

Windows WSL 2 설치

2022. 9.

정 준 수 PhD

PC(Local) 환경구축: WSL2

윈도우 환경에서 원하는 Linux를 실행

1. 관리자 권한으로 명령 프롬프트(CMD) 실행
2. <https://docs.microsoft.com/ko-kr/windows/wsl/tutorials/gui-apps>
3. C> Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-windows-Subsystem-Linux
4. 또는 <https://ivyit.tistory.com/264>
5. Microsoft Store 열기
6. Ubuntu 또는 **Debian** 선택

<https://docs.microsoft.com/ko-kr/windows/wsl/install>

<https://docs.microsoft.com/ko-kr/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package>

제어판 홈

설치된 업데이트 보기

Windows 기능 켜기/끄기

프로그램 제거 또는 변경

프로그램을 제거하려면 목록에서 선택한 후 [제거], [변경] 또는 [복구]를 클릭하십시오.

구성

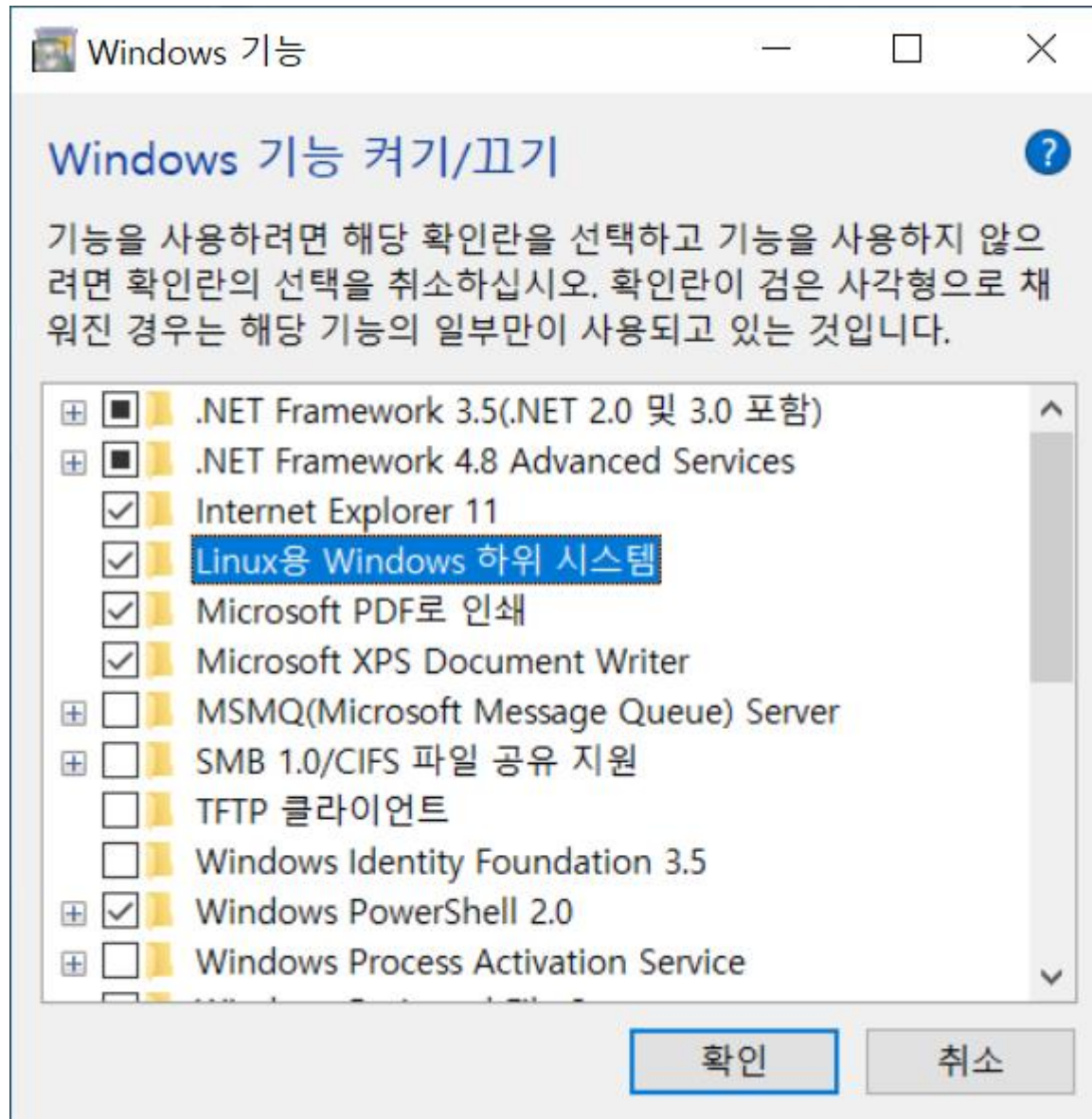
이름	게시자	설치 날짜	크기	버전
AhnLab Online Security	AhnLab, Inc	2022-02-24		
AhnLab Safe Transaction	AhnLab, Inc.	2022-02-23		1.5.1.1581
Anaconda3 2021.11 (Python 3.9.7 64-bit)	Anaconda, Inc.	2022-02-12		2021.11
Brave	Brave Software Inc	2022-07-06		103.1.40.113
Chrome	Google LLC	2022-07-05		103.0.5060.114
Cisco Webex Meetings	Cisco Webex LLC	2022-03-18	296MB	42.3.1
CREON		2022-02-24		
Delfino G3 (x86) 버전 3.6.8.4	Wizvera	2022-07-03	38.2MB	3.6.8.4
FileZilla Client 3.57.0	Tim Kosse	2022-02-01	40.9MB	3.57.0
iniLINE CrossEX Service	iniLINE Co., Ltd.	2022-07-03		1.0.2.9
INISAFE CrossWeb EX V3	Initech, Inc.	2022-03-08		3.3.2.26
IPinside Agent	interezen	2022-02-24		1.0.2.8
Java(TM) SE Development Kit 18 (64-bit)	Oracle Corporation	2022-03-27	292MB	18.0.0.0
Microsoft Edge	Microsoft Corporation	2022-07-08		103.0.1264.49
Microsoft Edge WebView2 런타임	Microsoft Corporation	2022-07-09		103.0.1264.49
Microsoft Teams	Microsoft Corporation	2022-02-28	118MB	1.5.00.4689



현재 설치된 프로그램

전체 크기: 7.04GB

41개의 프로그램이 설치되었습니다.



Google

docker - Google Sheets

Install Docker Desktop on Wind

+

docs.docker.com/desktop/windows/install/

🔖 ☆ ⚙️ 🗖️ 👤 ⋮

docker docs

🔍 Search the docs

Home

Guides

Manuals

Reference

Samples

📄

🏠 / Manuals / Docker Desktop / Windows / Install Docker Desktop for Windows

Docker Desktop

Overview

Mac

Windows

Install Docker Desktop for Windows

User manual

Logs and troubleshooting

Docker Desktop WSL 2 backend

Linux

Dashboard

Explore networking features

Dev Environments (Beta)

Extensions (Beta)

Extensions SDK (Beta)

Multi-arch support

Deploy on Kubernetes

Back up and restore data

FAQs

Get support

Give feedback

Release notes

Previous versions

Docker Engine

Docker Compose

Docker Hub

Docker subscription

Administration

📄

📄 Edit this page

✓ Request docs changes

⚙️ ☒ 🌙

On this page:

System requirements

WSL 2 backend

Hyper-V backend and Windows containers

About Windows containers

Install Docker Desktop on Windows

Install interactively

Install from the command line

Start Docker Desktop

Quick Start Guide

Updates

Uninstall Docker Desktop

Where to go next

📄

Install Docker Desktop on Windows

Estimated reading time: 10 minutes

📄 Update to the Docker Desktop terms

Commercial use of Docker Desktop in larger enterprises (more than 250 employees OR more than \$10 million USD in annual revenue) now requires a paid subscription.

Welcome to Docker Desktop for Windows. This page contains information about Docker Desktop for Windows system requirements, download URL, instructions to install and update Docker Desktop for Windows.

📄 Download Docker Desktop for Windows

Docker Desktop for Windows

For checksums, see [Release notes](#)

System requirements

Your Windows machine must meet the following requirements to successfully install Docker Desktop.

WSL 2 backend

Hyper-V backend and Windows containers

WSL 2 backend

- Windows 11 64-bit: Home or Pro version 21H2 or higher, or Enterprise or Education version 21H2 or higher.
- Windows 10 64-bit: Home or Pro 21H1 (build 19043) or higher, or Enterprise or Education 20H2 (build 19042) or higher.
- Enable the WSL 2 feature on Windows. For detailed instructions, refer to the [Microsoft documentation](#).
- The following hardware prerequisites are required to successfully run WSL 2 on Windows 10 or Windows 11:

Docker Desktop 1...exe

331/496MB, 9초 남음

모두 표시

✕

6



Installing Docker Desktop 4.10.1 (82475)



Configuration

☒ Add shortcut to desktop

Ok



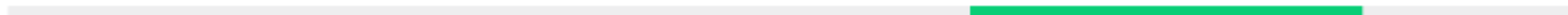
Installing Docker Desktop 4.10.1 (82475)



Docker Desktop 4.10.1

Unpacking files...

Unpacking file: resources/docker-desktop.iso
Unpacking file: resources/ddvp.ico
Unpacking file: resources/config-options.json
Unpacking file: resources/componentsVersion.json
Unpacking file: resources/bin/docker-compose
Unpacking file: resources/bin/docker
Unpacking file: resources/.gitignore
Unpacking file: InstallerCli.pdb
Unpacking file: InstallerCli.exe.config
Unpacking file: frontend/vk_swiftshader_icd.json
Unpacking file: frontend/v8_context_snapshot.bin
Unpacking file: frontend/snapshot_blob.bin
Unpacking file: frontend/resources/regedit/vbs/util.vbs
Unpacking file: frontend/resources/regedit/vbs/regUtil.vbs





Installing Docker Desktop 4.10.1 (82475)



Docker Desktop 4.10.1

Installation succeeded

Close





Containers



Images



Volumes

Dev Environments **BETA**

Extensions

BETA

Add Extensions



Docker Desktop - Install WSL 2 kernel update

**WSL 2 installation is incomplete.**

The WSL 2 Linux kernel is now installed using a separate MSI update package. Please click the link and follow the instructions to install the kernel update: <https://aka.ms/wsl2kernel>.

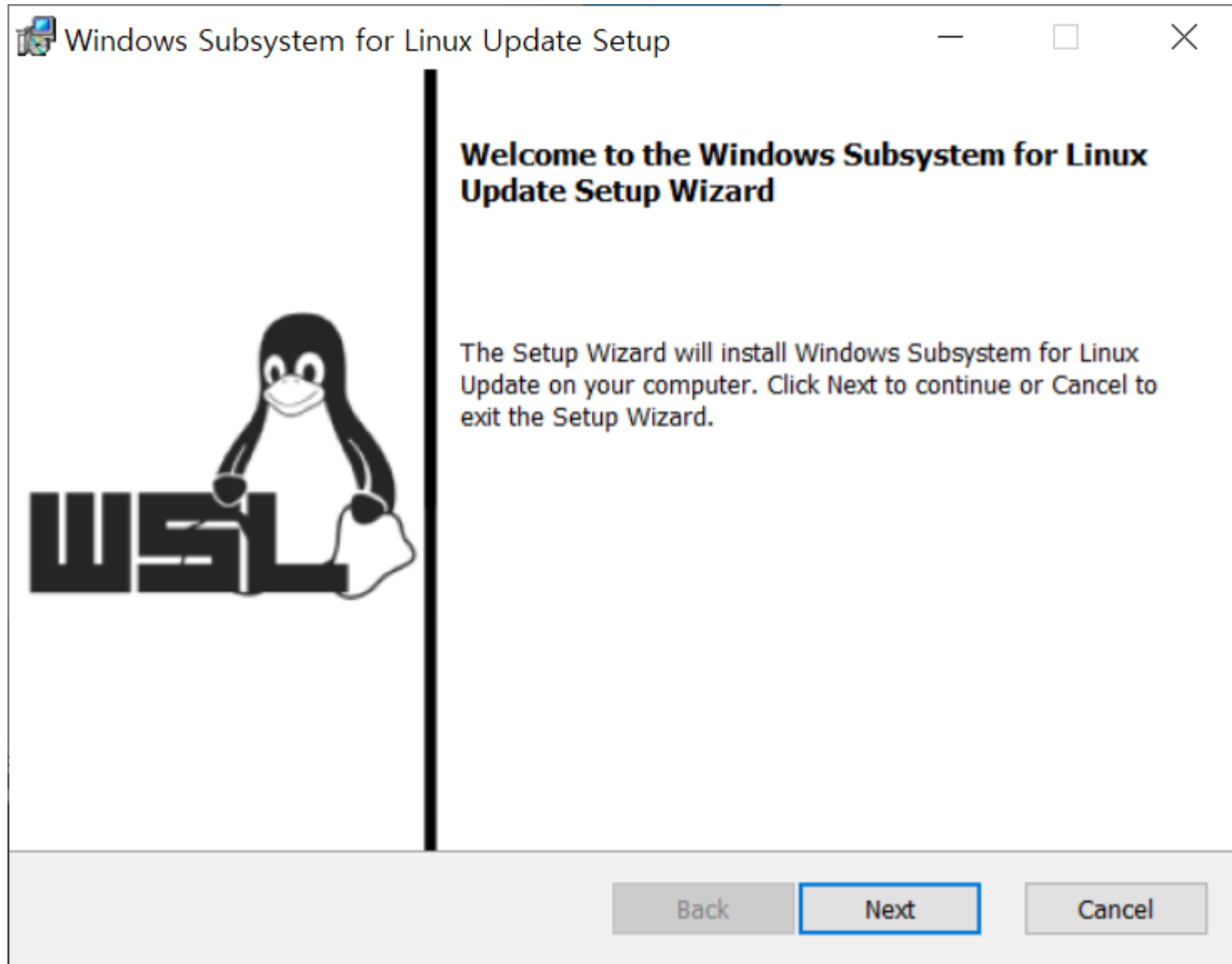
Press Restart after installing the Linux kernel.

Restart

Cancel

WSL 2 installation incomplete error 발생시 설치 Package

<https://docs.microsoft.com/ko-kr/windows/wsl/install-manual#step-4--download-the-linux-kernel-update-package>



C:\> 관리자: 명령 프롬프트

Microsoft Windows [Version 10.0.19044.1766]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>wsl --set-default-version 2_

C:\ 관리자: 명령 프롬프트

Microsoft Windows [Version 10.0.19044.1766]

(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>wsl --set-default-version 2

WSL 2와의 주요 차이점에 대한 자세한 내용은 <https://aka.ms/wsl2>를 참조하세요
작업을 완료했습니다.

C:\Windows\system32>_

설치 가능한 리눅스 배포본은 "-l -o" 옵션으로 확인

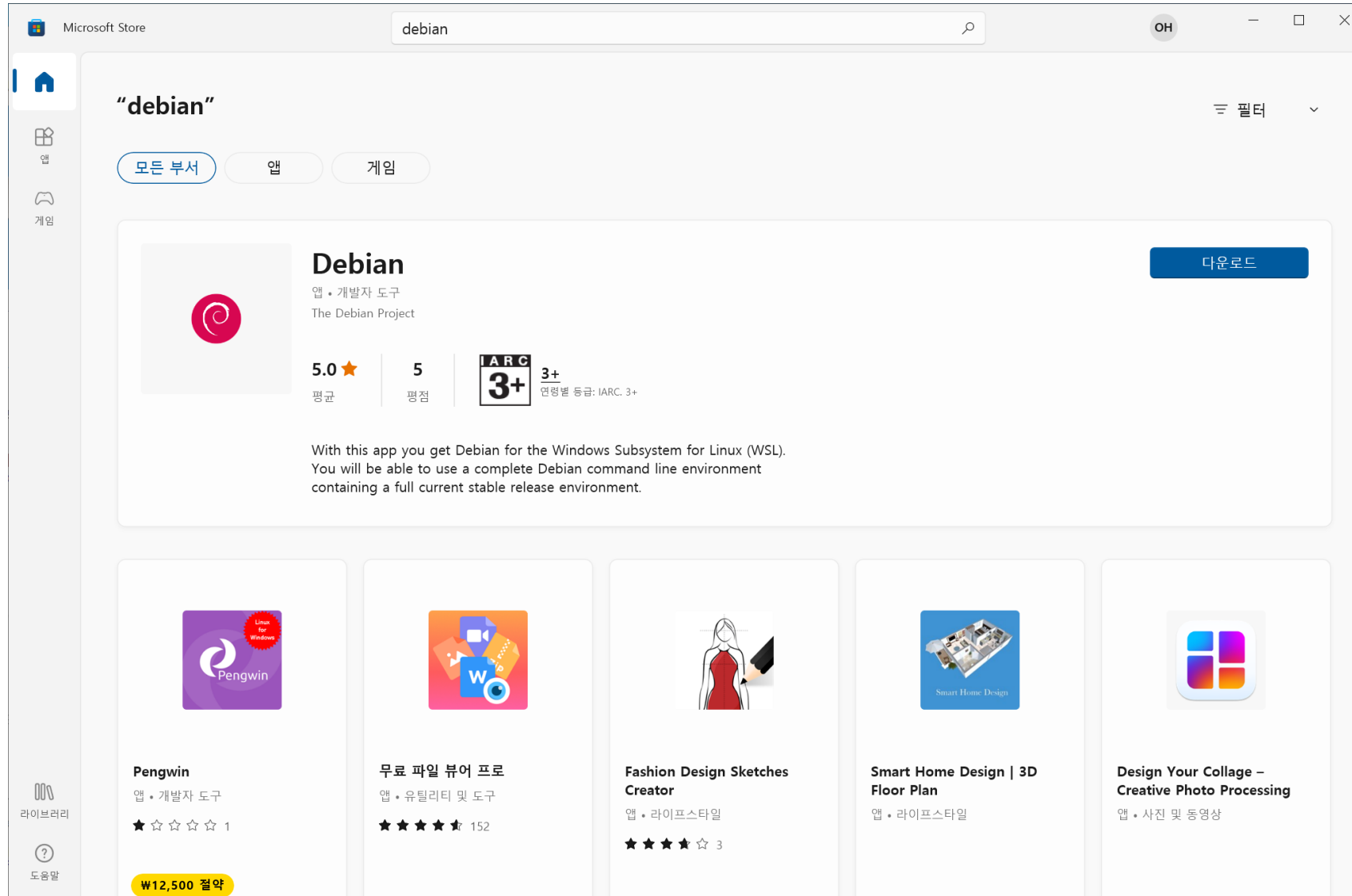
```
관리자: 명령 프롬프트
Microsoft Windows [Version 10.0.19044.1766]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>wsl -l -o
다음은 설치할 수 있는 유효한 배포 목록입니다.
'wsl --install -d <배포>'를 사용하여 설치하세요.

NAME                FRIENDLY NAME
Ubuntu              Ubuntu
Debian              Debian GNU/Linux
kali-linux          Kali Linux Rolling
openSUSE-42         openSUSE Leap 42
SLES-12             SUSE Linux Enterprise Server v12
Ubuntu-16.04        Ubuntu 16.04 LTS
Ubuntu-18.04        Ubuntu 18.04 LTS
Ubuntu-20.04        Ubuntu 20.04 LTS

C:\Windows\system32>
```

Microsoft Store에서: Linux 선택



←

ubuntu

OH

—

□

×

🏠

📁

🎮

앱

게임

20.04 LTS

Ubuntu 20.04.4 LTS

Canonical Group Limited

4.6 ⭐

10

3+

80.53MB/494.6MB 다운로드...

평균

평점

연령별 등급: IARC. 3+

Install a complete Ubuntu terminal environment in minutes with Windows Subsystem for Linux (WSL)....

개발자 도구

🔗

스크린샷

➤

WSL - Ubuntu

```
1000 # base Arch Linux 22.01
1001 #
1002 # If not listed
1003 # list.append(['', ''])
1004 # take care of the done
1005 #
1006 # [cmd, kind] = list[0][1]
1007 # kind[0] == 'f' and not
1008 # list.index(1)
1009 # list[0][1] = list[1]
1010 #
1011 # [cmd, kind] = list[1][1]
1012 # kind[1] == 'f' and not
1013 # list.index(1)
1014 # list[1][1] = list[2]
1015 #
1016 # [cmd, kind] = list[2][1]
1017 # kind[2] == 'f' and not
1018 # list.index(1)
1019 # list[2][1] = list[3]
1020 #
1021 # [cmd, kind] = list[3][1]
1022 # kind[3] == 'f' and not
1023 # list.index(1)
1024 # list[3][1] = list[4]
1025 #
1026 # [cmd, kind] = list[4][1]
1027 # kind[4] == 'f' and not
1028 # list.index(1)
1029 # list[4][1] = list[5]
1030 #
1031 # [cmd, kind] = list[5][1]
1032 # kind[5] == 'f' and not
1033 # list.index(1)
1034 # list[5][1] = list[6]
1035 #
1036 # [cmd, kind] = list[6][1]
1037 # kind[6] == 'f' and not
1038 # list.index(1)
1039 # list[6][1] = list[7]
1040 #
1041 # [cmd, kind] = list[7][1]
1042 # kind[7] == 'f' and not
1043 # list.index(1)
1044 # list[7][1] = list[8]
1045 #
1046 # [cmd, kind] = list[8][1]
1047 # kind[8] == 'f' and not
1048 # list.index(1)
1049 # list[8][1] = list[9]
1050 #
1051 # [cmd, kind] = list[9][1]
1052 # kind[9] == 'f' and not
1053 # list.index(1)
1054 # list[9][1] = list[10]
1055 #
1056 # [cmd, kind] = list[10][1]
1057 # kind[10] == 'f' and not
1058 # list.index(1)
1059 # list[10][1] = list[11]
1060 #
1061 # [cmd, kind] = list[11][1]
1062 # kind[11] == 'f' and not
1063 # list.index(1)
1064 # list[11][1] = list[12]
1065 #
1066 # [cmd, kind] = list[12][1]
1067 # kind[12] == 'f' and not
1068 # list.index(1)
1069 # list[12][1] = list[13]
1070 #
1071 # [cmd, kind] = list[13][1]
1072 # kind[13] == 'f' and not
1073 # list.index(1)
1074 # list[13][1] = list[14]
1075 #
1076 # [cmd, kind] = list[14][1]
1077 # kind[14] == 'f' and not
1078 # list.index(1)
1079 # list[14][1] = list[15]
1080 #
1081 # [cmd, kind] = list[15][1]
1082 # kind[15] == 'f' and not
1083 # list.index(1)
1084 # list[15][1] = list[16]
1085 #
1086 # [cmd, kind] = list[16][1]
1087 # kind[16] == 'f' and not
1088 # list.index(1)
1089 # list[16][1] = list[17]
1090 #
1091 # [cmd, kind] = list[17][1]
1092 # kind[17] == 'f' and not
1093 # list.index(1)
1094 # list[17][1] = list[18]
1095 #
1096 # [cmd, kind] = list[18][1]
1097 # kind[18] == 'f' and not
1098 # list.index(1)
1099 # list[18][1] = list[19]
1100 #
1101 # [cmd, kind] = list[19][1]
1102 # kind[19] == 'f' and not
1103 # list.index(1)
1104 # list[19][1] = list[20]
1105 #
1106 # [cmd, kind] = list[20][1]
1107 # kind[20] == 'f' and not
1108 # list.index(1)
1109 # list[20][1] = list[21]
1110 #
1111 # [cmd, kind] = list[21][1]
1112 # kind[21] == 'f' and not
1113 # list.index(1)
1114 # list[21][1] = list[22]
1115 #
1116 # [cmd, kind] = list[22][1]
1117 # kind[22] == 'f' and not
1118 # list.index(1)
1119 # list[22][1] = list[23]
1120 #
1121 # [cmd, kind] = list[23][1]
1122 # kind[23] == 'f' and not
1123 # list.index(1)
1124 # list[23][1] = list[24]
1125 #
1126 # [cmd, kind] = list[24][1]
1127 # kind[24] == 'f' and not
1128 # list.index(1)
1129 # list[24][1] = list[25]
1130 #
1131 # [cmd, kind] = list[25][1]
1132 # kind[25] == 'f' and not
1133 # list.index(1)
1134 # list[25][1] = list[26]
1135 #
1136 # [cmd, kind] = list[26][1]
1137 # kind[26] == 'f' and not
1138 # list.index(1)
1139 # list[26][1] = list[27]
1140 #
1141 # [cmd, kind] = list[27][1]
1142 # kind[27] == 'f' and not
1143 # list.index(1)
1144 # list[27][1] = list[28]
1145 #
1146 # [cmd, kind] = list[28][1]
1147 # kind[28] == 'f' and not
1148 # list.index(1)
1149 # list[28][1] = list[29]
1150 #
1151 # [cmd, kind] = list[29][1]
1152 # kind[29] == 'f' and not
1153 # list.index(1)
1154 # list[29][1] = list[30]
1155 #
1156 # [cmd, kind] = list[30][1]
1157 # kind[30] == 'f' and not
1158 # list.index(1)
1159 # list[30][1] = list[31]
1160 #
1161 # [cmd, kind] = list[31][1]
1162 # kind[31] == 'f' and not
1163 # list.index(1)
1164 # list[31][1] = list[32]
1165 #
1166 # [cmd, kind] = list[32][1]
1167 # kind[32] == 'f' and not
1168 # list.index(1)
1169 # list[32][1] = list[33]
1170 #
1171 # [cmd, kind] = list[33][1]
1172 # kind[33] == 'f' and not
1173 # list.index(1)
1174 # list[33][1] = list[34]
1175 #
1176 # [cmd, kind] = list[34][1]
1177 # kind[34] == 'f' and not
1178 # list.index(1)
1179 # list[34][1] = list[35]
1180 #
1181 # [cmd, kind] = list[35][1]
1182 # kind[35] == 'f' and not
1183 # list.index(1)
1184 # list[35][1] = list[36]
1185 #
1186 # [cmd, kind] = list[36][1]
1187 # kind[36] == 'f' and not
1188 # list.index(1)
1189 # list[36][1] = list[37]
1190 #
1191 # [cmd, kind] = list[37][1]
1192 # kind[37] == 'f' and not
1193 # list.index(1)
1194 # list[37][1] = list[38]
1195 #
1196 # [cmd, kind] = list[38][1]
1197 # kind[38] == 'f' and not
1198 # list.index(1)
1199 # list[38][1] = list[39]
1200 #
1201 # [cmd, kind] = list[39][1]
1202 # kind[39] == 'f' and not
1203 # list.index(1)
1204 # list[39][1] = list[40]
1205 #
1206 # [cmd, kind] = list[40][1]
1207 # kind[40] == 'f' and not
1208 # list.index(1)
1209 # list[40][1] = list[41]
1210 #
1211 # [cmd, kind] = list[41][1]
1212 # kind[41] == 'f' and not
1213 # list.index(1)
1214 # list[41][1] = list[42]
1215 #
1216 # [cmd, kind] = list[42][1]
1217 # kind[42] == 'f' and not
1218 # list.index(1)
1219 # list[42][1] = list[43]
1220 #
1221 # [cmd, kind] = list[43][1]
1222 # kind[43] == 'f' and not
1223 # list.index(1)
1224 # list[43][1] = list[44]
1225 #
1226 # [cmd, kind] = list[44][1]
1227 # kind[44] == 'f' and not
1228 # list.index(1)
1229 # list[44][1] = list[45]
1230 #
1231 # [cmd, kind] = list[45][1]
1232 # kind[45] == 'f' and not
1233 # list.index(1)
1234 # list[45][1] = list[46]
1235 #
1236 # [cmd, kind] = list[46][1]
1237 # kind[46] == 'f' and not
1238 # list.index(1)
1239 # list[46][1] = list[47]
1240 #
1241 # [cmd, kind] = list[47][1]
1242 # kind[47] == 'f' and not
1243 # list.index(1)
1244 # list[47][1] = list[48]
1245 #
1246 # [cmd, kind] = list[48][1]
1247 # kind[48] == 'f' and not
1248 # list.index(1)
1249 # list[48][1] = list[49]
1250 #
1251 # [cmd, kind] = list[49][1]
1252 # kind[49] == 'f' and not
1253 # list.index(1)
1254 # list[49][1] = list[50]
1255 #
1256 # [cmd, kind] = list[50][1]
1257 # kind[50] == 'f' and not
1258 # list.index(1)
1259 # list[50][1] = list[51]
1260 #
1261 # [cmd, kind] = list[51][1]
1262 # kind[51] == 'f' and not
1263 # list.index(1)
1264 # list[51][1] = list[52]
1265 #
1266 # [cmd, kind] = list[52][1]
1267 # kind[52] == 'f' and not
1268 # list.index(1)
1269 # list[52][1] = list[53]
1270 #
1271 # [cmd, kind] = list[53][1]
1272 # kind[53] == 'f' and not
1273 # list.index(1)
1274 # list[53][1] = list[54]
1275 #
1276 # [cmd, kind] = list[54][1]
1277 # kind[54] == 'f' and not
1278 # list.index(1)
1279 # list[54][1] = list[55]
1280 #
1281 # [cmd, kind] = list[55][1]
1282 # kind[55] == 'f' and not
1283 # list.index(1)
1284 # list[55][1] = list[56]
1285 #
1286 # [cmd, kind] = list[56][1]
1287 # kind[56] == 'f' and not
1288 # list.index(1)
1289 # list[56][1] = list[57]
1290 #
1291 # [cmd, kind] = list[57][1]
1292 # kind[57] == 'f' and not
1293 # list.index(1)
1294 # list[57][1] = list[58]
1295 #
1296 # [cmd, kind] = list[58][1]
1297 # kind[58] == 'f' and not
1298 # list.index(1)
1299 # list[58][1] = list[59]
1300 #
1301 # [cmd, kind] = list[59][1]
1302 # kind[59] == 'f' and not
1303 # list.index(1)
1304 # list[59][1] = list[60]
1305 #
1306 # [cmd, kind] = list[60][1]
1307 # kind[60] == 'f' and not
1308 # list.index(1)
1309 # list[60][1] = list[61]
1310 #
1311 # [cmd, kind] = list[61][1]
1312 # kind[61] == 'f' and not
1313 # list.index(1)
1314 # list[61][1] = list[62]
1315 #
1316 # [cmd, kind] = list[62][1]
1317 # kind[62] == 'f' and not
1318 # list.index(1)
1319 # list[62][1] = list[63]
1320 #
1321 # [cmd, kind] = list[63][1]
1322 # kind[63] == 'f' and not
1323 # list.index(1)
1324 # list[63][1] = list[64]
1325 #
1326 # [cmd, kind] = list[64][1]
1327 # kind[64] == 'f' and not
1328 # list.index(1)
1329 # list[64][1] = list[65]
1330 #
1331 # [cmd, kind] = list[65][1]
1332 # kind[65] == 'f' and not
1333 # list.index(1)
1334 # list[65][1] = list[66]
1335 #
1336 # [cmd, kind] = list[66][1]
1337 # kind[66] == 'f' and not
1338 # list.index(1)
1339 # list[66][1] = list[67]
1340 #
1341 # [cmd, kind] = list[67][1]
1342 # kind[67] == 'f' and not
1343 # list.index(1)
1344 # list[67][1] = list[68]
1345 #
1346 # [cmd, kind] = list[68][1]
1347 # kind[68] == 'f' and not
1348 # list.index(1)
1349 # list[68][1] = list[69]
1350 #
1351 # [cmd, kind] = list[69][1]
1352 # kind[69] == 'f' and not
1353 # list.index(1)
1354 # list[69][1] = list[70]
1355 #
1356 # [cmd, kind] = list[70][1]
1357 # kind[70] == 'f' and not
1358 # list.index(1)
1359 # list[70][1] = list[71]
1360 #
1361 # [cmd, kind] = list[71][1]
1362 # kind[71] == 'f' and not
1363 # list.index(1)
1364 # list[71][1] = list[72]
1365 #
1366 # [cmd, kind] = list[72][1]
1367 # kind[72] == 'f' and not
1368 # list.index(1)
1369 # list[72][1] = list[73]
1370 #
1371 # [cmd, kind] = list[73][1]
1372 # kind[73] == 'f' and not
1373 # list.index(1)
1374 # list[73][1] = list[74]
1375 #
1376 # [cmd, kind] = list[74][1]
1377 # kind[74] == 'f' and not
1378 # list.index(1)
1379 # list[74][1] = list[75]
1380 #
1381 # [cmd, kind] = list[75][1]
1382 # kind[75] == 'f' and not
1383 # list.index(1)
1384 # list[75][1] = list[76]
1385 #
1386 # [cmd, kind] = list[76][1]
1387 # kind[76] == 'f' and not
1388 # list.index(1)
1389 # list[76][1] = list[77]
1390 #
1391 # [cmd, kind] = list[77][1]
1392 # kind[77] == 'f' and not
1393 # list.index(1)
1394 # list[77][1] = list[78]
1395 #
1396 # [cmd, kind] = list[78][1]
1397 # kind[78] == 'f' and not
1398 # list.index(1)
1399 # list[78][1] = list[79]
1400 #
1401 # [cmd, kind] = list[79][1]
1402 # kind[79] == 'f' and not
1403 # list.index(1)
1404 # list[79][1] = list[80]
1405 #
1406 # [cmd, kind] = list[80][1]
1407 # kind[80] == 'f' and not
1408 # list.index(1)
1409 # list[80][1] = list[81]
1410 #
1411 # [cmd, kind] = list[81][1]
1412 # kind[81] == 'f' and not
1413 # list.index(1)
1414 # list[81][1] = list[82]
1415 #
1416 # [cmd, kind] = list[82][1]
1417 # kind[82] == 'f' and not
1418 # list.index(1)
1419 # list[82][1] = list[83]
1420 #
1421 # [cmd, kind] = list[83][1]
1422 # kind[83] == 'f' and not
1423 # list.index(1)
1424 # list[83][1] = list[84]
1425 #
1426 # [cmd, kind] = list[84][1]
1427 # kind[84] == 'f' and not
1428 # list.index(1)
1429 # list[84][1] = list[85]
1430 #
1431 # [cmd, kind] = list[85][1]
1432 # kind[85] == 'f' and not
1433 # list.index(1)
1434 # list[85][1] = list[86]
1435 #
1436 # [cmd, kind] = list[86][1]
1437 # kind[86] == 'f' and not
1438 # list.index(1)
1439 # list[86][1] = list[87]
1440 #
1441 # [cmd, kind] = list[87][1]
1442 # kind[87] == 'f' and not
1443 # list.index(1)
1444 # list[87][1] = list[88]
1445 #
1446 # [cmd, kind] = list[88][1]
1447 # kind[88] == 'f' and not
1448 # list.index(1)
1449 # list[88][1] = list[89]
1450 #
1451 # [cmd, kind] = list[89][1]
1452 # kind[89] == 'f' and not
1453 # list.index(1)
1454 # list[89][1] = list[90]
1455 #
1456 # [cmd, kind] = list[90][1]
1457 # kind[90] == 'f' and not
1458 # list.index(1)
1459 # list[90][1] = list[91]
1460 #
1461 # [cmd, kind] = list[91][1]
1462 # kind[91] == 'f' and not
1463 # list.index(1)
1464 # list[91][1] = list[92]
1465 #
1466 # [cmd, kind] = list[92][1]
1467 # kind[92] == 'f' and not
1468 # list.index(1)
1469 # list[92][1] = list[93]
1470 #
1471 # [cmd, kind] = list[93][1]
1472 # kind[93] == 'f' and not
1473 # list.index(1)
1474 # list[93][1] = list[94]
1475 #
1476 # [cmd, kind] = list[94][1]
1477 # kind[94] == 'f' and not
1478 # list.index(1)
1479 # list[94][1] = list[95]
1480 #
1481 # [cmd, kind] = list[95][1]
1482 # kind[95] == 'f' and not
1483 # list.index(1)
1484 # list[95][1] = list[96]
1485 #
1486 # [cmd, kind] = list[96][1]
1487 # kind[96] == 'f' and not
1488 # list.index(1)
1489 # list[96][1] = list[97]
1490 #
1491 # [cmd, kind] = list[97][1]
1492 # kind[97] == 'f' and not
1493 # list.index(1)
1494 # list[97][1] = list[98]
1495 #
1496 # [cmd, kind] = list[98][1]
1497 # kind[98] == 'f' and not
1498 # list.index(1)
1499 # list[98][1] = list[99]
1500 #
1501 # [cmd, kind] = list[99][1]
1502 # kind[99] == 'f' and not
1503 # list.index(1)
1504 # list[99][1] = list[100]
1505 #
1506 # [cmd, kind] = list[100][1]
1507 # kind[100] == 'f' and not
1508 # list.index(1)
1509 # list[100][1] = list[101]
1510 #
1511 # [cmd, kind] = list[101][1]
1512 # kind[101] == 'f' and not
1513 # list.index(1)
1514 # list[101][1] = list[102]
1515 #
1516 # [cmd, kind] = list[102][1]
1517 # kind[102] == 'f' and not
1518 # list.index(1)
1519 # list[102][1] = list[103]
1520 #
1521 # [cmd, kind] = list[103][1]
1522 # kind[103] == 'f' and not
1523 # list.index(1)
1524 # list[103][1] = list[104]
1525 #
1526 # [cmd, kind] = list[104][1]
1527 # kind[104] == 'f' and not
1528 # list.index(1)
1529 # list[104][1] = list[105]
1530 #
1531 # [cmd, kind] = list[105][1]
1532 # kind[105] == 'f' and not
1533 # list.index(1)
1534 # list[105][1] = list[106]
1535 #
1536 # [cmd, kind] = list[106][1]
1537 # kind[106] == 'f' and not
1538 # list.index(1)
1539 # list[106][1] = list[107]
1540 #
1541 # [cmd, kind] = list[107][1]
1542 # kind[107] == 'f' and not
1543 # list.index(1)
1544 # list[107][1] = list[108]
1545 #
1546 # [cmd, kind] = list[108][1]
1547 # kind[108] == 'f' and not
1548 # list.index(1)
1549 # list[108][1] = list[109]
1550 #
1551 # [cmd, kind] = list[109][1]
1552 # kind[109] == 'f' and not
1553 # list.index(1)
1554 # list[109][1] = list[110]
1555 #
1556 # [cmd, kind] = list[110][1]
1557 # kind[110] == 'f' and not
1558 # list.index(1)
1559 # list[110][1] = list[111]
1560 #
1561 # [cmd, kind] = list[111][1]
1562 # kind[111] == 'f' and not
1563 # list.index(1)
1564 # list[111][1] = list[112]
1565 #
1566 # [cmd, kind] = list[112][1]
1567 # kind[112] == 'f' and not
1568 # list.index(1)
1569 # list[112][1] = list[113]
1570 #
1571 # [cmd, kind] = list[113][1]
1572 # kind[113] == 'f' and not
1573 # list.index(1)
1574 # list[113][1] = list[114]
1575 #
1576 # [cmd, kind] = list[114][1]
1577 # kind[114] == 'f' and not
1578 # list.index(1)
1579 # list[114][1] = list[115]
1580 #
1581 # [cmd, kind] = list[115][1]
1582 # kind[115] == 'f' and not
1583 # list.index(1)
1584 # list[115][1] = list[116]
1585 #
1586 # [cmd, kind] = list[116][1]
1587 # kind[116] == 'f' and not
1588 # list.index(1)
1589 # list[116][1] = list[117]
1590 #
1591 # [cmd, kind] = list[117][1]
1592 # kind[117] == 'f' and not
1593 # list.index(1)
1594 # list[117][1] = list[118]
1595 #
1596 # [cmd, kind] = list[118][1]
1597 # kind[118] == 'f' and not
1598 # list.index(1)
1599 # list[118][1] = list[119]
1600 #
1601 # [cmd, kind] = list[119][1]
1602 # kind[119] == 'f' and not
1603 # list.index(1)
1604 # list[119][1] = list[120]
1605 #
1606 # [cmd, kind] = list[120][1]
1607 # kind[120] == 'f' and not
1608 # list.index(1)
1609 # list[120][1] = list[121]
1610 #
1611 # [cmd, kind] = list[121][1]
1612 # kind[121] == 'f' and not
1613 # list.index(1)
1614 # list[121][1] = list[122]
1615 #
1616 # [cmd, kind] = list[122][1]
1617 # kind[122] == 'f' and not
1618 # list.index(1)
1619 # list[122][1] = list[123]
1620 #
1621 # [cmd, kind] = list[123][1]
1622 # kind[123] == 'f' and not
1623 # list.index(1)
1624 # list[123][1] = list[124]
1625 #
1626 # [cmd, kind] = list[124][1]
1627 # kind[124] == 'f' and not
1628 # list.index(1)
1629 # list[124][1] = list[125]
1630 #
1631 # [cmd, kind] = list[125][1]
1632 # kind[125] == 'f' and not
1633 # list.index(1)
1634 # list[125][1] = list[126]
1635 #
1636 # [cmd, kind] = list[126][1]
1637 # kind[126] == 'f' and not
1638 # list.index(1)
1639 # list[126][1] = list[127]
1640 #
1641 # [cmd, kind] = list[127][1]
1642 # kind[127] == 'f' and not
1643 # list.index(1)
1644 # list[127][1] = list[128]
1645 #
1646 # [cmd, kind] = list[128][1]
1647 # kind[128] == 'f' and not
1648 # list.index(1)
1649 # list[128][1] = list[129]
1650 #
1651 # [cmd, kind] = list[129][1]
1652 # kind[129] == 'f' and not
1653 # list.index(1)
1654 # list[129][1] = list[130]
1655 #
1656 # [cmd, kind] = list[130][1]
1657 # kind[130] == 'f' and not
1658 # list.index(1)
1659 # list[130][1] = list[131]
1660 #
1661 # [cmd, kind] = list[131][1]
1662 # kind[131] == 'f' and not
1663 # list.index(1)
1664 # list[131][1] = list[132]
1665 #
1666 # [cmd, kind] = list[132][1]
1667 # kind[132] == 'f' and not
1668 # list.index(1)
1669 # list[132][1] = list[133]
1670 #
1671 # [cmd, kind] = list[133][1]
1672 # kind[133] == 'f' and not
1673 # list.index(1)
1674 # list[133][1] = list[134]
1675 #
1676 # [cmd, kind] = list[134][1]
1677 # kind[134] == 'f' and not
1678 # list.index(1)
1679 # list[134][1] = list[135]
1680 #
1681 # [cmd, kind] = list[135][1]
1682 # kind[135] == 'f' and not
1683 # list.index(1)
1684 # list[135][1] = list[136]
1685 #
1686 # [cmd, kind] = list[136][1]
1687 # kind[136] == 'f' and not
1688 # list.index(1)
1689 # list[136][1] = list[137]
1690 #
1691 # [cmd, kind] = list[137][1]
1692 # kind[137] == 'f' and not
1693 # list.index(1)
1694 # list[137][1] = list[138]
1695 #
1696 # [cmd, kind] = list[138][1]
1697 # kind[138] == 'f' and not
1698 # list.index(1)
1699 # list[138][1] = list[139]
1700 #
1701 # [cmd, kind] = list[139][1]
1702 # kind[139] == 'f' and not
1703 # list.index(1)
1704 # list[139][1] = list[140]
1705 #
1706 # [cmd, kind] = list[140][1]
1707 # kind[140] == 'f' and not
1708 # list.index(1)
1709 # list[140][1] = list[141]
1710 #
1711 # [cmd, kind] = list[141][1]
1712 # kind[141] == 'f' and not
1713 # list.index(1)
1714 # list[141][1] = list[142]
1715 #
1716 # [cmd, kind] = list[142][1]
1717 # kind[142] == 'f' and not
1718 # list.index(1)
1719 # list[142][1] = list[143]
1720 #
1721 # [cmd, kind] = list[143][1]
1722 # kind[143] == 'f' and not
1723 # list.index(1)
1724 # list[143][1] = list[144]
1725 #
1726 # [cmd, kind] = list[144][1]
1727 # kind[144] == 'f' and not
1728 # list.index(1)
1729 # list[144][1] = list[145]
1730 #
1731 # [cmd, kind] = list[145][1]
1732 # kind[145] == 'f' and not
1733 # list.index(1)
1734 # list[145][1] = list[146]
1735 #
1736 # [cmd, kind] = list[146][1]
1737 # kind[146] == 'f' and not
1738 # list.index(1)
1739 # list[146][1] = list[147]
1740 #
1741 # [cmd, kind] = list[147][1]
1742 # kind[147] == 'f' and not
1743 # list.index(1)
1744 # list[147][1] = list[148]
1745 #
1746 # [cmd, kind] = list[148][1]
1747 # kind[148] == 'f' and not
1748 # list.index(1)
1749 # list[148][1] = list[149]
1750 #
1751 # [cmd, kind] = list[149][1]
1752 # kind[149] == 'f' and not
1753 # list.index(1)
1754 # list[149][1] = list[150]
1755 #
1756 # [cmd, kind] = list[150][1]
1757 # kind[150] == 'f' and not
1758 # list.index(1)
1759 # list[150][1] = list[151]
1760 #
1761 # [cmd, kind] = list[151][1]
1762 # kind[151] == 'f' and not
1763 # list.index(1)
1764 # list[151][1] = list[152]
1765 #
1766 # [cmd, kind] = list[152][1]
1767 # kind[152] == 'f' and not
1768 # list.index(1)
1769 # list[152][1] = list[153]
1770 #
1771 # [cmd, kind] = list[153][1]
1772 # kind[153] == 'f' and not
1773 # list.index(1)
1774 # list[153][1] = list[154]
1775 #
1776 # [cmd, kind] = list[154][1]
1777 # kind[154] == 'f' and not
1778 # list.index(1)
1779 # list[154][1] = list[155]
1780 #
1781 # [cmd, kind] = list[155][1]
1782 # kind[155] == 'f' and not
1783 # list.index(1)
1784 # list[155][1] = list[156]
1785 #
1786 # [cmd, kind] = list[156][1]
1787 # kind[156] == 'f' and not
1788 # list.index(1)
1789 # list[156][1] = list[157]
1790 #
1791 # [cmd, kind] = list[157][1]
1792 # kind[157] == 'f' and not
1793 # list.index(1)
1794 # list[157][1] = list[158]
1795 #
1796 # [cmd, kind] = list[158][1]
1797 # kind[158] == 'f' and not
1798 # list.index(1)
1799 # list[158][1] = list[159]
1800 #
1801 # [cmd, kind] = list[159][1]
1802 # kind[159] == 'f' and not
1803 # list.index(1)
1804 # list[159][1] = list[160]
1805 #
1806 # [cmd, kind] = list[160][1]
1807 # kind[160] == 'f' and not
1808 # list.index(1)
1809 # list[160][1] = list[161]
1810 #
1811 # [cmd, kind] = list[161][1]
1812 # kind[161] == 'f' and not
1813 # list.index(1)
1814 # list[161][1] = list[162]
1815 #
1816 # [cmd, kind] = list[162][1]
1817 # kind[162] == 'f' and not
1818 # list.index(1)
1819 # list[162][1] = list[163]
1820 #
1821 # [cmd, kind] = list[163][1]
1822 # kind[163] == 'f' and not
1823 # list.index(1)
1824 # list[163][1]
```

Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: <https://aka.ms/wslusers>
Enter new UNIX username: admin

```
UNIX username: me  
passwd: diana  
$ sudo passwd root  
[sudo] password for me: diana
```

<https://positivemh.tistory.com/583>

```
me@DESKTOP-HAAIOJO: ~  
Retype new password:  
passwd: password updated successfully  
Installation successful!  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Sat Jul  9 14:52:19 KST 2022  
  
System load:  0.06      Processes:            8  
Usage of /:   0.5% of 250.98GB   Users logged in:    0  
Memory usage: 0%      IPv4 address for eth0: 172.30.199.88  
Swap usage:   0%  
  
1 update can be applied immediately.  
To see these additional updates run: apt list --upgradable  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
This message is shown once a day. To disable it please create the  
/home/me/.hushlogin file.  
me@DESKTOP-HAAIOJO: ~$
```

C:\> 관리자: 명령 프롬프트

Microsoft Windows [Version 10.0.19044.1766]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>wsl --set-default-version 2_

C:\ 관리자: 명령 프롬프트

Microsoft Windows [Version 10.0.19044.1766]

(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>wsl --set-default-version 2

WSL 2와의 주요 차이점에 대한 자세한 내용은 <https://aka.ms/wsl2>를 참조하세요
작업을 완료했습니다.

C:\Windows\system32>_



앱



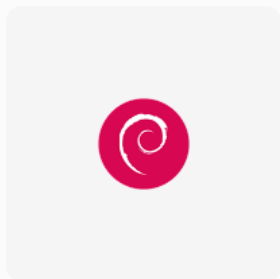
게임



라이브러리



도움말



Debian

[The Debian Project](#)

설치

5.0 ★

평균

5

평점

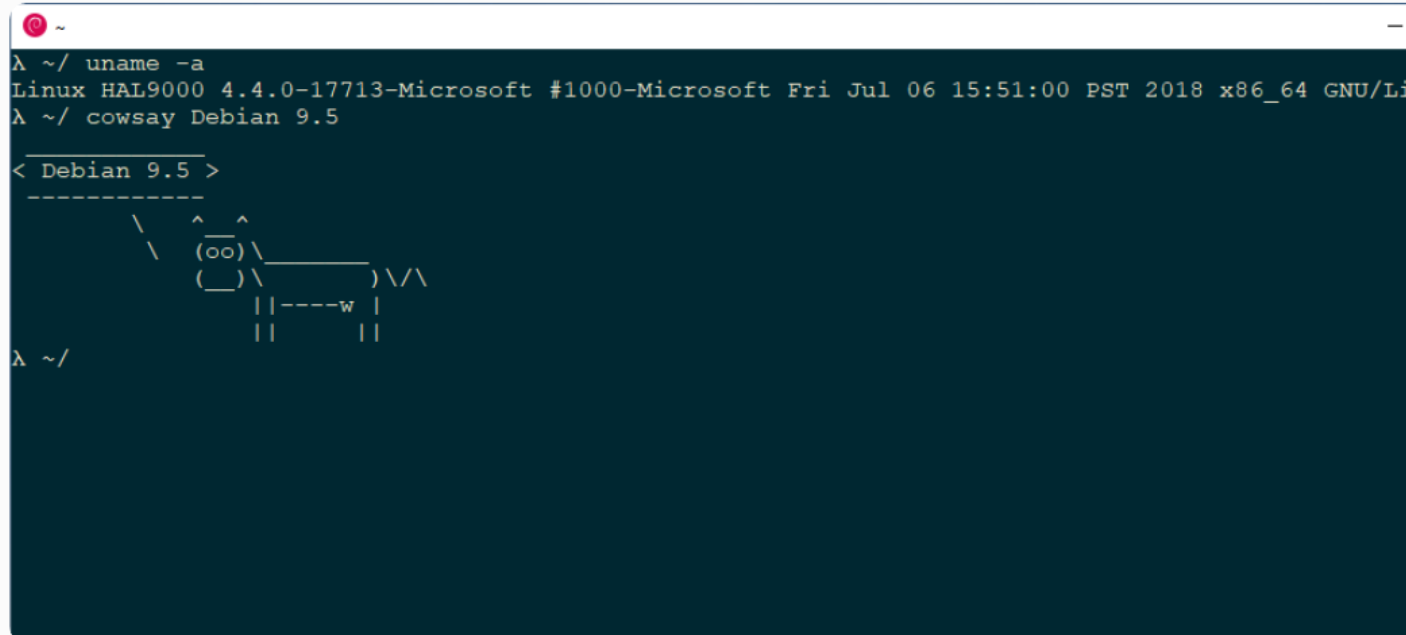


3+

연령별 등급: IARC. 3+



스크린샷



설명

With this app you get Debian for the Windows Subsystem for Linux (WSL).

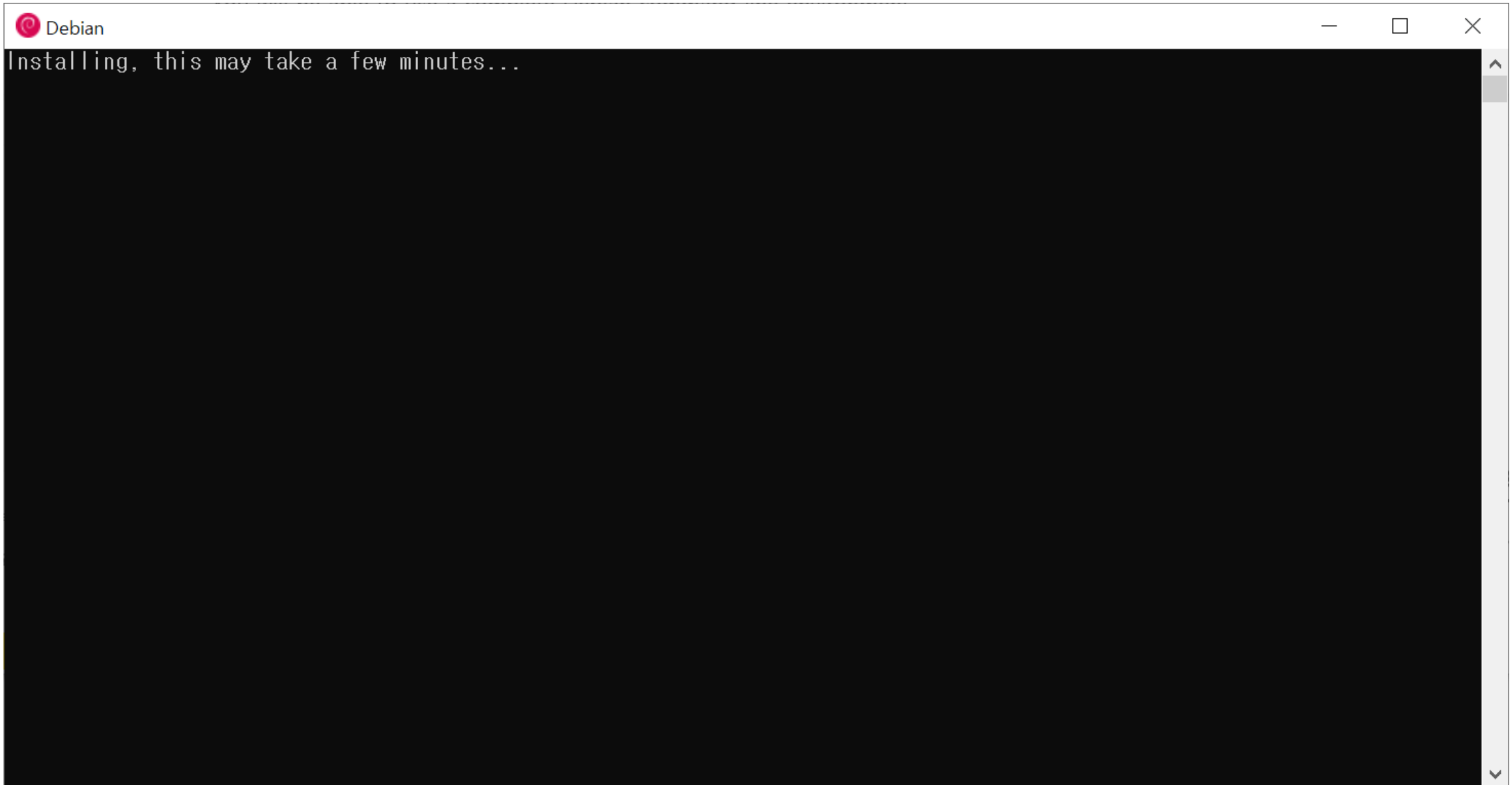
You will be able to use a complete Debian command line environment containing a full current stable release environment.

If this is your first WSL app you might have to enable WSL first (or contact your device administrator to do so).

<https://docs.microsoft.com/en-us/windows/wsl/install>

After the installation you can start the WSL console by either clicking the "Debian" tile in the start

Debian 설치 과정



Debian 설치 과정

```
oakyo@DESKTOP-HAAIOJO: ~  
Installing, this may take a few minutes...  
Please create a default UNIX user account. The username does not need to match your Windows username.  
For more information visit: https://aka.ms/wslusers  
Enter new UNIX username: oakyo  
New password:  
Retype new password:  
passwd: password updated successfully  
Installation successful!  
oakyo@DESKTOP-HAAIOJO:~$
```

Python 설치 과정(Anaconda)

```
$ sudo apt update
```

```
$ sudo apt upgrade
```

```
$ sudo apt-get install wget
```

Anaconda 설치 : <https://repo.anaconda.com/archive/>

```
$ wget https://repo.anaconda.com/archive/Anaconda3-2021.11-Linux-x86\_64.sh
```

```
$ bash Anaconda3-2021.11-Linux-x86_64.sh
```

PATH 추가

```
$ export PATH=/home/{login ID}/anaconda3/bin:$PATH
```

Python 설치 과정

- Python3 설치

```
$ sudo apt install python3
```

- Python3-venv 설치

```
$ sudo apt install python3-venv
```

- Python3-venv 설치

```
$ sudo apt install python3-venv
```

- 가상 환경 세팅

```
$ python3 -m venv ~/virtualenv
```

```
$ source ~/virtualenv/bin/activate
```

Python version 확인

\$ python -version

PC(Local) 파일 확인

\$ explorer.exe .

Java 설치 과정

```
$ sudo apt install default-jre
```

```
$ sudo apt install default-jdk
```

```
$ export JAVA_HOME=/usr/lib/jvm/java-1.11.0-openjdk-amd64
```

```
$ java -version
```

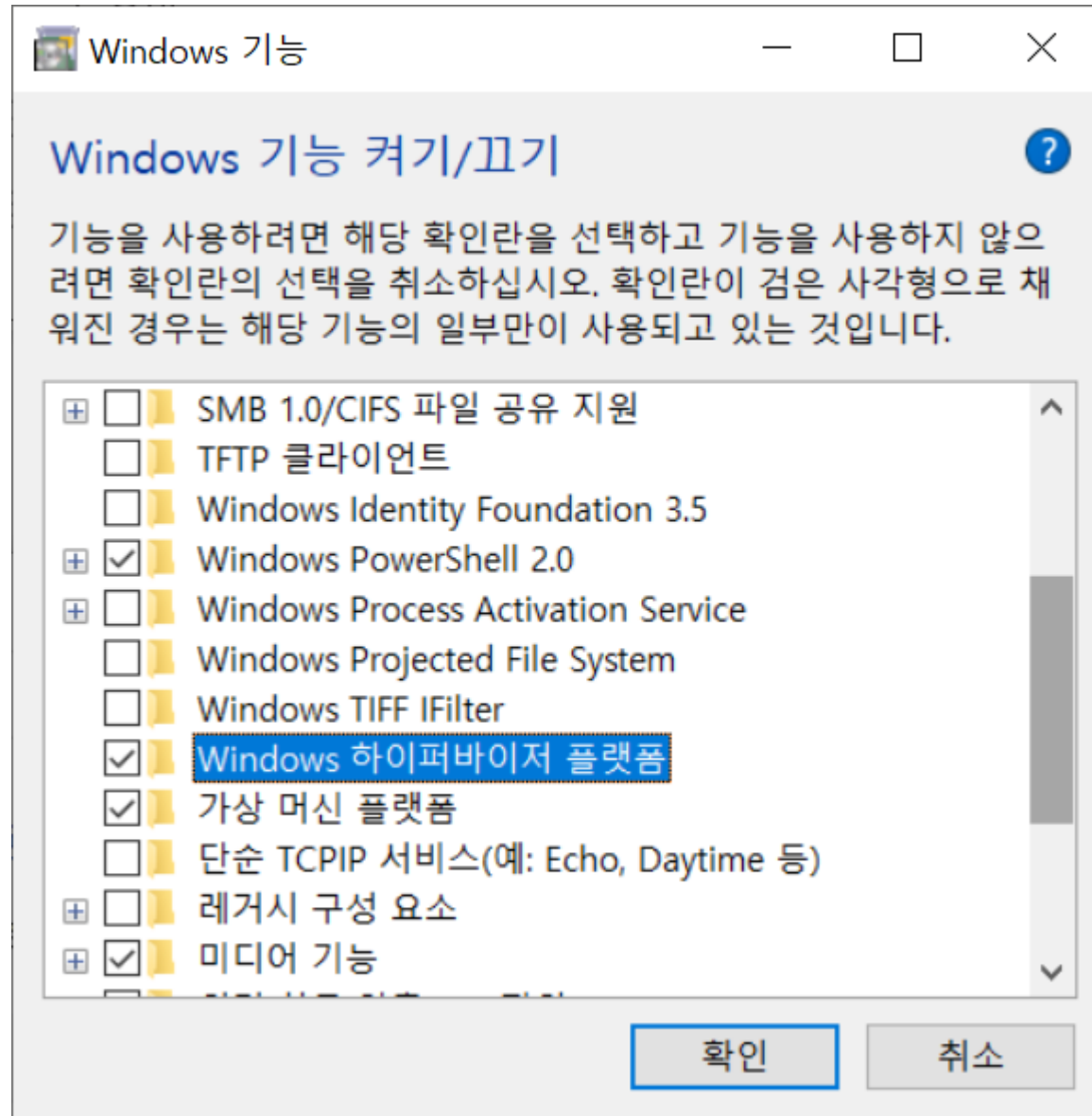
<https://serverspace.io/support/help/how-to-install-java-with-apt-on-ubuntu-18-04/>

SSH 설치 과정

```
$ ssh-keygen -t rsa          # ssh key 생성
$ cd
$ cat .ssh/id_rsa.pub >> .ssh/authorized_keys
$ sudo apt-get install openssh-server
$ sudo systemctl status ssh  # The system confirms that the SSH service is running.
$ sudo service ssh start (or stop)

$ ssh <ID>
$ sudo apt install nettools
$ netstat -a    # port 확인
```

Kubernetes Getting Started on Windows



The screenshot shows the Docker Docs website for the 'Install Docker Desktop on Windows' page. The browser's address bar shows the URL `docs.docker.com/desktop/install/windows-install/`. The page has a blue header with the Docker logo and navigation links: Home, Guides, Manuals, Reference, Samples, and Contribute. The breadcrumb trail is: Home / Manuals / Docker Desktop / Install Docker Desktop / Install on Windows.

Install Docker Desktop on Windows
Estimated reading time: 9 minutes

Docker Desktop terms

Commercial use of Docker Desktop in larger enterprises (more than 250 employees OR more than \$10 million USD in annual revenue) requires a paid subscription.

Welcome to Docker Desktop for Windows. This page contains information about Docker Desktop for Windows system requirements, download URL, instructions to install and update Docker Desktop for Windows.

Download Docker Desktop for Windows

[Docker Desktop for Windows](#)

For checksums, see [Release notes](#)

System requirements

Your Windows machine must meet the following requirements to successfully install Docker Desktop.

[WSL 2 backend](#) [Hyper-V backend and Windows containers](#)

WSL 2 backend

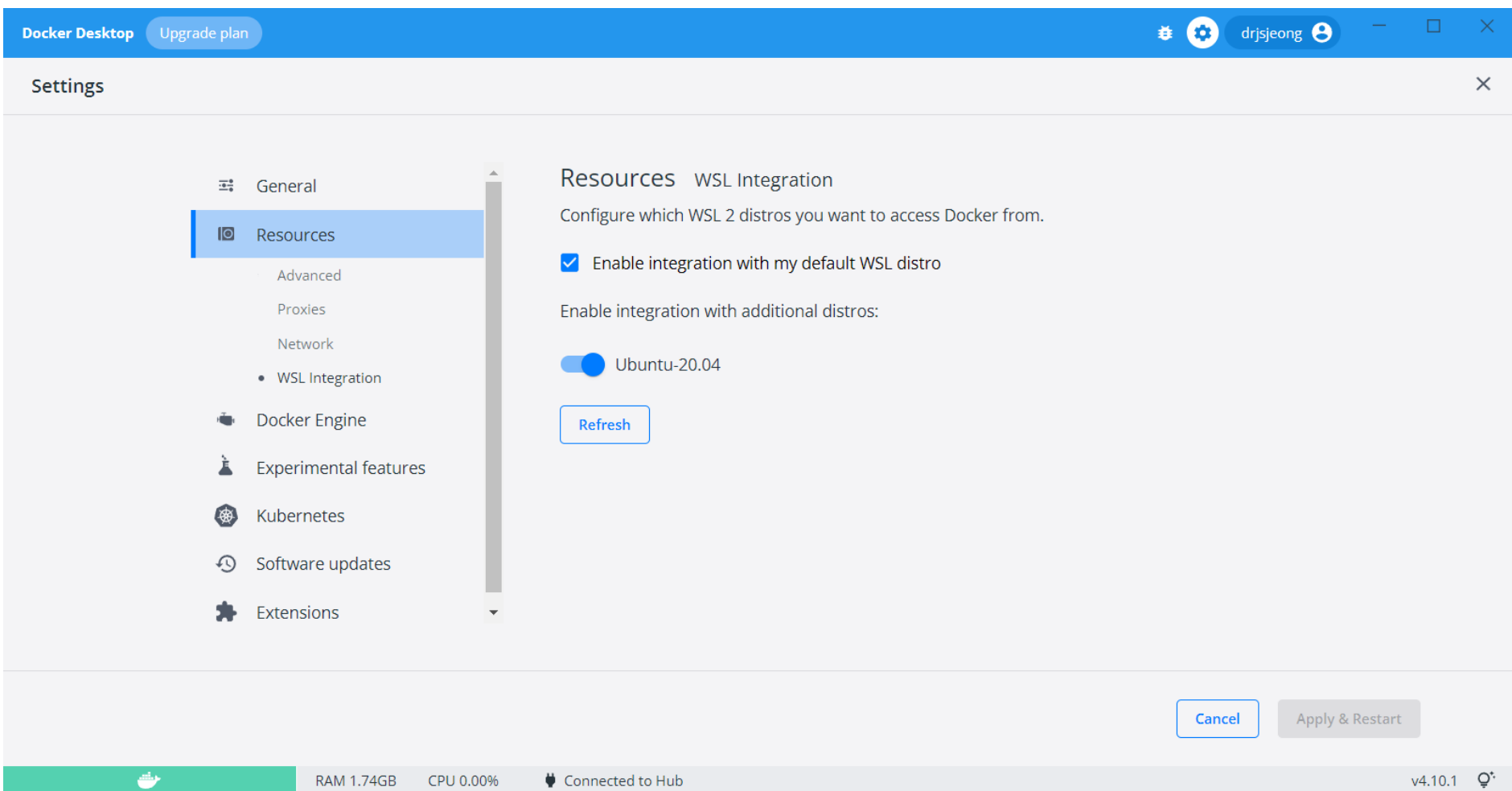
- Windows 11 64-bit: Home or Pro version 21H2 or higher, or Enterprise or Education version 21H2 or higher.
- Windows 10 64-bit: Home or Pro 21H1 (build 19043) or higher, or Enterprise or Education 20H2 (build 19042) or higher.
- Enable the WSL 2 feature on Windows. For detailed instructions, refer to the [Microsoft documentation](#).
- The following hardware prerequisites are required to successfully run WSL 2 on Windows 10 or Windows 11:

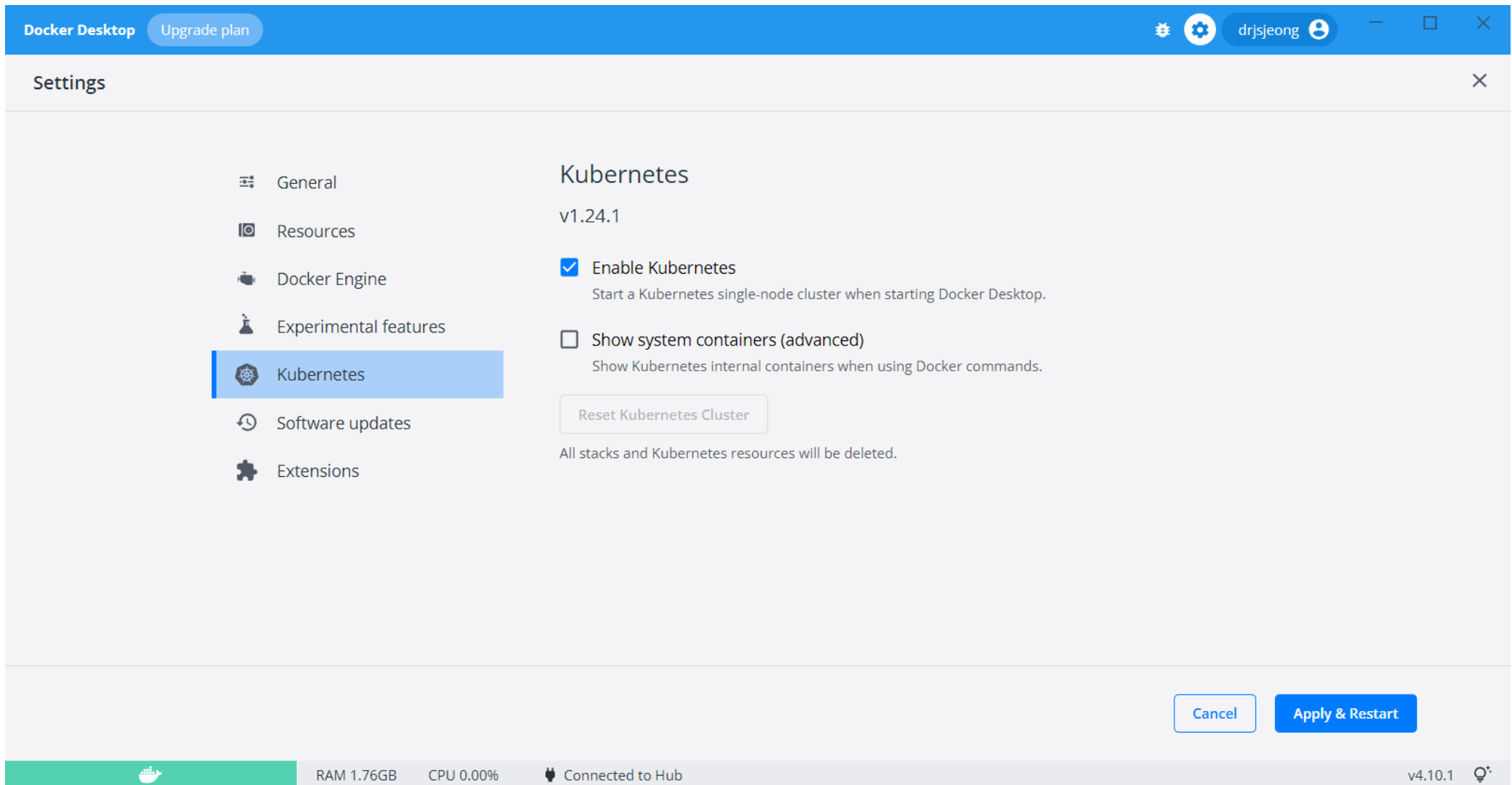
Contents:

- System requirements
- WSL 2 backend
- Hyper-V backend and Windows containers
- About Windows containers
- Install Docker Desktop on Windows
 - Install interactively
 - Install from the command line
- Start Docker Desktop
- Updates
- Uninstall Docker Desktop
- Where to go next

[Edit this page](#)
[Request changes](#)




<https://docs.docker.com/desktop/install/windows-install/>










Docker Desktop


Upgrade plan


 drjsjeong 





Settings 


 General


 Resources

 Docker Engine

 Experimental features

 Kubernetes

 Software updates

 Extensions

Kubernetes

v1.24.1

☒ Enable Kubernetes
Starting ...



☐ Show system containers (advanced)
Show Kubernetes internal containers when using Docker commands.


Reset Kubernetes Cluster


All stacks and Kubernetes resources will be deleted.

Cancel

Apply & Restart



RAM 1.88GB CPU 0.07%  Connected to Hub

v4.10.1 

Kubectl 설치

<https://kubernetes.io/ko/docs/tasks/tools/>

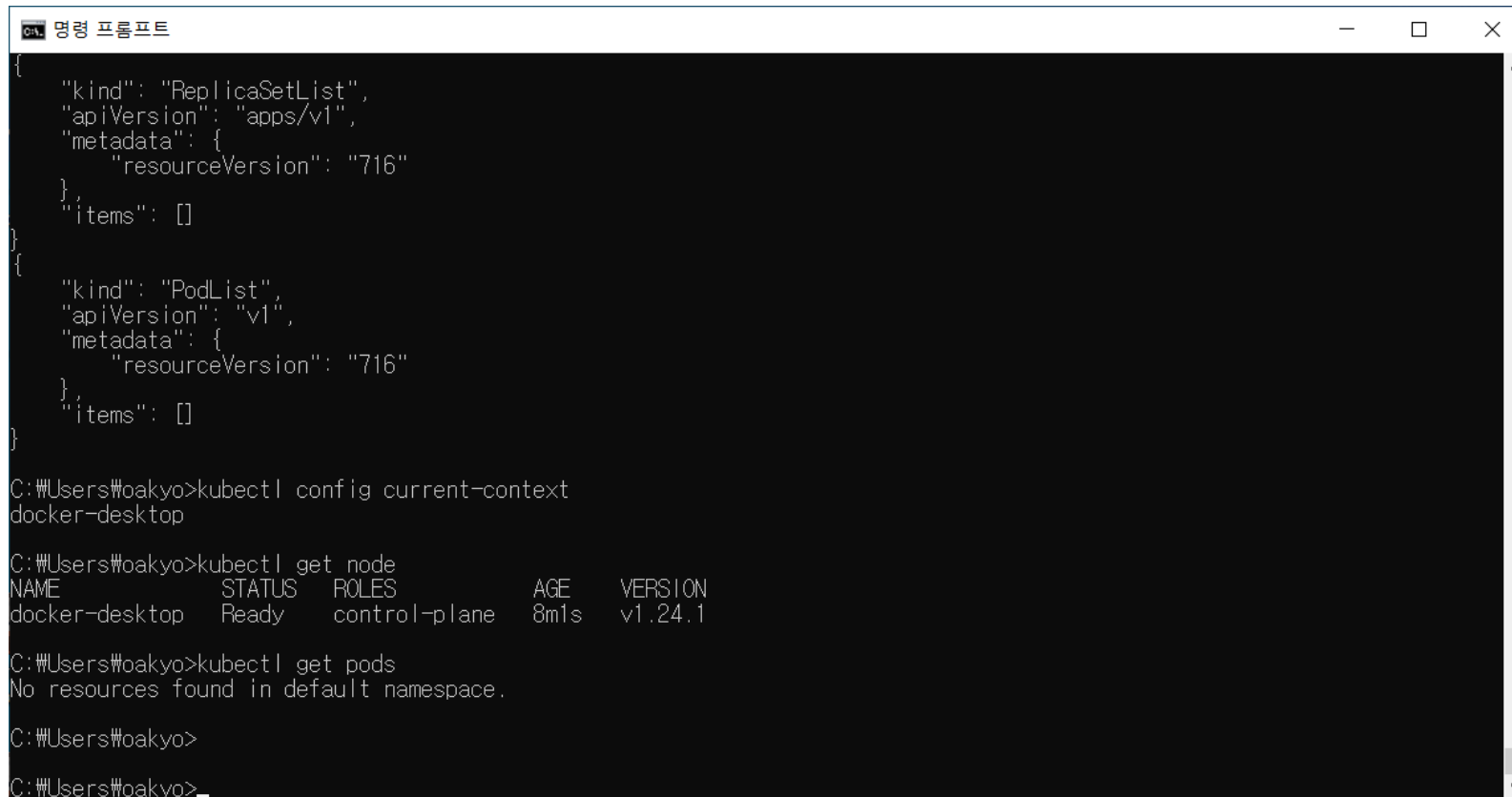
[```
\$ curl -LO "https://dl.k8s.io/release/v1.25.0/bin/windows/amd64/kubect
```](https://kubernetes.io/ko/docs/tasks/tools/install-kubect</a>l-windows/</p></div><div data-bbox=)

## Kubernetes 설치 확인

C> kubectl config current-context

C> kubectl get node

C> kubectl get pods



```
C:\Users\oakyo> kubectl config current-context
docker-desktop

C:\Users\oakyo> kubectl get node
NAME STATUS ROLES AGE VERSION
docker-desktop Ready control-plane 8ms v1.24.1

C:\Users\oakyo> kubectl get pods
No resources found in default namespace.

C:\Users\oakyo>
```





## 정 준 수 / PhD ( jsjeong@hansung.ac.kr )

- 前) 삼성전자 연구원
- 前) 삼성의료원 (삼성생명과학연구소)
- 前) 삼성SDS (정보기술연구소)
- 現) (사)한국인공지능협회, AI, 머신러닝 강의
- 現) 한국소프트웨어산업협회, AI, 머신러닝 강의
- 現) 서울디지털재단, AI 자문위원
- 現) 한성대학교 교수(겸)
- 전문분야: Computer Vision, 머신러닝(ML), RPA
- <https://github.com/JSJeong-me/>

