

NSIS

Setting Environment

1. NSIS Install; Nullsoft Install System을 설치한다.
2. HM NIS Edit Install; HM NIS Edit를 설치한다
3. Install Plugins; 아래 플러그인을 NSIS 설치한 위치의 Plugins 디렉토리로 복사한다.
 - EnVar-Plugin
 - Nsis7z plug-in

Preparing Files

- 설치 패키지를 위한 파일을 준비한다. (ustra.7z = VSCode + STS4 + git + node.js + yarn + ..., ustra.ico)
- NSIS Script file; 설치 패키지를 만드는 스크립트, *.nsi (ustra-dev-tools.nsi)
- Prepare files to install; 설치할 파일을 준비한다.
 - make 7z file into Input directory; 생성한 ustra.7z 파일을 Input 디렉토리에 넣는다.

```
├─tools
│   ├───apache-maven-3.8.1
│   ├───gradle-6.9.2
│   ├───java
│   │   └───java-1.8.0-openjdk-1.8.0.212-1
│   ├───sts-4.7.1
│   ├───VSCode-win32-x64-1.48.0
│   ├───Git-2.37.0-64-bit.exe
│   ├───node-v12.22.9-x64.msi
│   └───yarn-1.22.19.msi
└─workspace
```

- Create Output Directory; 설치 패키지의 최종 결과 파일을 저장할 디렉토리를 생성한다.
- Structures; 설치 패키지를 위한 디렉토리 구조는 아래와 같다.

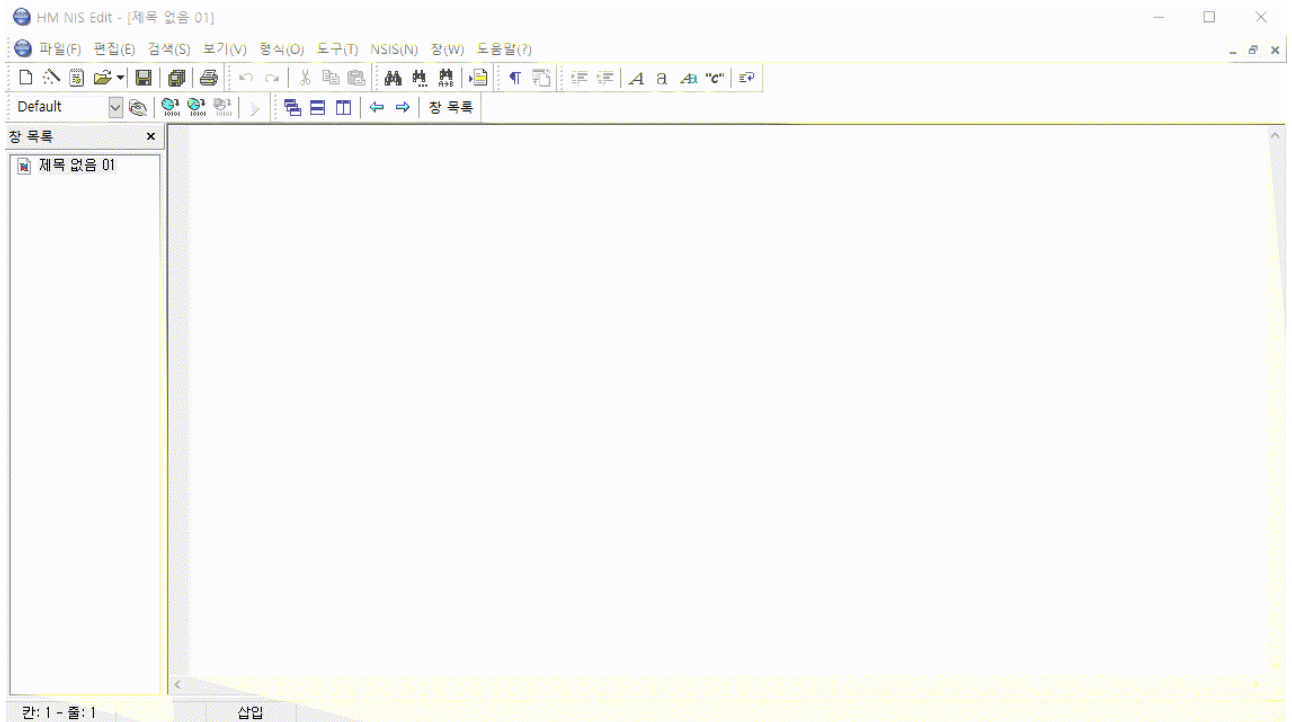
```
└─*.nsi
  ├───Input
  │   ├───ustra.ico
  │   └───ustra.7z
  └───Output
      └───executable setup file will be here.
```

Scripts

- HM NIS Edit 를 실행한다.



- 스크립트를 작성하거나 작성된 스크립트 파일을 연다.



```
; Script generated by the HM NIS Edit Script Wizard.

; HM NIS Edit Wizard helper defines
!define PRODUCT_NAME "U.STRa Framework 개발 환경"
!define PRODUCT_VERSION "0.1"
!define PRODUCT_UNINST_KEY
"Software\Microsoft\Windows\CurrentVersion\Uninstall\${PRODUCT_NAME}"
!define PRODUCT_UNINST_ROOT_KEY "HKLM"

; MUI 1.67 compatible -----
!include "MUI.nsh"

; MUI Settings
!define MUI_ABORTWARNING
;!define MUI_ICON "${NSISDIR}\Contrib\Graphics\Icons\modern-
install.ico"
!define MUI_ICON ".\Input\ustra.ico"
;!define MUI_UNICON "${NSISDIR}\Contrib\Graphics\Icons\modern-
uninstall.ico"
!define MUI_UNICON ".\Input\ustra.ico"
```

```
; Welcome page
!insertmacro MUI_PAGE_WELCOME
; Directory page
!insertmacro MUI_PAGE_DIRECTORY
; Instfiles page
;!define MUI_FINISHPAGE_NOAUTOCLOSE
!insertmacro MUI_PAGE_INSTFILES
; Finish page
!insertmacro MUI_PAGE_FINISH

; Uninstaller pages
!insertmacro MUI_UNPAGE_INSTFILES

; Language files
!insertmacro MUI_LANGUAGE "Korean"

; MUI end -----

Name "${PRODUCT_NAME} ${PRODUCT_VERSION}"
OutFile ".\Output\Setup_Ustra_Dev.exe"
;InstallDir "$PROGRAMFILES\U.STRA Framework 개발 환경"
InstallDir "C:\ustra_devtools"
ShowInstDetails show
ShowUnInstDetails show

; Global Variables -----
Var fileSTS4Ini
Var fileGitInstaller
Var fileNodejsInstaller
Var fileYarnInstaller

Function InitInstallers
    StrCpy $fileSTS4Ini "$INSTDIR\tools\sts-4.7.1\SpringToolSuite4.ini"
    StrCpy $fileGitInstaller "$INSTDIR\tools\Git-2.37.0-64-bit.exe"
    StrCpy $fileNodejsInstaller "$INSTDIR\tools\node-v12.22.9-x64.msi"
    StrCpy $fileYarnInstaller "$INSTDIR\tools\yarn-1.22.19.msi"
FunctionEnd

Function WriteSTS4IniFile
    ; Write SpringToolSuite4.ini file
    ;FileOpen $9 $INSTDIR\tools\sts-4.7.1\SpringToolSuite4.ini w          ;
    Opens a Empty File and fills it
    FileOpen $9 $fileSTS4Ini w          ; Opens a Empty File and fills it
    FileWrite $9 "-startup$\r$\n\
plugins/org.eclipse.equinox.launcher_1.5.700.v20200207-2156.jar$\r$\n\
    --launcher.library$\r$\n\
plugins/org.eclipse.equinox.launcher.win32.win32.x86_64_1.1.1200.v20200
508-1552$\r$\n\
    -product$\r$\n\
    org.springframework.boot.ide.branding.sts4$\r$\n\"
```

```
        --launcher.defaultAction$r$\n\
        openFile$r$\n\
        -vm$r$\n\
        $INSTDIR\tools\java\java-1.8.0-
openjdk-1.8.0.212-1\bin\javaw.exe$r$\n\
        -vmargs$r$\n\
        -Dosgi.requiredJavaVersion=1.8$r$\n\
        -Dosgi.dataAreaRequiresExplicitInit=true$r$\n\
        -Xms256m$r$\n-Xmx2048m$r$\n\
        --add-modules=ALL-SYSTEM$r$\n\
        -javaagent:$INSTDIR\tools\sts-4.7.1\lombok.jar$r$\n\
        -Dfile.encoding=UTF-8$r$\n"
    FileClose $9 ; Closes the filled file
FunctionEnd

Function AddEnvironmentVariablesInHKCU
    ; add Environment Variables
    EnVar::Check "temp" "NULL"
    Pop $0
    DetailPrint "EnVar::Check temp:NULL returned=|$0|"
    ; JAVA_HOME
    EnVar::AddValue "JAVA_HOME" "$INSTDIR\tools\java\java-1.8.0-
openjdk-1.8.0.212-1"
    Pop $0
    DetailPrint "EnVar::AddValue JAVA_HOME returned=|$0|"
    ; GRADLE_HOME
    EnVar::AddValue "GRADLE_HOME" "$INSTDIR\tools\gradle-6.9.2"
    Pop $0
    DetailPrint "EnVar::AddValue GRADLE_HOME returned=|$0|"
    ; MAVEN_HOME
    EnVar::AddValue "MAVEN_HOME" "$INSTDIR\tools\apache-maven-3.8.1"
    Pop $0
    DetailPrint "EnVar::AddValue MAVEN_HOME returned=|$0|"
FunctionEnd

Function AddPathsInHKCU
    ; add Paths
    EnVar::AddValueEx "Path" "%JAVA_HOME%\bin"
    Pop $0
    DetailPrint "EnVar::AddValue Path %JAVA_HOME%\bin = |$0|"
    EnVar::AddValueEx "Path" "%GRADLE_HOME%\bin"
    Pop $0
    DetailPrint "EnVar::AddValue Path %GRADLE_HOME%\bin = |$0|"
    EnVar::AddValueEx "Path" "%MAVEN_HOME%\bin"
    Pop $0
    DetailPrint "EnVar::AddValue Path %MAVEN_HOME%\bin = |$0|"
    ;DetailPrint "EnVar::Update HKLM JAVA_HOME returned=|$0|"
    ;EnVar::Check "NULL" "NULL"
    ;Pop $0
    ;DetailPrint "EnVar::Check write access HKCU returned=|$0|"
FunctionEnd
```

```
Function InstallGit
    ; Install git
    ; ExecWait "$INSTDIR\tools\Git-2.36.1-64-bit.exe"
    ExecWait $fileGitInstaller ;"$INSTDIR\tools\Git-2.37.0-64-bit.exe"
FunctionEnd

Function InstallNodejs
    ; Install Node.js v12.22.9
    ${If} $R0 == ""
        ;ExecWait 'C:\Windows\SysWOW64\msiexec.exe /i "$INSTDIR\tools\node-
v12.22.9-x64.msi"'
        ExecWait 'C:\Windows\SysWOW64\msiexec.exe /i $fileNodejsInstaller'
    ${Else}
        ${If} $R0 == "12.22.9"
            ; already installed, skip this step
        ${Else}
            ; uninstall different version and install v12.22.9 version
            ExecWait "C:\Windows\SysWOW64\msiexec.exe /x {C1B586E6-C075-42F3-
B7E4-0DE4222DEEC0}"
            ;ExecWait 'C:\Windows\SysWOW64\msiexec.exe i
"$INSTDIR\tools\node-v12.22.9-x64.msi"'
            ExecWait 'C:\Windows\SysWOW64\msiexec.exe i $fileNodejsInstaller'
        ${EndIf}
    ${EndIf}
FunctionEnd

Function InstallYarn
    ; Install Yarn
    ; ExecWait 'C:\Windows\SysWOW64\msiexec.exe /i
"$INSTDIR\tools\yarn-1.22.4.msi"'
    ExecWait 'C:\Windows\SysWOW64\msiexec.exe /i $fileYarnInstaller'
FunctionEnd

Function CreateDesktopShortcuts
    ; Create Desktop Shortcuts
    CreateShortCut "$DESKTOP\U.STRA Framework 개발 환경 - VSCODE.lnk"
"$INSTDIR\tools\VSCode-win32-x64-1.48.0\Code.exe"
    CreateShortCut "$DESKTOP\U.STRA Framework 개발 환경 - STS.lnk"
"$INSTDIR\tools\sts-4.7.1\SpringToolSuite4.exe" "-data
$INSTDIR\workspace"
FunctionEnd

Section "MainSection" SEC01
    SetOutPath "$INSTDIR"
    SetOverwrite ifnewer
    File ".\Input\ustra.7z"
    ; First above all, Extract Compressed file
    Nsis7z::ExtractWithDetails "$INSTDIR\ustra.7z" "Unpacking %s..."

    ; Set Variables to Install
    Call InitInstallers
```

```
; Set SpringToolSuite4.ini File
Call WriteSTS4IniFile
; Install Utilities
; Install Node.js
Call InstallNodejs
; Install Yarn
Call InstallYarn
; Install Git
Call InstallGit
; Set Environment Variables, HKEY_CURRENT_USER
; Add Environment Values; JAVA_HOME, GRADLE_HOME, MAVEN_HOME
Call AddEnvironmentVariablesInHKCU
; Add Paths; %JAVA_HOME%\bin, %GRADLE_HOME%\bin, %MAVEN_HOME%\bin
Call AddPathsInHKCU
; Create Desktop Shortcuts for VSCode, STS4
Call CreateDesktopShortcuts
; When installation completed, delete compressed file
Delete "$INSTDIR\ustra.7z"

SectionEnd

Section -AdditionalIcons
    CreateDirectory "$SMPROGRAMS\U.STRA Framework 개발 환경"
    CreateShortCut "$SMPROGRAMS\U.STRA Framework 개발 환경\Uninstall.lnk"
"$INSTDIR\uninst.exe"
SectionEnd

Section -Post
    WriteUninstaller "$INSTDIR\uninst.exe"
    WriteRegStr ${PRODUCT_UNINST_ROOT_KEY} "${PRODUCT_UNINST_KEY}"
"DisplayName" "$(^Name)"
    WriteRegStr ${PRODUCT_UNINST_ROOT_KEY} "${PRODUCT_UNINST_KEY}"
"UninstallString" "$INSTDIR\uninst.exe"
    WriteRegStr ${PRODUCT_UNINST_ROOT_KEY} "${PRODUCT_UNINST_KEY}"
"DisplayVersion" "${PRODUCT_VERSION}"
SectionEnd

Function un.onUninstSuccess
    HideWindow
    MessageBox MB_ICONINFORMATION|MB_OK "$(^Name)는(은) 완전히 제거되었습니다."
FunctionEnd

Function un.onInit
    MessageBox MB_ICONQUESTION|MB_YESNO|MB_DEFBUTTON2 "$(^Name)을(를) 제거하
시겠습니까?" IDYES +2
    Abort
FunctionEnd

Section Uninstall
; Delete "$INSTDIR\uninst.exe"
```

```

; Delete "$INSTDIR\ustra.7z"
; Delete "$INSTDIR\tools\sts-4.7.1\SpringToolSuite4.ini"

; Delete environment variables
; JAVA_HOME
; EnVar::Delete "JAVA_HOME"
EnVar::DeleteValue "JAVA_HOME" "$INSTDIR\tools\java\java-1.8.0-
openjdk-1.8.0.212-1"
; GRADLE_HOME
; EnVar::Delete "GRADLE_HOME"
EnVar::DeleteValue "GRADLE_HOME" "$INSTDIR\tools\gradle-6.9.2"
; MAVEN_HOME
; EnVar::Delete "MAVEN_HOME"
EnVar::DeleteValue "MAVEN_HOME" "$INSTDIR\tools\apache-maven-3.8.1"
EnVar::Delete ""

; Delete Path
EnVar::DeleteValue "Path" "%JAVA_HOME%\bin"
EnVar::DeleteValue "Path" "%GRADLE_HOME%\bin"
EnVar::DeleteValue "Path" "%MAVEN_HOME%\bin"

Delete "$SMPROGRAMS\U.STRA Framework 개발 환경\Uninstall.lnk"

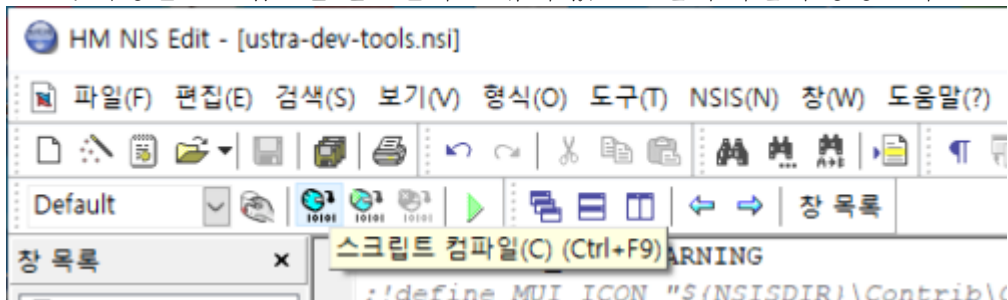
Delete "$DESKTOP\U.STRA Framework 개발 환경 - VSCODE.lnk"
Delete "$DESKTOP\U.STRA Framework 개발 환경 - STS.lnk"
RMDir "$SMPROGRAMS\U.STRA Framework 개발 환경"
RMDir /r "$INSTDIR"

DeleteRegKey ${PRODUCT_UNINST_ROOT_KEY} "${PRODUCT_UNINST_KEY}"
SetAutoClose true
SectionEnd

```

Build & Test

- Build; 작성한 스크립트를 빌드한다. 오류가 없으면 설치 파일이 생성된다.



```

Command Line: "C:\Program Files (x86)\NSIS\makensis.exe" /S /O "C:\Users\GITH\Documents\Setup_Ultra_Dev\Ultra-Dev-nsis.exe"
Processing config: C:\Program Files (x86)\NSIS\makensis.conf.nsh
Processing script: C:\Users\GITH\Documents\Setup_Ultra_Dev\Ultra-Dev-nsis.nsi (MCP)
Processed 1 file, writing output (MCP-unicode):
Output: "C:\Users\GITH\Documents\Setup_Ultra_Dev\Output\Setup_Ultra_Dev.exe"
Install: 4 pages (256 bytes), 3 sections (2 required) (4024 bytes), 436 instructions (12360 bytes), 206 strings (7264 bytes), 1 language table (192 bytes).
Uninstall: 1 page (128 bytes), 1 section (2072 bytes), 103 instructions (2884 bytes), 98 strings (2670 bytes), 1 language table (220 bytes).
Using x86 compression.
EXE header size: 147360 / 30036 bytes
Install code: 4596 / 24564 bytes
Install data: 9149217 / 91492736 bytes
Uninstall code/data: 21456 / 11564 bytes
CRC (sha512): 4 / 4 bytes
Total size: 91492470 / 91493420 bytes (99.9%)
  
```

• Test on Windows

- 설치 테스트를 위한 Windows 운영체제를 준비한다; 포맷 후 Windows만 설치한 PC 혹은 Hyper-V, VMWare, Virtual Box 등의 가상 머신에 Windows 설치. (가능한 한 응용 프로그램을 설치하지 않은 Windows 운영체제를 추천)
- 준비한 Windows 운영체제에 빌드한 설치 파일을 복사한다.
- 설치 파일을 이용하여 프로그램을 설치한다.
- 설치한 프로그램 동작을 확인한다.
 - SpringToolSuite4
 - VSCode
 - Node.js
 - Yarn
 - Git
- 시스템 환경 변수 편집 혹은 CMD 등에서 환경 설정 내용을 확인한다; JAVA_HOME, GRADLE_HOME, MAVEN_HOME, Path
- 확인 후 의도한 대로 설치되지 않았으면 스크립트 수정 후 빌드 → 테스트를 다시 진행하고, 의도한 대로 설치되었다면 다른 Windows 시스템에도 테스트 진행하여 확인한다. 다시 테스트를 진행하기 전에 반드시 이전 설치 및 설정된 내역을 리셋한다. (Uninstall, Windows 복귀 등)
- 확인이 완료되면 배포한다.

References

- 개발 환경 및 관련 리소스는 아래의 링크에서 확인할 수 있다.
 - [NSIS Download](#)
 - [NSIS Users Manual](#)
 - [NSIS Wiki](#)
 - [HM NIS EDIT: A Free NSIS Editor/IDE](#)
 - [Nsis7z plugin](#)
 - [EvVar Plugin for NSIS](#)
 - [NSIS 사용자 설명서 @opentutorials.org](#)



From:

<https://theta5912.net/> - reth

Permanent link:

<https://theta5912.net/doku.php?id=public:computer:nsis&rev=1658299431>

Last update: **2022/07/20 15:43**

