Pipfile

[[source]]  
url = "https://pypi.org/simple"  
verify\_ssl = true  
name = "pypi"  
  
[packages]  
python-docx = "\*"  
termcolor = "\*"  
  
[dev-packages]  
pyinstaller = "\*"  
  
[requires]  
python\_version = "3.10"  
  
[scripts]  
build = "pyinstaller --onefile --distpath . bundle.py"

README.md

# Homework Bundler  
Bundles homework for easy submission to Canvas  
  
## Using the bundler...  
  
Place the bundler executable in the same folder as your homework and run the app!  
  
The \*\*bundle.exe\*\* executable will produce a \*\*bundle.docx\*\* file for submission.  
  
## Building the bundler...  
  
 pipenv install --dev  
 pipenv run build

bundle.py

import os  
import sys  
from glob import glob  
from docx import Document  
from docx.shared import Inches, Pt  
from termcolor import colored  
  
if getattr(sys, 'frozen', False):  
 app\_path = os.path.dirname(sys.executable)  
elif \_\_file\_\_:  
 app\_path = os.path.dirname(\_\_file\_\_)  
os.chdir(app\_path)  
  
types = [  
 '\*\*/\*.md', '\*\*/\*.toml', '\*\*/Pipfile',  
 '\*\*/\*.py', '\*\*/\*.htm', '\*\*/\*.html',  
 '\*\*/\*.csv', '\*\*/\*.json', '\*\*/\*.xml',  
 '\*\*/\*.png', '\*\*/\*.jpg'  
]  
  
files = []  
for t in types: files.extend(glob(t, recursive=True))  
files = sorted(files)  
  
document = Document()  
  
print(colored(f"Bundling files...", 'white', attrs=['bold']))  
for file in files:  
 if 'build' in file or 'dist' in file or '\_\_init\_\_.py' in file: continue  
 if 'manage.py' in file or 'asgi.py' in file or 'wsgi.py' in file: continue  
 print(' - ' + colored(f"{file}", 'green'))  
 h = document.add\_heading(file, 0)  
 h.style.font.size = Pt(18)  
 h.style.font.bold = True  
 if '.png' in file or '.jpg' in file:  
 document.add\_picture(file, width=Inches(6))  
 else:  
 with open(file, 'r', encoding='utf-8', errors='ignore') as f:  
 document.add\_paragraph(f.read()).style.font.size = Pt(9)  
 document.add\_page\_break()  
  
document.save('bundle.docx')

python.jpg

