

Secure Coding Lab – 4

Dasari Mohan Amrithesh

18BCN7038

Write a python script to print hello world:

```
print("Hello world")
```

```
x=input("Press Enter to exit:")
```

A screenshot of a text editor window titled "temp.py - C:\Users\dmam\Desktop\Secure lab temp\temp.py (3.8.5)". The editor shows the following code:

```
print("Hello world")
x=input("Press Enter to exit:")
```

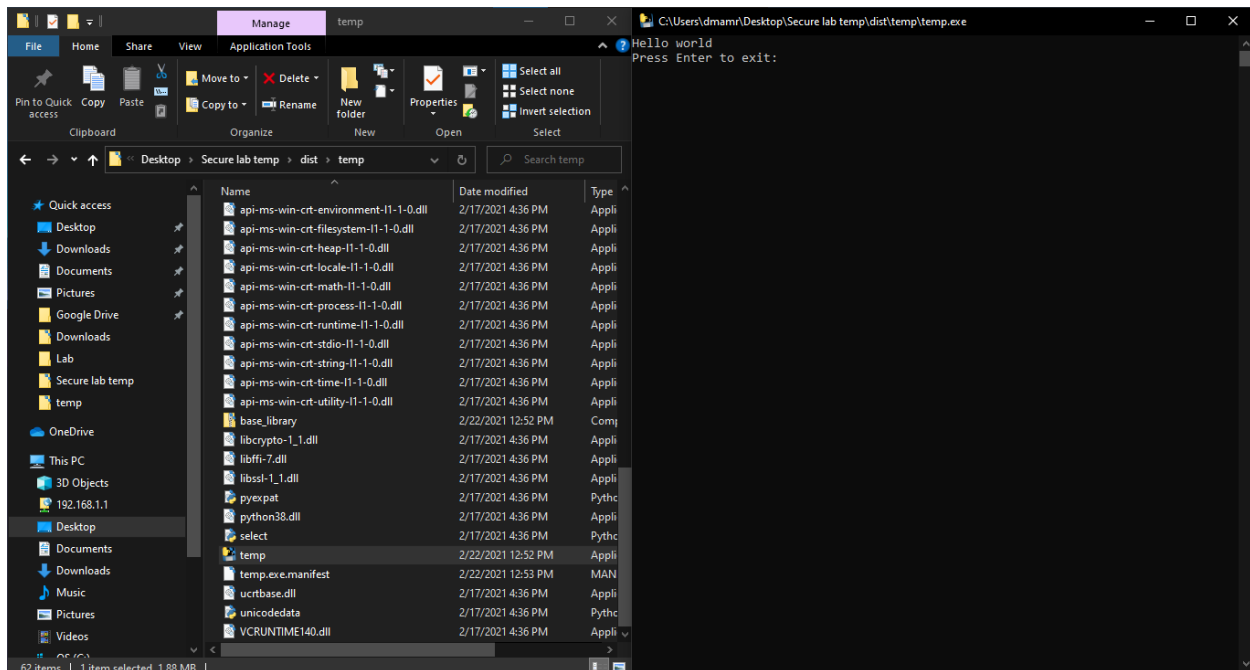
Convert the script into executable (use pyinstaller or py2exe – any of your choice).:

C:\Users\dmam\Desktop\Secure lab temp>pyinstaller temp.py

A screenshot of a Windows Command Prompt window showing the output of the command "pyinstaller temp.py". The output is a long list of informational messages from PyInstaller, including details about the Python version (3.8.5), platform (Windows-10-10.0.19041-SP0), and the various steps involved in building the executable, such as analyzing the code, building the PYZ archive, and writing the final executable file. The process completes successfully, and the final executable is located at "C:\Users\dmam\Desktop\Secure lab temp\build\temp\temp.exe".

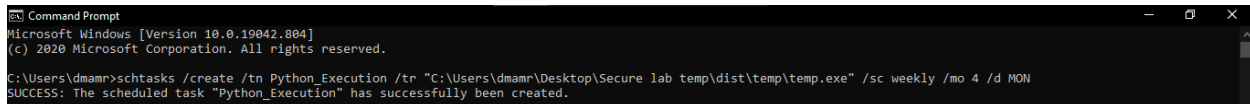
```
Select Command Prompt
C:\Users\dmam\Desktop\Secure lab temp>pyinstaller temp.py
938 INFO: PyInstaller: 4.2
938 INFO: Python: 3.8.5
939 INFO: Platform: Windows-10-10.0.19041-SP0
940 INFO: wrote C:\Users\dmam\Desktop\Secure lab temp\temp.spec
945 INFO: UPX is not available.
954 INFO: Extending PYTHONPATH with paths
['C:\Users\dmam\Desktop\Secure lab temp',
 'C:\Users\dmam\Desktop\Secure lab temp']
1099 INFO: checking Analysis
1099 INFO: Building Analysis because Analysis-00.toc is non-existent
1099 INFO: Initializing module dependency graph...
1102 INFO: Caching module graph hooks...
1171 WARNING: Several hooks defined for module 'win32ctypes.core'. Please take care they do not conflict.
1255 INFO: Analyzing base_library.zip ...
4953 INFO: Processing pre-find module path hook distutils from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks\pre_find_module_path\hook-distutils.py'.
4971 INFO: distutils: retargeting to non-venv dir 'e:\python 3.8.5\lib'
11534 INFO: Caching module dependency graph...
11683 INFO: running Analysis Analysis-00.toc
11687 INFO: Adding Microsoft.Windows.Common-Controls to dependent assemblies of final executable
required by e:\python 3.8.5\python.exe
12902 INFO: Analyzing C:\Users\dmam\Desktop\Secure lab temp\temp.py
12904 INFO: Processing module hooks...
12905 INFO: Loading module hook 'hook-difflib.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
12978 INFO: Excluding import of doctest from module difflib
12979 INFO: Loading module hook 'hook-distutils.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13002 INFO: Loading module hook 'hook-distutils.util.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13004 INFO: Excluding import of lib2to3.refactor from module distutils.util
13006 INFO: Loading module hook 'hook-encodings.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13083 INFO: Loading module hook 'hook-heapq.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13095 INFO: Excluding import of doctest from module heapq
13095 INFO: Loading module hook 'hook-lib2to3.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13409 INFO: Loading module hook 'hook-multiprocessing.util.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13501 INFO: Excluding import of test from module multiprocessing.util
13502 INFO: Excluding import of test.support from module multiprocessing.util
13502 INFO: Loading module hook 'hook-pickle.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13511 INFO: Excluding import of argparse from module pickle
13512 INFO: Loading module hook 'hook-sysconfig.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13523 INFO: Loading module hook 'hook-xml.etree.cElementTree.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13533 INFO: Loading module hook 'hook-xml.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13638 INFO: Loading module hook 'hook-tkinter.py' from 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks'...
13749 INFO: checking Tree
13749 INFO: Building Tree because Tree-00.toc is non-existent
13750 INFO: Building Tree Tree-00.toc
16148 INFO: checking Tree
16148 INFO: Building Tree because Tree-01.toc is non-existent
16149 INFO: Building Tree Tree-01.toc
16147 INFO: checking Tree
16147 INFO: Building Tree because Tree-02.toc is non-existent
16118 INFO: Building Tree Tree-02.toc
Select Command Prompt
16359 INFO: Looking for ctypes DLLs
16405 INFO: Analyzing run-time hooks ...
16409 INFO: Including run-time hook 'e:\python 3.8.5\lib\site-packages\PyInstaller\hooks\rthooks\pyi_rth_multiprocessing.py'
16440 INFO: Looking for dynamic libraries
16663 INFO: Looking for eggs
16863 INFO: Using Python library e:\python 3.8.5\python38.dll
16864 INFO: Found binding redirects:
[]
16870 INFO: Warnings written to C:\Users\dmam\Desktop\Secure lab temp\build\temp\warn-temp.txt
16934 INFO: Graph cross-reference written to C:\Users\dmam\Desktop\Secure lab temp\build\temp\xref-temp.html
16981 INFO: checking PYZ
16982 INFO: Building PYZ because PYZ-00.toc is non-existent
16983 INFO: Building PYZ (ZlibArchive) C:\Users\dmam\Desktop\Secure lab temp\build\temp\PYZ-00.pyz
18092 INFO: Building PYZ (ZlibArchive) C:\Users\dmam\Desktop\Secure lab temp\build\temp\PYZ-00.pyz completed successfully.
18108 INFO: checking PKG
18108 INFO: Building PKG because PKG-00.toc is non-existent
18108 INFO: Building PKG (CArchive) PKG-00.pkg
18188 INFO: Building PKG (CArchive) PKG-00.pkg completed successfully.
18190 INFO: Bootloader e:\python 3.8.5\lib\site-packages\PyInstaller\bootloader\Windows-64bit\run.exe
18190 INFO: checking EXE
18191 INFO: Building EXE because EXE-00.toc is non-existent
18191 INFO: Building EXE from EXE-00.toc
18923 INFO: Copying icons from ['e:\python 3.8.5\lib\site-packages\PyInstaller\bootloader\images\icon-console.ico']
19032 INFO: Writing RT_GROUP_ICON 0 resource with 104 bytes
19032 INFO: Writing RT_ICON 1 resource with 3752 bytes
19033 INFO: Writing RT_ICON 2 resource with 2216 bytes
19034 INFO: Writing RT_ICON 3 resource with 1304 bytes
19034 INFO: Writing RT_ICON 4 resource with 37019 bytes
19034 INFO: Writing RT_ICON 5 resource with 9640 bytes
19035 INFO: Writing RT_ICON 6 resource with 4264 bytes
19035 INFO: Writing RT_ICON 7 resource with 1128 bytes
19038 INFO: Appending archive to EXE C:\Users\dmam\Desktop\Secure lab temp\build\temp\temp.exe
19714 INFO: Building EXE from EXE-00.toc completed successfully.
19719 INFO: checking COLLECT
19719 INFO: Building COLLECT because COLLECT-00.toc is non-existent
19721 INFO: Building COLLECT COLLECT-00.toc
21058 INFO: Building COLLECT COLLECT-00.toc completed successfully.

C:\Users\dmam\Desktop\Secure lab temp>
```



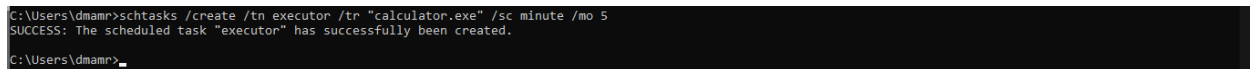
Schedule a task named “Python execution” to run the above executable on the first Monday of every month.

C:\Users\dmamr>schtasks /create /tn Python_Execution /tr "C:\Users\dmamr\Desktop\Secure lab temp\dist\temp\temp.exe" /sc weekly /mo 4 /d MON



Schedule a task named “Executor” on your machine to run calculator for every five minutes starting from the specified start time with no end.

C:\Users\dmamr>schtasks /create /tn executor /tr "calculator.exe" /sc minute /mo 5



Schedule a task named "Executor2" to run notepad starting at 5:00PM and automatically terminating at 5:40PM hours every day.

C:\Users\dmamr>schtasks /create /tn Executor2 /tr "notepad.exe" /sc daily /st 17:00 /et 17:40

```
Command Prompt
Microsoft Windows [Version 10.0.19042.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\dmamr>schtasks /create /tn Python_Execution /tr "C:\Users\dmamr\Desktop\Secure lab temp\dist\temp\temp.exe" /sc weekly /mo 4 /d MON
SUCCESS: The scheduled task "Python_Execution" has successfully been created.

C:\Users\dmamr>schtasks /create /tn executor /tr "calculator.exe" /sc minute /mo 5
SUCCESS: The scheduled task "executor" has successfully been created.

C:\Users\dmamr>schtasks /create /tn Executor2 /tr "notepad.exe" /sc daily /st 17:00 /et 17:40
SUCCESS: The scheduled task "Executor2" has successfully been created.

C:\Users\dmamr>
```

Schedule a task to defragment any of your local disk daily at 10AM

C:\Users\dmamr>schtasks /create /tn Executor3 /tr "defrag.exe C:" /sc daily /st 10:00

```
Select Command Prompt
Microsoft Windows [Version 10.0.19042.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\dmamr>schtasks /create /tn Executor3 /tr "defrag.exe C:" /sc daily /st 10:00
SUCCESS: The scheduled task "Executor3" has successfully been created.

C:\Users\dmamr>
```

PowerCFG:

```
Select Command Prompt
Microsoft Windows [Version 10.0.19042.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\dmamr>powercfg /lastwake
Wake History Count - 1
Wake History [0]
Wake Source Count - 0

C:\Users\dmamr>powercfg /a
The following sleep states are available on this system:
    Standby (S3)
    Hibernate
    Fast Startup

The following sleep states are not available on this system:
    Standby (S1)
        The system firmware does not support this standby state.

    Standby (S2)
        The system firmware does not support this standby state.

    Standby (S0 Low Power Idle)
        The system firmware does not support this standby state.

    Hybrid Sleep
        The hypervisor does not support this standby state.

C:\Users\dmamr>
```

Write a script to perform the following jobs:

To lock your windows PC (Windows+L):

```
import ctypes  
ctypes.windll.user32.LockWorkStation()
```

To clear your Recycle Bin:

```
import winshell  
winshell.recycle_bin().empty(confirm=False, show_progress=False, sound=False)
```