CREATION OF DEB EXECUTABLE IN LINUX

CREATION OF DEB FILE (PY to DEB)

- It is possible to convert the python file into executable debian package file (deb) in linux
- The package manager **dpkg** is used to convert python file into deb file in the debian based linux OS like Debian, ubuntu, mint, kali, etc,…

REQUIREMENTS

- Python Script (.py)
- Debian Based Linux Distros

DEB (Software Package)

- It stands for **Deb**ian
- It is similar to windows executable files like .exe, .msi
- It is called as **debian software package file** used by the debian linux distribution and its variants like ubuntu, kali, mint, etc,…
- DEB files are mainly used to install or update linux / unix applications
- Each DEB file is a standard unix archive file which contains two .tar files
    1. One for software installer control information (control)
    2. Another for installable data (data).

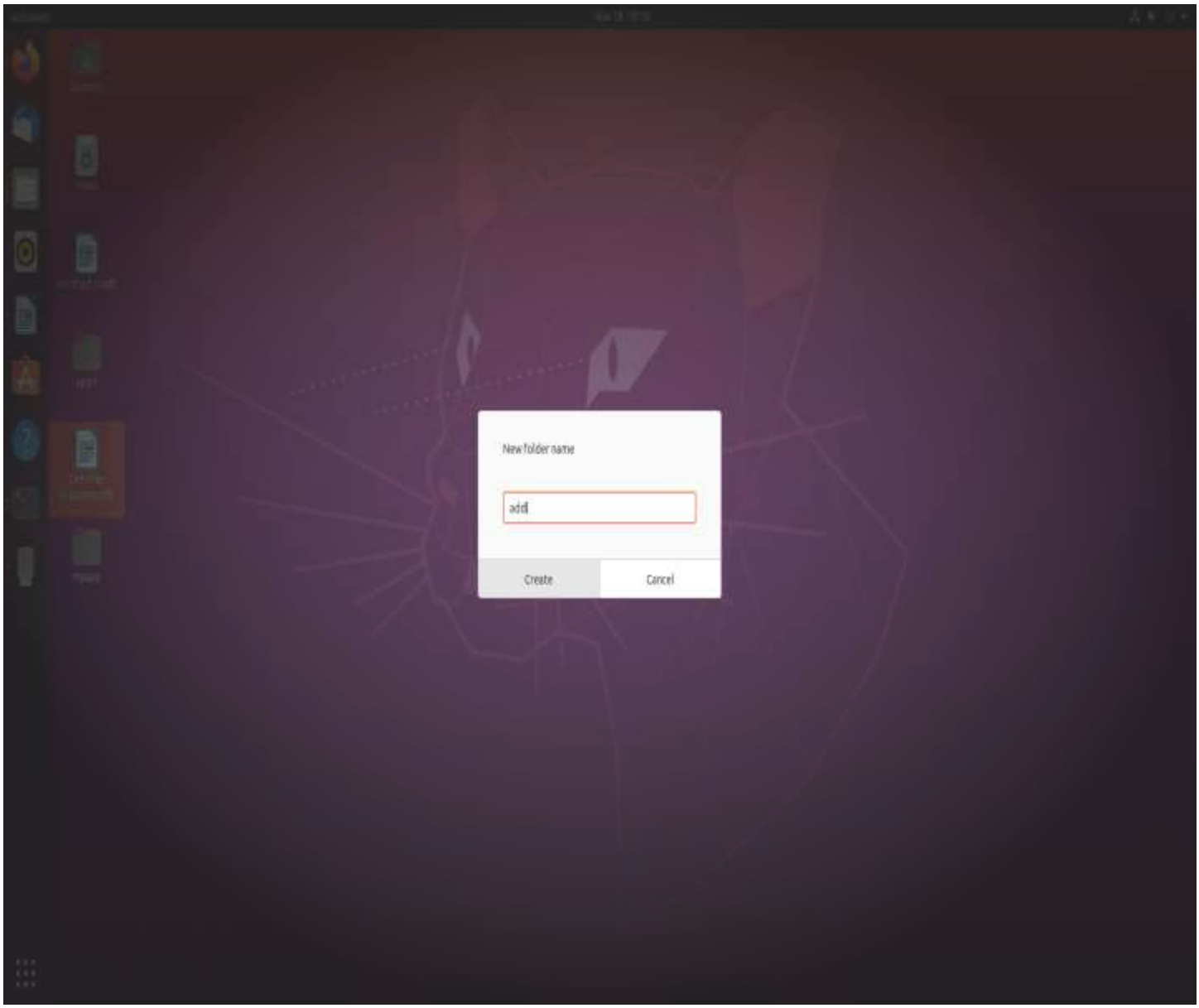
dpkg command tool

- Dpkg stands for **D**ebian **P**ac**k**a**g**e Manager
- It is a command line package manager used in DEB linux
- It is used to install and manipulate debian packages (softwares)

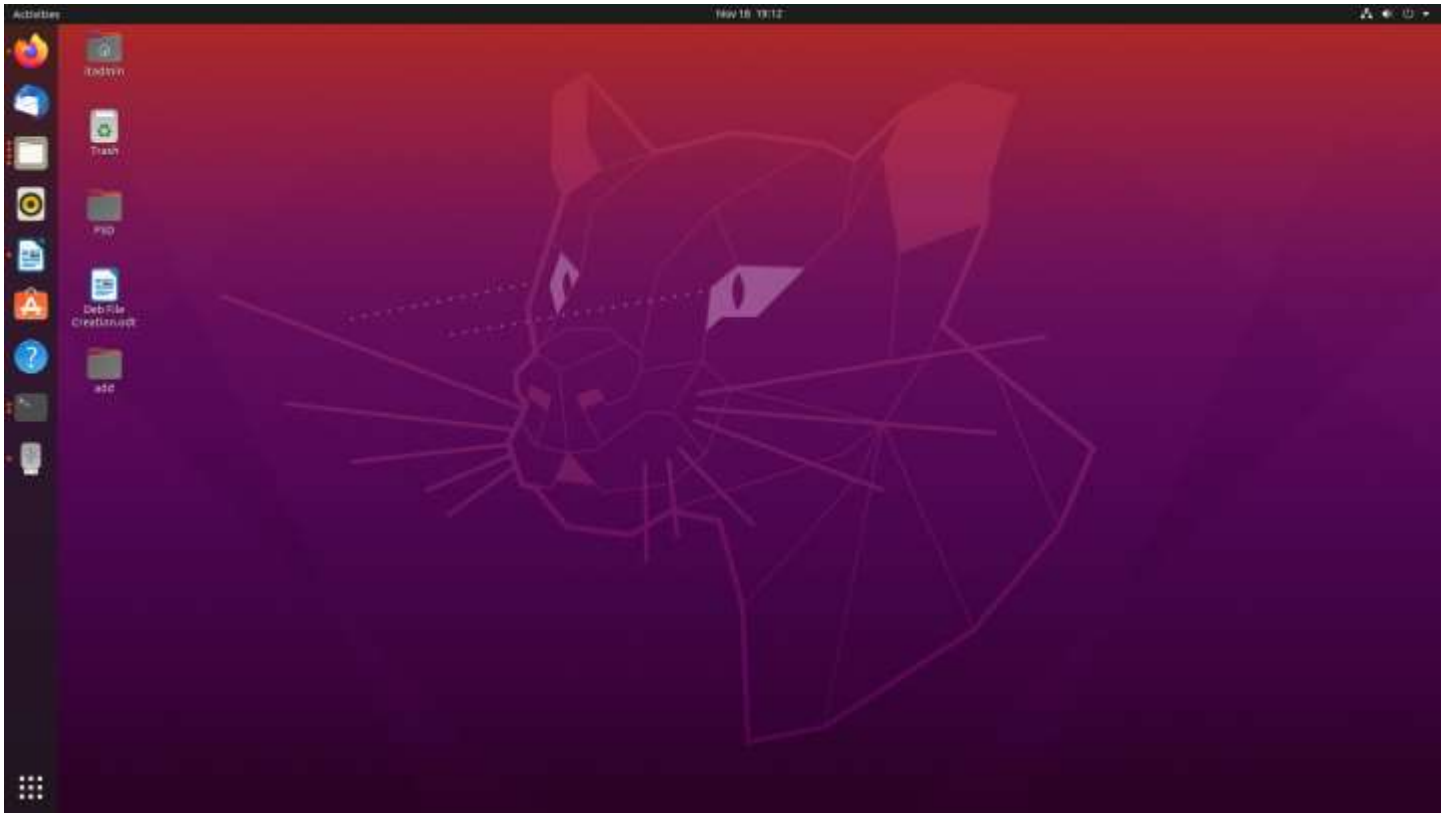
I. CONVERSION OF PYTHON SCRIPT TO DEB FILE

Input File	:	add.py (python-wxpython file)
Tested OS	:	Ubuntu Linux

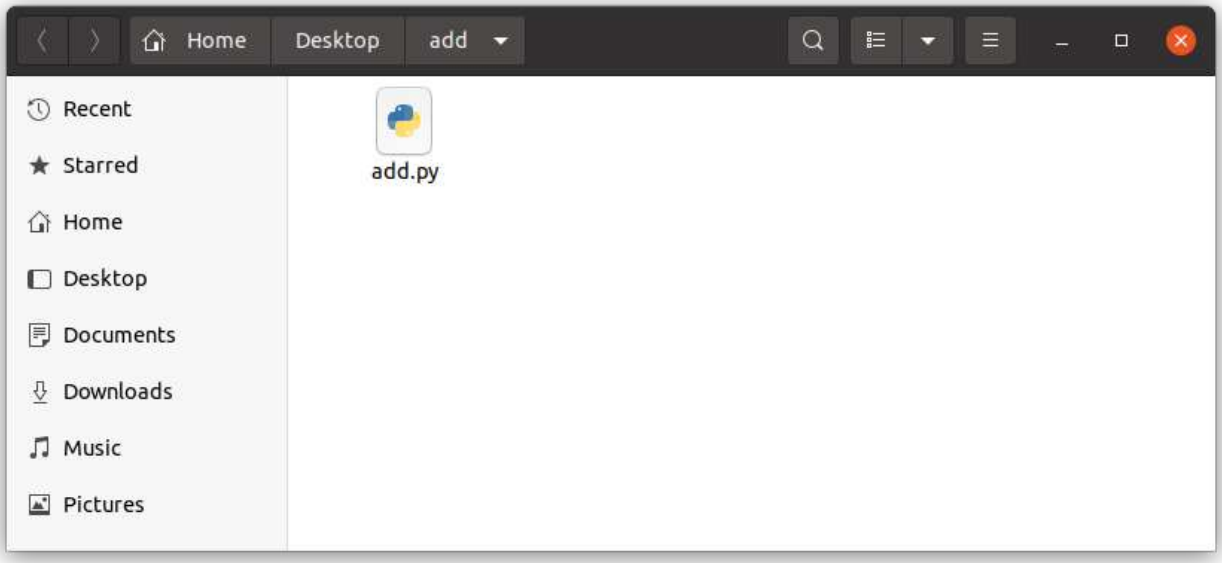
Step 1: Create a Folder on Desktop in Ubuntu OS



Folder Verification in Desktop

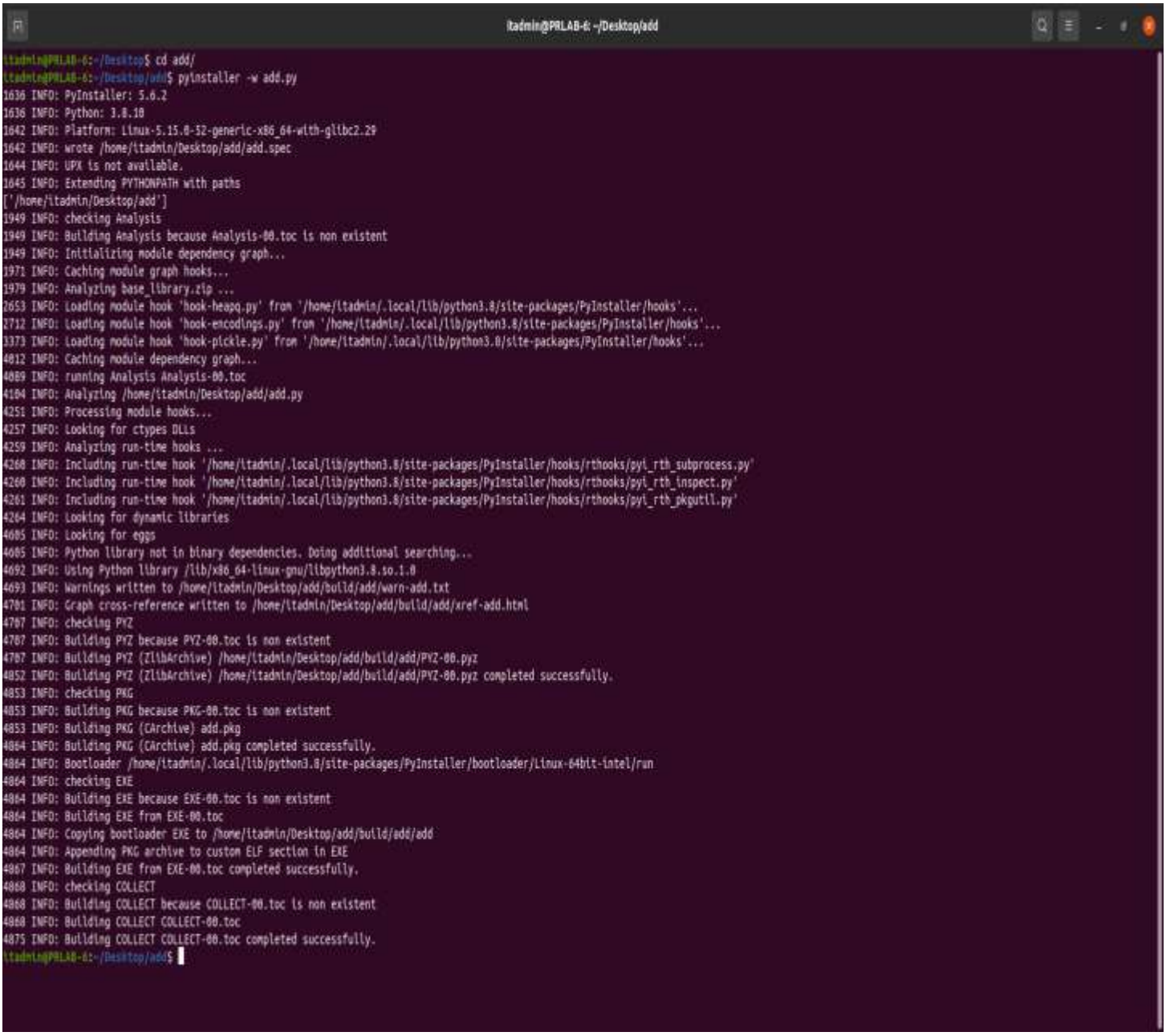


Contents of add directory



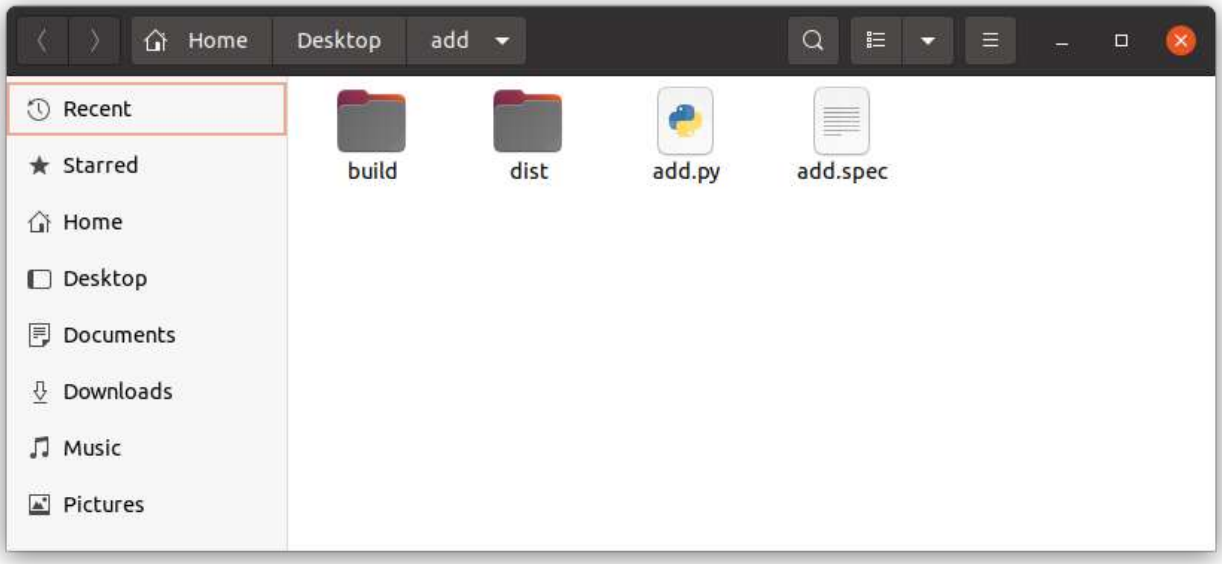
Step 2: Navigate to the directory add and run the following command in terminal

```
pyinstaller -w filename.py
pyinstaller -w add.py
```



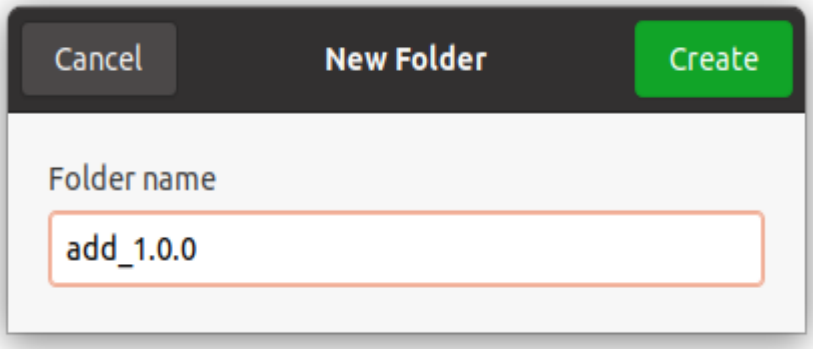
- Executable (.exe) file will be created in the **dist folder** after the execution of the above command

Contents of directory **add** after the execution of above command.

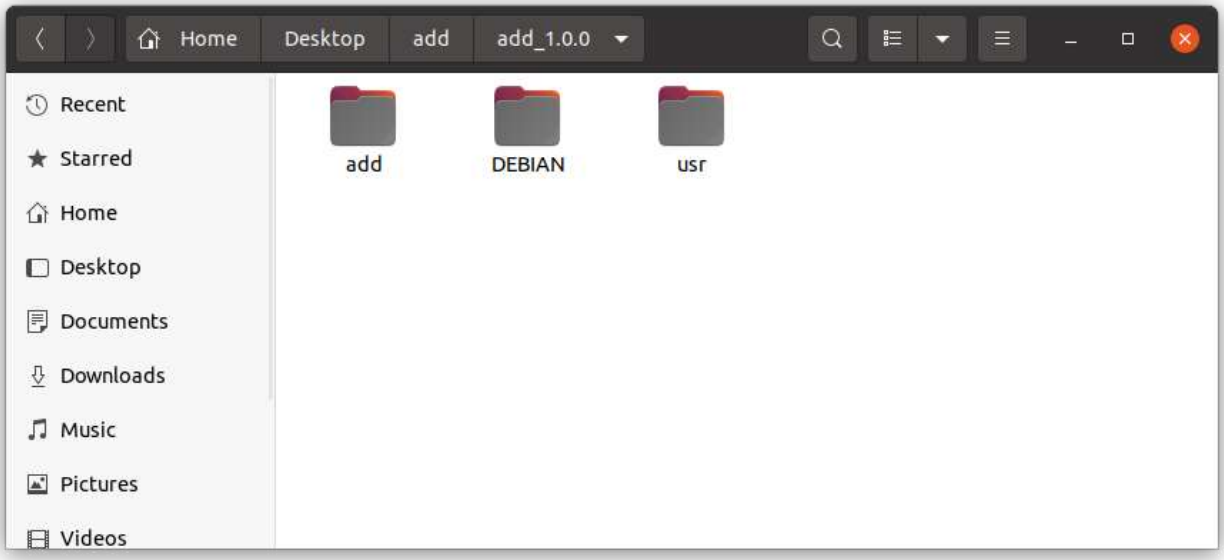


Step 3: Create the following directories in the current directory "add"

- **add_1.0.0/DEBIAN**
- **add_1.0.0/add**
- **add_1.0.0/usr/share/applications**



Contents of directory add_1.0.0



Step 4: Create a new text file add_1.0.0/DEBIAN/control and add the following contents into it

Syntax

```
Package: my-app
Version: 1.0.0
Architecture: all
Maintainer: [Your name]
Copyright: [year] [Your name]
License: MIT
Homepage: [homepage url]
Description: My deb package.
```

Example: control text file

Package: **add**
Version: **1.0.0**
Architecture: all
Maintainer: MIT
Copyright: 2022@mit
License: MIT
Homepage: https://www.mitindia.edu
Description: This is the creation of deb in debian based OS...

Screenshot

Step 5: Create a new file add_1.0.0/usr/share/applications/add.desktop and include the following contents into it

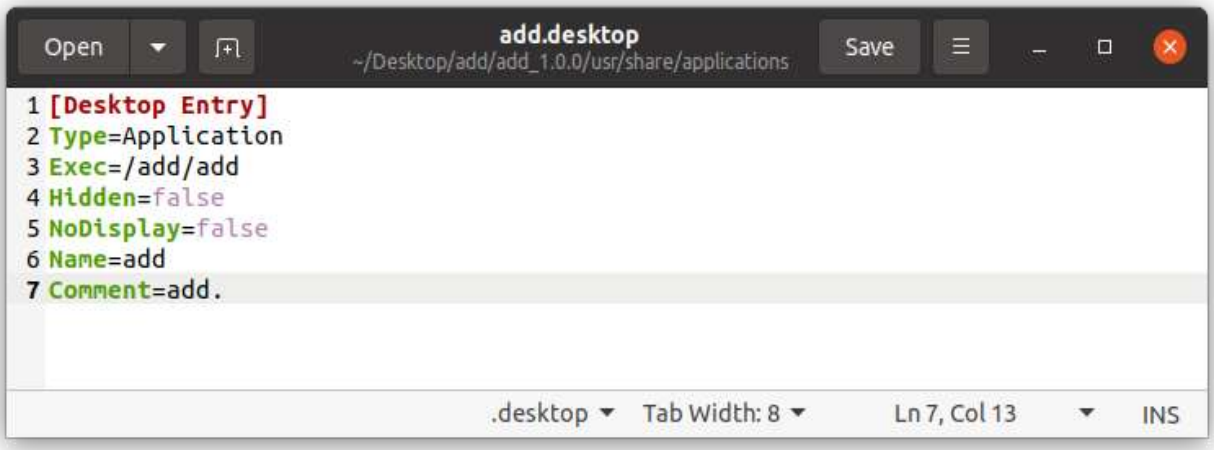
Syntax

```
[Desktop Entry]
Type=Application
Exec=Folder Path
Hidden=false
NoDisplay=false
Name=App Name
Comment=Description about application / software.
```

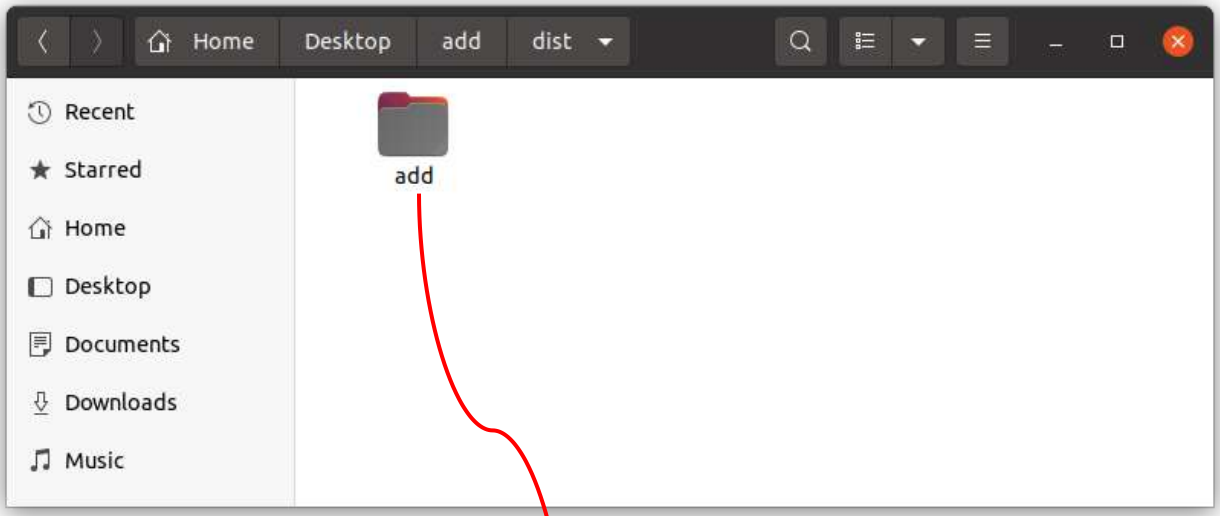
Example

```
[Desktop Entry]
Type=Application
Exec=/add/add
Hidden=false
NoDisplay=false
Name=add
Comment=add.
```

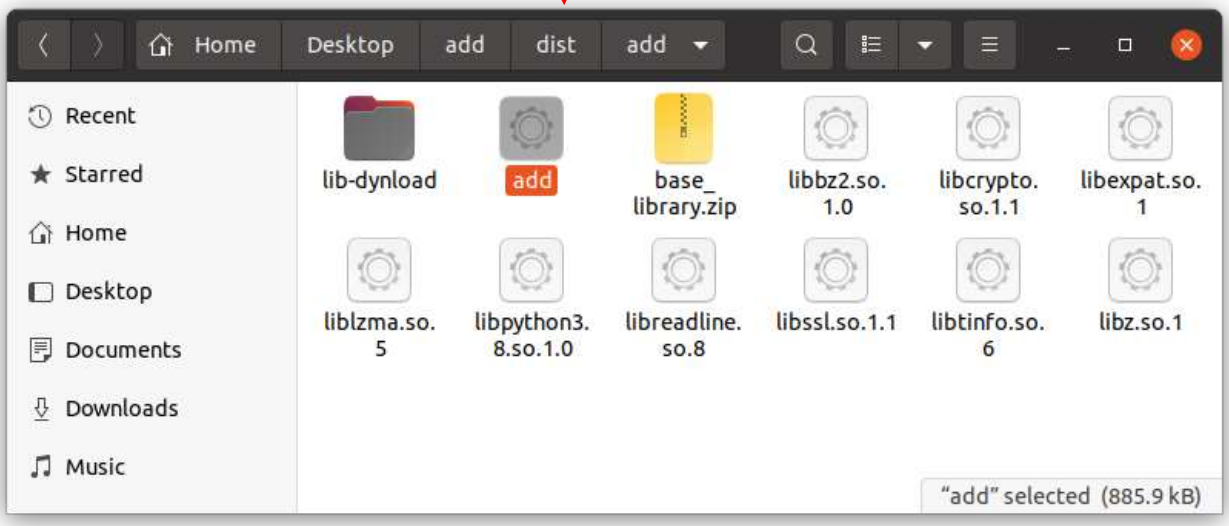
Screenshot



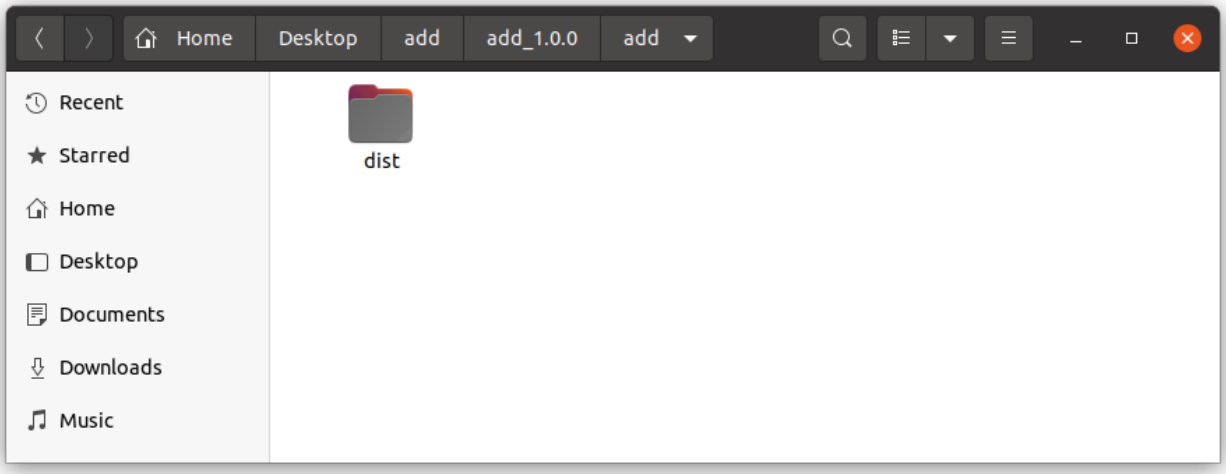
Contents of **dist** folder (add/dist)



dist/add/



Step 6: Copy all files and directories from add/dist to add_1.0.0/add



Step 7: Navigate to the parent directory of add_1.0.0 and enter the following command like below

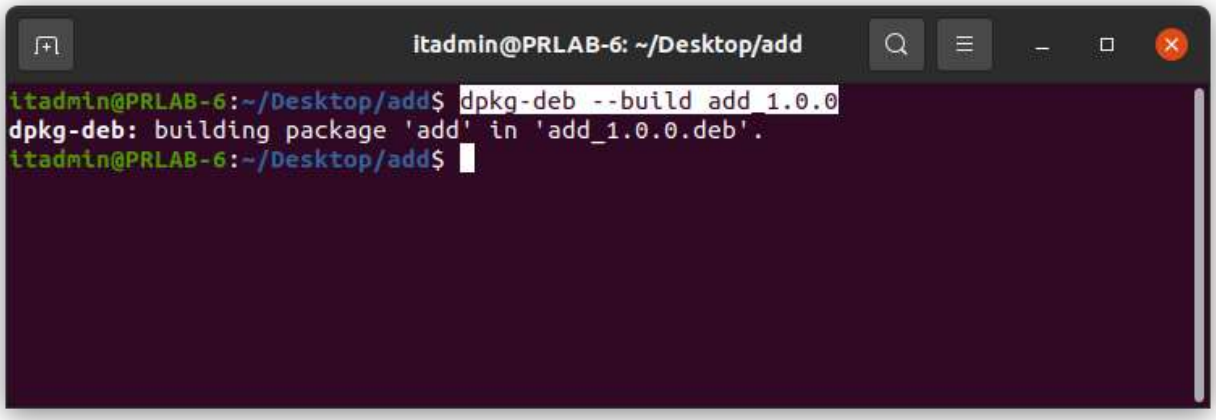
Syntax

```
dpkg-deb --build <application folder>
```

Example

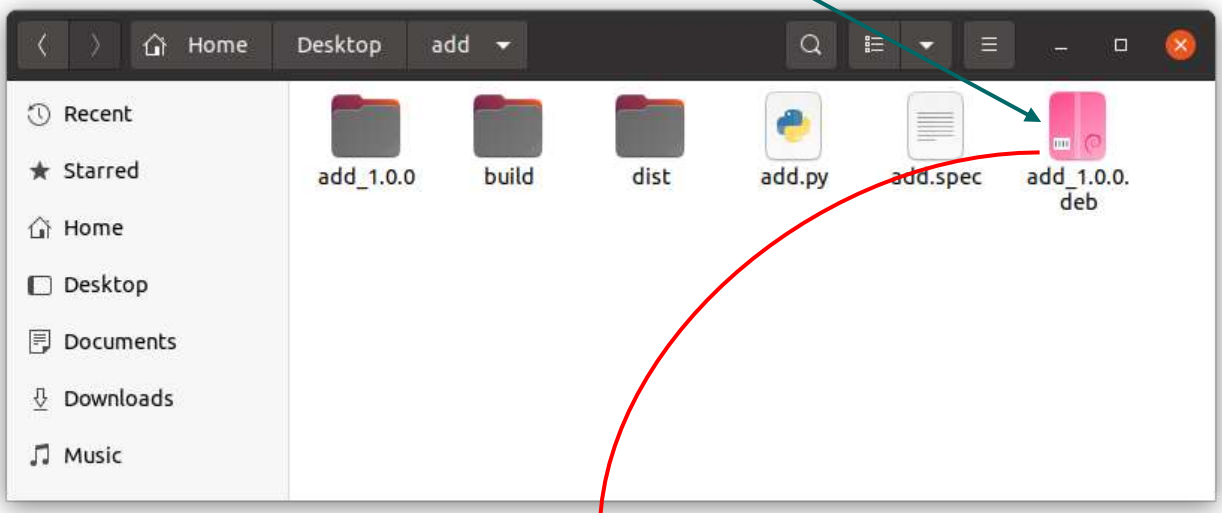
```
dpkg-deb --build add_1.0.0
```

Screenshot



Step 8: Wait for it to complete and check the .deb package named add_1.0.0.deb in the directory input add

VERIFICATION OF NEWLY CREATED .DEB EXECUTABLE FILE IN THE DIRECTORY ADD



PROPERTIES OF NEWLY CREATED .DEB EXECUTABLE FILE