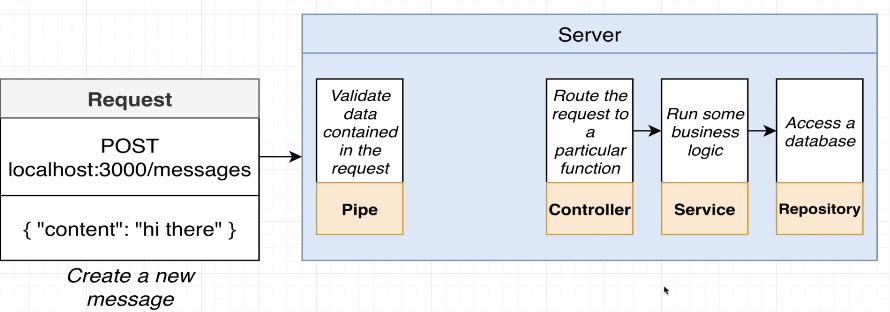
9. Routing Decorators

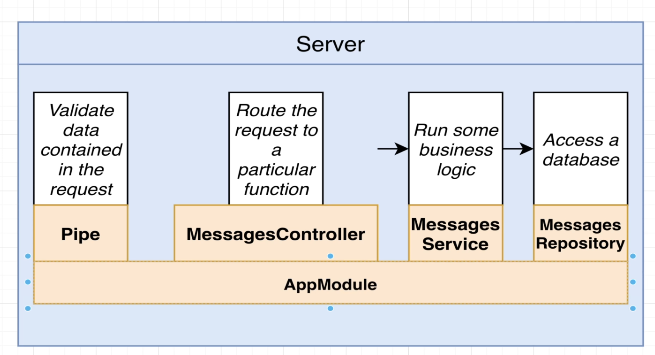
Section 3: Generating Projects with the Nest CLI

10. App Setup

npm install -g @nestjs/cli



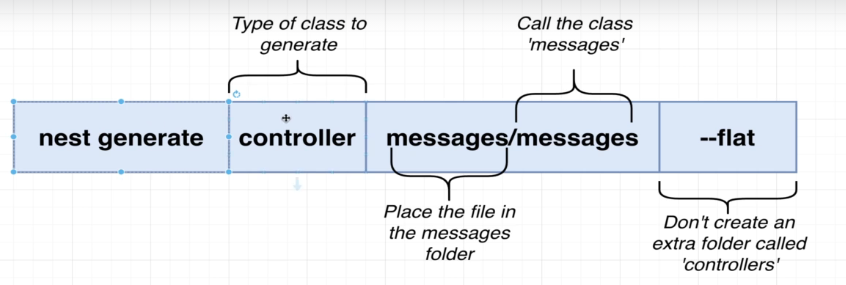




11. Using the Nest CLI to Generate Files

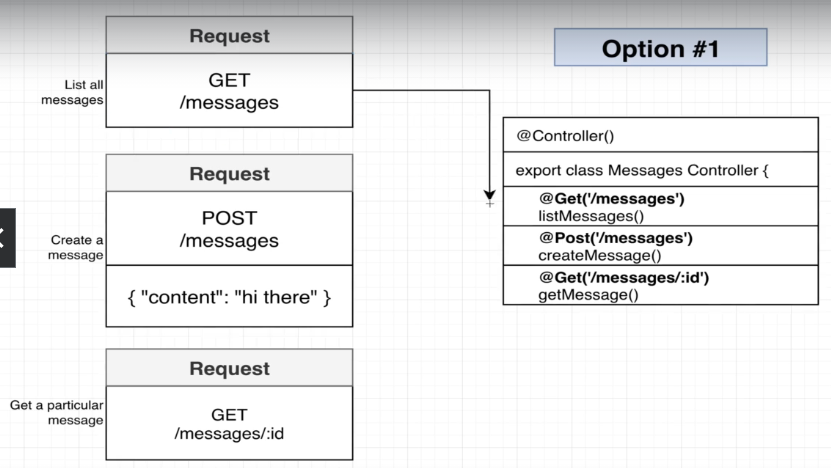
nest generate module messages

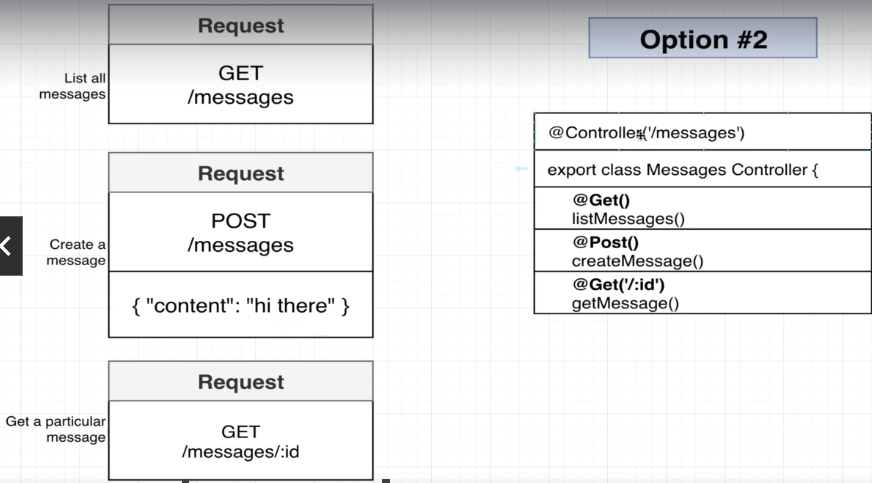
nest generate controller messages/messages –flat



12. More on Generating Files

13. Adding Routing Logic





14. [Optional] Postman Setup

15. [Optional] VSCode REST Client Extension

Create rest.http in root

### List all messages

GET http://localhost:3000/messages

### Create new message

POST http://localhost:3000/messages

content-type: application/json

{

    "content": "hi there"

}

### Get a particular message

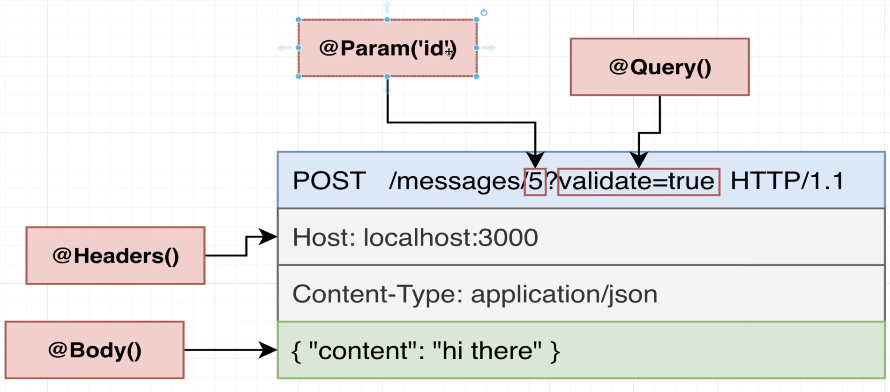
GET http://localhost:3000/messages/123

Section 4: Validating Request Data with Pipes

16. Accessing Request Data with Decorators

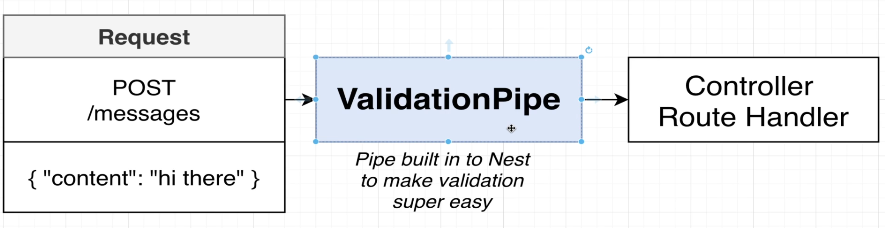
Get and Post decorators 🡪 method decorators

Param and Query 🡪 body decorators

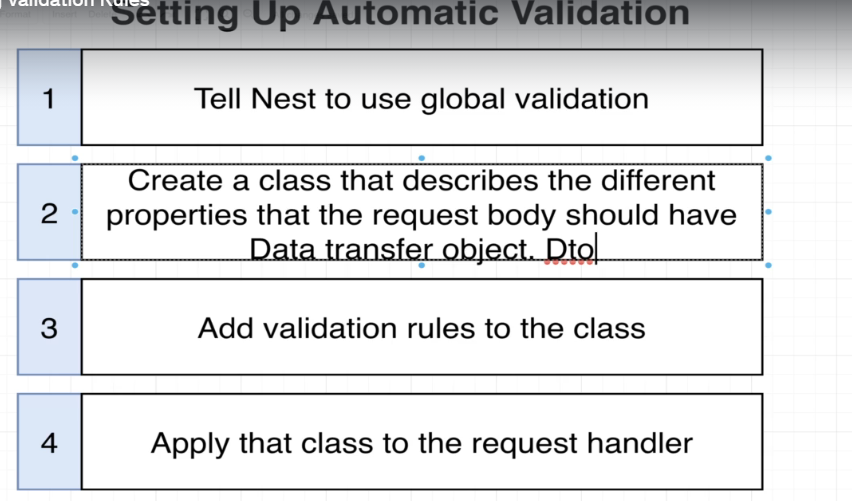


17. Installing Extra Required Libraries

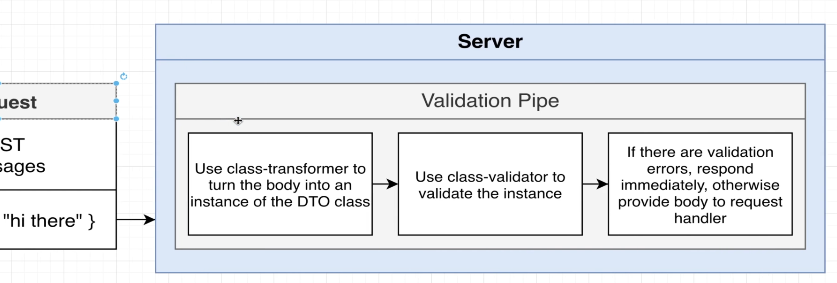
18. Using Pipes for Validation



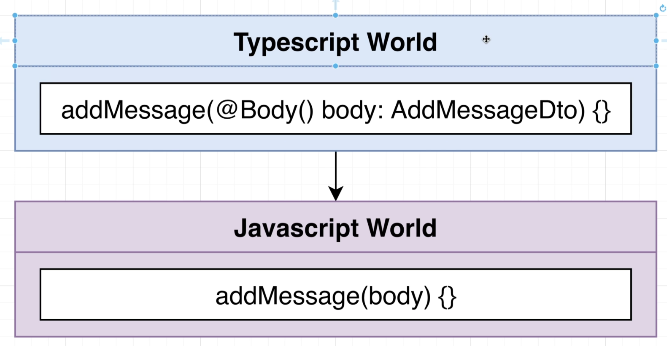
19. Adding Validation Rules



20. Behind the Scenes of Validation

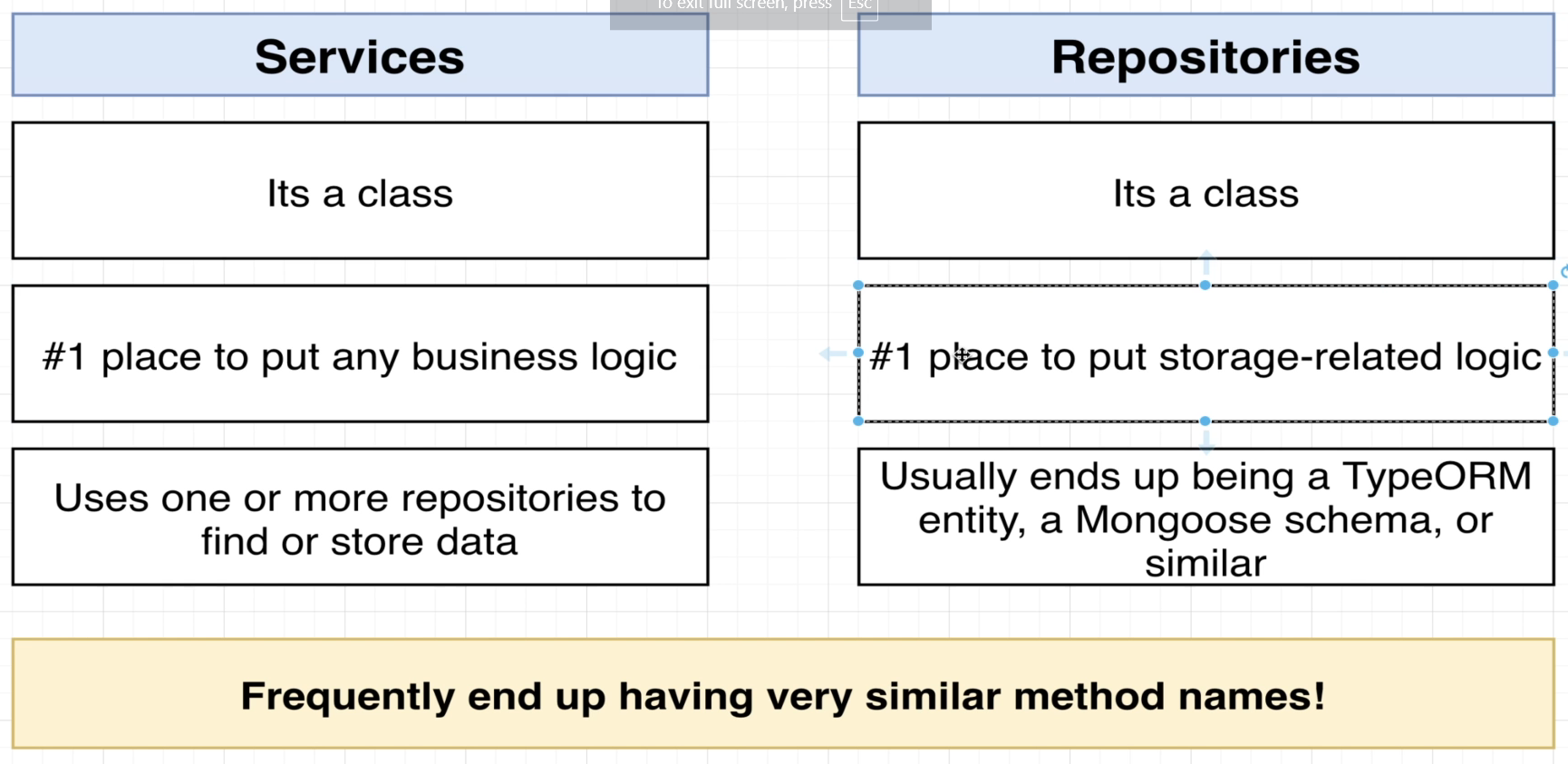


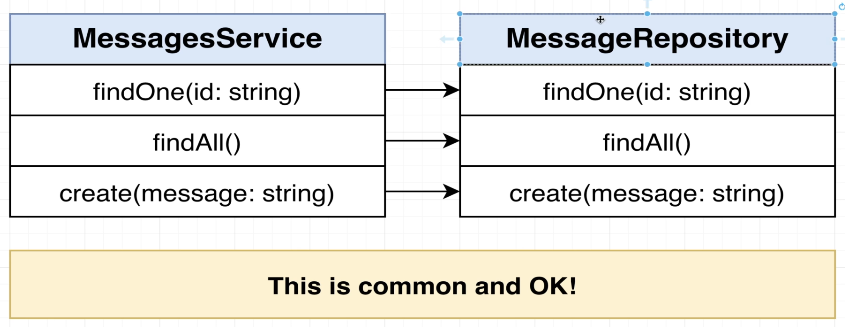
21. How Type Info is Preserved



Section 5: Nest Architecture: Services and Repositories

22. Services and Repositories





23. Implementing a Repository

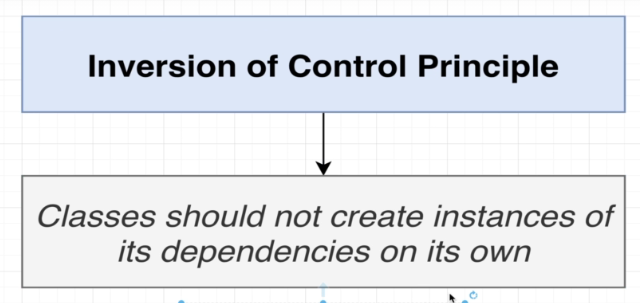
24. Reading and Writing to a Storage File

25. Implementing a Service

26. Manual Testing of the Controller

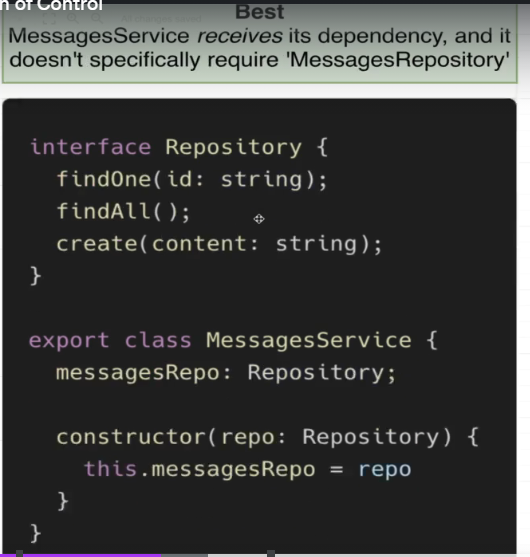
27. Reporting Errors with Exceptions

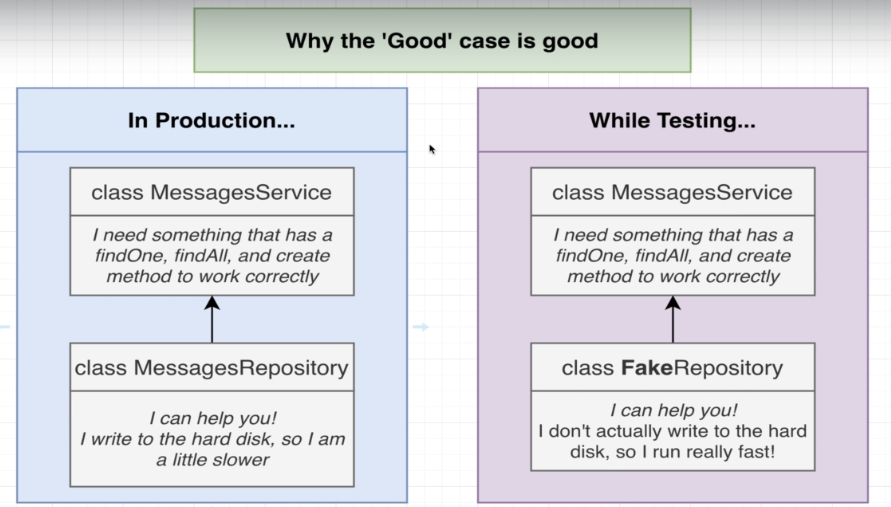
28. Understanding Inversion of Control











29. Introduction to Dependency Injection

30. Refactoring to Use Dependency Injection

31. Few More Notes on DI

Section 6: Nest Architecture: Organizing Code with Modules

32. Project Overview

33. Generating a Few Files

34. Setting Up DI Between Modules

35. More on DI Between Modules

36. Consuming Multiple Modules

37. Modules Wrapup

Section 7: Big Project Time!

38. App Overview

39. API Design

40. Module Design!

41. Generating Modules, Controllers, and Services

Section 8: Persisting Data with TypeORM

42. Persistent Data with Nest

43. Setting Up a Database Connection

44. Creating an Entity and Repository

45. Viewing a DB's Contents

46. Understanding TypeORM Decorators

47. One Quick Note on Repositories

48. A Few Extra Routes

49. Setting Up Body Validation

50. Manual Route Testing

Section 9: Creating and Saving User Data

51. Creating and Saving a User

52. Quick Breather and Review

53. More on Create vs Save

54. Required Update for find and findOne Methods

55. Querying for Data

56. Updating Data

57. Removing Users

58. Finding and Filtering Records

59. Removing Records

60. Updating Records

61. A Few Notes on Exceptions

Section 10: Custom Data Serialization

62. Excluding Response Properties

63. Solution to Serialization

64. How to Build Interceptors

65. Serialization in the Interceptor

66. Customizing the Interceptor's DTO

67. Wrapping the Interceptor in a Decorator

68. Controller-Wide Serialization

69. A Bit of Type Safety Around Serialize

Section 11: Authentication From Scratch

70. Authentication Overview

71. Reminder on Service Setup

72. Implementing Signup Functionality

73. [Optional] Understanding Password Hashing

74. Salting and Hashing the Password

75. Creating a User

76. Handling User Sign In

77. Setting up Sessions

78. Changing and Fetching Session Data

79. Signing in a User

80. Getting the Current User

81. Signing Out a User

82. Two Automation Tools

83. Custom Param Decorators

84. Why a Decorator and Interceptor

85. Communicating from Interceptor to Decorator

86. Small Fix for CurrentUserInterceptor

87. Connecting an Interceptor to Dependency Injection

88. Globally Scoped Interceptors

89. Preventing Access with Authentication Guards

Section 12: Getting ed with Unit Testing

0 / 21 | 1hr 30 of 21 lectures completed1hr 3

90. Testing Overview

91. Testing Setup

92. Yes, Testing is Confusing

93. Getting TypeScript to Help With Mocks

94. Quick Note to Help Speed Up Your Tests

95. Improving File Layout

96. Ensuring Password Gets Hashed

97. Refactor 'Email in Use' Test to Address Jest Breaking Changes

98. Changing Mock Implementations

99. Refactor 'Unused Email' Test to Address Jest Breaking Changes

100. Testing the Signin Flow

101. Refactor 'Invalid Password' Test to Address Jest Breaking Changes

102. Checking Password Comparison

103. More Intelligent Mocks

104. Refactor 3 Mocks to Address Jest Breaking Changes

105. Refactoring to Use Intelligent Mocks

106. Unit Testing a Controller

107. More Mock Implementations

108. Refactor 'User id is not found' Test to Address Jest Breaking Changes

109. Not Super Effective Tests

110. Testing the Signin Method

Section 13: Integration Testing

0 / 7 | 30 of 7 lectures completed3

111. Getting ed with End to End Testing

112. Creating an End to End Test

113. App Setup Issues in Spec Files

114. Applying a Globally Scoped Pipe

115. Applying a Globally Scoped Middleware

116. Solving Failures Around Repeat Test Runs

117. Creating Separate Test and Dev Databases

Section 14: Managing App Configuration

0 / 6 | 30min0 of 6 lectures completed30min

118. Understanding Dotenv

119. Applying Dotenv for Config

120. Specifying the Runtime Environment

121. Solving a SQLite Error

122. It Works!

123. A Followup Test

Section 15: Relations with TypeORM

0 / 14 | 1hr 0 of 14 lectures completed1hr

124. Back to Reports

125. Adding Properties to Reports

126. A DTO for Report Creation

127. Receiving Report Creation Requests

128. Saving a Report with the Reports Service

129. Testing Report Creation

130. Building Associations

131. Types of Associations

132. The ManyToOne and OneToMany Decorators

133. Important Note About Deleting Database - Do Not Skip

134. More on Decorators

135. Setting up the Association

136. Formatting the Report Response

137. Transforming Properties with a DTO

Section 16: A Basic Permissions System

0 / 12 | 50 of 12 lectures completed5

138. Adding in Report Approval

139. Required Update for changeApproval Method

140. Testing Report Approval

141. Authorization vs Authentication

142. Adding an Authorization Guard

143. The Guard Doesn't Work?!

144. Middlewares, Guards, and Interceptors

145. Assigning CurrentUser with a Middleware

146. Fixing a Type Definition Error

147. Validating Query String Values

148. Transforming Query String Data

149. How Will We Generate an Estimate

Section 17: Query Builders with TypeORM

0 / 3 | 10 of 3 lectures completed1

150. Creating a Query Builder

151. Writing a Query to Produce the Estimate

152. Testing the Estimate Logic

Section 18: Production Deployment

0 / 16 | 1hr 10 of 16 lectures completed1hr 1

153. The Path to Production

154. Providing the Cookie Key

155. Understanding the Synchronize Flag

156. The Dangers of Synchronize

157. The Theory Behind Migrations

158. Headaches with Config Management

159. TypeORM and Nest Config is Great

2

160. Env-Specific Database Config

161. Installing the TypeORM CLI

162. Generating and Running Migrations

163. Required Migration Update for Production

164. Running Migrations During E2E Tests

165. Production DB Config

166. Heroku CLI Setup Instructions

167. Heroku Specific Project Config

168. Deploying the App - (Final Lecture)