**Node.js**

* 1. Node History,Story *…………………………….finished*
  2. *2009 – node.js*
  3. *2010 – npm*
  4. *2010 – mongodb*
  5. *json*
  6. *react.js*
  7. *express.js*
  8. *then popular grouwing node*
  9. *javascript everywhere*
  10. *node on mobile*
  11. *node on desktop*
  12. Node How Work? *………………………………...finished*
  13. *runtime*
  14. *v8 engine*
  15. *threads*
  16. *server*
  17. *database*
  18. *single thread*
  19. *request/response*
  20. *assistent/worker*
  21. *no-blocking i/o*
  22. *asynchronous*
  23. *callback*
  24. *cpu intensive*
  25. *event loop*
  26. *libuv*
  27. *kernel*
  28. *multi-thread*
  29. *then blessing fast, high performance*
  30. Node Intro………………………………*finished*
  31. Node Get Started………………………*.finished*
  32. Node Modules System*…………………………..finished*
  33. *encapsulated*
  34. *wrapper function*
  35. *exports // named exportts*
  36. *module.exports // default exports*
  37. *require() // do not need .js*
  38. *\_\_dirname*
  39. *\_\_filename*
  40. *“type”: “module” // ES6 module*
  41. *import, export // must be use .js*
  42. *born npm*
  43. *three types module – external,custom,default*
  44. Node Global Object *……………………………..finished*
  45. *not have window*
  46. *global*
  47. *setTimeout()*
  48. *setInterval()*
  49. *clearTimeout()*
  50. *clearInterval()*
  51. *fetch()*
  52. *clearImmediate()*
  53. *setImmediate()*
  54. *queueMicrotask()*
  55. Node Include Modules*………………...finished*
  56. *require()*
  57. Node Create Custom Modules*…………...finished*
  58. *exports.custom = custom*
  59. Node Path Module *………………………………………...finished*
  60. *path.basename()*
  61. *path.dirname()*
  62. *path.extname()*
  63. *path.format()*
  64. *path.parse()*
  65. Node Os Module *…………………………………..finished*
  66. *os.cpus()*
  67. *os.freemem()*
  68. *os.homedir()*
  69. *os.hostname()*
  70. *os.platform()*
  71. *os.tmpdir()*
  72. *os.version()*
  73. Node HTTP Modules*………….finished*
  74. *http.createServer().listen(port,host,callback())*
  75. *http.createServer().on()*
  76. *req.url*
  77. *req.on(‘data’,callback(chunk))*
  78. *req.on(‘end’, callback())*
  79. *req.method*
  80. *req.headers*
  81. *res.writeHead()*
  82. *res.write()*
  83. *res.end()*
  84. *res.statusCode = 200;*
  85. *res.setHeader(‘Content-Type’, ‘text/plain’);*
  86. Node File System*……………….finished*
  87. *require(‘fs’)*
  88. *fs.writeFile()*
  89. *fs.appendFile()*
  90. *fs.readFile()*
  91. *fs.unlink()*
  92. *fs.rename()*
  93. Streams & Buffers
  94. *buffer data type*
  95. *readStreame*
  96. *writeStreame*
  97. *fs.createReadStreame()*
  98. *fs.createWriteStreame()*
  99. *readStreame.on(‘data’,callback(chunk))*
  100. *writeStreame.write(chunk)*
  101. *readStreame.pipe(writeStreame)*
  102. Node String Decoder
  103. *{StringDecoder} = require(‘string\_decoder’)*
  104. *new StringDecoder(‘utf-8’)*
  105. *decoder.write(buffer)*
  106. *decoder.end()*
  107. Node URL Module*…………………..finished*
  108. *require(‘url’)*
  109. *url.parse()*
  110. *url.format()*
  111. Node NPM*…………………………………..finished*
  112. *npm (Node Package Manager) was included with Node.js by default. When you installed Node.js, npm was automatically installed along with it. This allowed you to use npm to easily manage packages and dependencies for your Node.js projects.*
  113. *npm CLI*
  114. *See Node\_Package\_Manager.docx*
  115. Node Events*…………………………….finished*
  116. *require(‘events’)*
  117. *new events.EventEmmitter()*
  118. *eventEmmitter.on()*
  119. *eventEmmitter.emit()*
  120. *event bell ring module system*
  121. Node Worker Threads *………………………..finished*
  122. ***main.js***
  123. *{Worker, isMainThread} = require(‘worker\_threads’)*
  124. *new Worker(‘worker.js’,{workerData})*
  125. *worker.postMessage(‘message’)*
  126. *worker.on(‘message’,callback(message))*
  127. *worker.terminate()*
  128. ***worker.js***
  129. *{workerData,parentPort}=require(‘worker\_threads’)*
  130. *parentPort.on(‘message’,callbak(message))*
  131. *parentPort.postMessage(result)*
  132. Raw-Node-API-Project
  133. *requirement analysis*
  134. *create server*
  135. *handling request/response*
  136. *manage route using switch*
  137. Node Useful Criterias
  138. *Uses for Create Server*
  139. *Uses for request, response smooth handling*
  140. *Uses for build scalable and high performance web application*
  141. *Uses for API Development*
  142. *Uses for create CLIs*
  143. *Uses for automate repettitive tasks*
  144. *Uses for system administration*
  145. *Uses for real-time application*
  146. *Uses for IoT application*
  147. *Uses for microservice application*
  148. *Uses for streamming services*
  149. *Uses for database operation*
  150. *Uses for authentication and authorization*
  151. *Uses for file system operation*
  152. *Uses for testing and automation*
  153. *Uses for package management*
  154. *Uses for dashboard an Reporting*
  155. *Uses for develop network applicaton*
  156. *Uses for create TCP/UDP server and client*
  157. *Uses for data processing*