

Exercises for tar, gzip, bzip2, zip, xz, cpio, dd

1. Tar Archiving and Extraction

- Create an archive of your home directory's Documents folder:

```
tar -cvf documents_backup.tar ~/docs
```

- List contents of the tar archive:

```
tar -tvf documents_backup.tar
```

- Extract the archive to /tmp/restore after creating the directory /tmp/restore.

```
mkdir /tmp/restore
```

```
tar -xvf documents_backup.tar -C /tmp/restore
```

2. gzip Compression and Decompression

- Compress a log file:
- What is the size of biglog.log?

```
gzip biglog.log
```

- What is the size of biglog.log.gz?
- Decompress it back:

```
gunzip biglog.log.gz
```

- Task: Use tar with gzip combined options to create a compressed archive of /etc in one command.
- Hint: use `tar zcvf ...`

3. bzip2 File Compression

- Compress a file using bzip2:

```
bzip2 sample.txt
```

- Decompress using:

```
bunzip2 sample.txt.bz2
```

- Exercise: Compare the compression ratio of gzip vs bzip2 on the same file with `ls -lh`.

4. zip and unzip for Archiving Multiple Files

- Create a zip archive with some text files:

```
zip textfiles.zip text1.txt text2.txt
```

- Extract to a chosen directory:

```
unzip textfiles.zip -d /tmp
```

- Challenge: Exclude files with `.bak` extension from being zipped by using a zip option or file pattern.

5. xz Compression for Maximum Compression Ratio

- Compress a file:

```
xz biglog.log
```

- Decompress:

```
unxz biglog.log.xz
```

- Question: Which xz option allows keeping the original file after compression?

6. cpio Usage for Archiving with Find

- Create an archive from files found in the current directory:

```
find . -type f | cpio -o > archive.cpio
```

- Extract with:

```
cpio -idv < archive.cpio
```

- Task: Pipe the output through gzip to create a compressed cpio archive.

Hint: use the following construct: `find | cpio | gzip > /tmp/archive.cpio`

7. dd for Creating Files and Device Imaging

- Create an empty 10 MB file filled with zeros:

```
dd if=/dev/zero of=emptyfile.img bs=1M count=10
```

- Copy an ISO image to a USB device (example):

```
dd if=/home/rocky/mini.iso of=/dev/sda bs=4M status=progress
```

- Exercise: What is the purpose of the `status=progress` option?