

# Development environment

This page will go over setting up your development environment to run Celestia software. This environment can be used for development, building binaries, and running nodes.

## Install dependencies

1. If you are on Ubuntu, first update and upgrade your OS:
2. APT
3. YUM
4. bash
5. sudo
6. apt
7. update
8. &&
9. sudo
10. apt
11. upgrade
12. -y
13. sudo
14. apt
15. update
16. &&
17. sudo
18. apt
19. upgrade
20. -y
21. bash
22. sudo
23. yum
24. update
25. sudo
26. yum
27. update
28. Install essential packages that are necessary to execute many tasks like downloading files, compiling, and monitoring the node:
29. APT
30. YUM
31. Mac
32. bash
33. sudo
34. apt
35. install
36. curl
37. tar
38. wget
39. aria2
40. clang
41. pkg-config
42. libssl-dev
43. jq
44. build-essential
45. \
46. git
47. make
48. ncdu
49. -y
50. sudo
51. apt
52. install
53. curl
54. tar
55. wget
56. aria2
57. clang
58. pkg-config
59. libssl-dev

```
60. jq
61. build-essential
62. \
63. git
64. make
65. ncd
66. -y
67. bash
68. sudo
69. yum
70. install
71. curl
72. tar
73. wget
74. aria2
75. clang
76. pkg-config
77. libssl-dev
78. jq
79. build-essential
80. \
81. git
82. make
83. ncd
84. -y
85. sudo
86. yum
87. install
88. curl
89. tar
90. wget
91. aria2
92. clang
93. pkg-config
94. libssl-dev
95. jq
96. build-essential
97. \
98. git
99. make
100. ncd
101. -y
102. bash
```

103. **these commands are for installing Homebrew, wget and jq**

104. **follow the instructions from the output after running this command**

```
105. /bin/bash
106. -c
107. "(
108. curl
109. -fsSL
110. https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

111. **then install wget & jq**

```
112. brew
113. install
114. wget
115. &&
```

116. brew  
117. install  
118. jq

119. **these commands are for installing Homebrew, wget and jq**

120. **follow the instructions from the output after running this command**

121. /bin/bash  
122. -c  
123. "(  
124. curl  
125. -fsSL  
126. https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

127. **then install wget & jq**

128. brew  
129. install  
130. wget  
131. &&  
132. brew  
133. install  
134. jq

## Install Golang

celestia-node is written in Golang so we must install Golang to build and run our node.

1. Set the version for your desired network:
2. Mainnet Beta
3. Mocha
4. Arabica
5. bash
6. ver
7. =
8. "1.21.1"
9. ver
10. =
11. "1.21.1"
12. bash
13. ver
14. =
15. "1.21.1"
16. ver
17. =
18. "1.21.1"
19. bash
20. ver
21. =
22. "1.21.1"
23. ver
24. =
25. "1.21.1"
26. Download and install Golang:
27. Ubuntu (AMD)
28. Ubuntu (ARM)
29. Mac (Apple)
30. Mac (Intel)
31. bash
32. cd

33. HOME
34. wget
35. "https://golang.org/dl/go
36. ver
37. .linux-amd64.tar.gz"
38. sudo
39. rm
40. -rf
41. /usr/local/go
42. sudo
43. tar
44. -C
45. /usr/local
46. -xzf
47. "go
48. ver
49. .linux-amd64.tar.gz"
50. rm
51. "go
52. ver
53. .linux-amd64.tar.gz"
54. cd
55. HOME
56. wget
57. "https://golang.org/dl/go
58. ver
59. .linux-amd64.tar.gz"
60. sudo
61. rm
62. -rf
63. /usr/local/go
64. sudo
65. tar
66. -C
67. /usr/local
68. -xzf
69. "go
70. ver
71. .linux-amd64.tar.gz"
72. rm
73. "go
74. ver
75. .linux-amd64.tar.gz"
76. bash
77. cd
78. HOME
79. wget
80. "https://golang.org/dl/go
81. ver
82. .linux-arm64.tar.gz"
83. sudo
84. rm
85. -rf
86. /usr/local/go
87. sudo
88. tar
89. -C
90. /usr/local
91. -xzf
92. "go
93. ver
94. .linux-arm64.tar.gz"
95. rm
96. "go
97. ver
98. .linux-arm64.tar.gz"
99. cd
100. HOME

```
101. wget
102. "https://golang.org/dl/go
103. ver
104. .linux-arm64.tar.gz"
105. sudo
106. rm
107. -rf
108. /usr/local/go
109. sudo
110. tar
111. -C
112. /usr/local
113. -xzf
114. "go
115. ver
116. .linux-arm64.tar.gz"
117. rm
118. "go
119. ver
120. .linux-arm64.tar.gz"
121. bash
122. cd
123. HOME
124. wget
125. "https://golang.org/dl/go
126. ver
127. .darwin-arm64.tar.gz"
128. sudo
129. rm
130. -rf
131. /usr/local/go
132. sudo
133. tar
134. -C
135. /usr/local
136. -xzf
137. "go
138. ver
139. .darwin-arm64.tar.gz"
140. rm
141. "go
142. ver
143. .darwin-arm64.tar.gz"
144. cd
145. HOME
146. wget
147. "https://golang.org/dl/go
148. ver
149. .darwin-arm64.tar.gz"
150. sudo
151. rm
152. -rf
153. /usr/local/go
154. sudo
155. tar
156. -C
157. /usr/local
158. -xzf
159. "go
160. ver
161. .darwin-arm64.tar.gz"
162. rm
163. "go
164. ver
165. .darwin-arm64.tar.gz"
166. bash
167. cd
168. HOME
```

```
169. wget
170. "https://golang.org/dl/go
171. ver
172. .darwin-amd64.tar.gz"
173. sudo
174. rm
175. -rf
176. /usr/local/go
177. sudo
178. tar
179. -C
180. /usr/local
181. -xzf
182. "go
183. ver
184. .darwin-amd64.tar.gz"
185. rm
186. "go
187. ver
188. .darwin-amd64.tar.gz"
189. cd
190. HOME
191. wget
192. "https://golang.org/dl/go
193. ver
194. .darwin-amd64.tar.gz"
195. sudo
196. rm
197. -rf
198. /usr/local/go
199. sudo
200. tar
201. -C
202. /usr/local
203. -xzf
204. "go
205. ver
206. .darwin-amd64.tar.gz"
207. rm
208. "go
209. ver
210. .darwin-amd64.tar.gz"
211. Add your/usr/local/go/bin
212. directory to yourPATH
213. if you have not already:
214. bash
215. zsh
216. bash
217. echo
218. "export PATH=
219. PATH
220. :/usr/local/go/bin:
221. HOME
222. /go/bin"
223.

224. HOME
225. ~/.bash_profile
226. source
227. HOME
228. ~/.bash_profile
229. echo
230. "export PATH=
231. PATH
232. :/usr/local/go/bin:
233. HOME
234. /go/bin"
235.
```

```
236. HOME
237. ~/.bash_profile
238. source
239. HOME
240. ~/.bash_profile
241. bash
242. echo
243. "export PATH=
244. PATH
245. :/usr/local/go/bin:
246. HOME
247. /go/bin"
248.
```

```
249. HOME
250. ~/.zshrc
251. source
252. HOME
253. ~/.zshrc
254. echo
255. "export PATH=
256. PATH
257. :/usr/local/go/bin:
258. HOME
259. /go/bin"
260.
```

```
261. HOME
262. ~/.zshrc
263. source
264. HOME
265. ~/.zshrc
266. To verify that the correct version of Go was installed correctly run:
267. bash
268. go
269. version
270. go
271. version
```

The output will show the version installed. [[I Edit this page on GitHub](#)] Last updated: [Previous page](#) [Deciding which node to run](#) [Next page](#) [Install celestia-node](#) []