Here is a **detailed sentence-by-sentence study note breakdown** of the document **“130. File Management Commands Notes.docx”**, tailored for CompTIA A+ 220-1102 Objective 1.8. This guide focuses on **file management in Linux using commands like mv, cp, rm, and wildcards**, with detailed examples and safety practices.

**🧠 Study Notes – Linux File Management (mv, cp, rm)**

**📁 mv – Move (or Rename) Files**

* The mv command moves files from one location to another or renames them.
* Basic syntax:
  + mv [options] source destination
  + Options are optional and shown in brackets [].
* To learn about mv, use:
  + mv --help
* **Example scenario**:
  + Files: bad.txt, bed.txt, beg.txt, cat.txt, fad.txt
  + To move cat.txt into a subfolder called cats, first:
  + mkdir cats
  + mv cat.txt cats
* To **rename a file** using mv:
* mv big\_cat.txt cat.txt
* To move and rename at the same time:
* mv cats/cat.txt ../big\_cat.txt

**🏷️ Wildcards in Linux**

* \* (asterisk) = matches **any number of characters**
  + ls b\* → matches: bad.txt, bed.txt, beg.txt
* ? (question mark) = matches **a single character**
  + ls be?.txt → matches: bed.txt, beg.txt (but not beet.txt)

**📂 Using mv with Wildcards**

* Example: Move all files starting with **b**:
* mv b\*.txt bword/
* Always check what files will be matched first:
* ls b\*.txt
* Avoid mistakes: bword folder also matches b\* — could move the folder itself!

**📎 Recursive Moves**

* Move **everything** from one folder up a level:
* mv \*.\* ../
* Clean up afterward:
* rmdir foldername

**📝 cp – Copy Files**

* cp duplicates files from source to destination.
* Basic syntax:
* cp source destination
* Difference between mv and cp:
  + mv moves (original is gone)
  + cp copies (original stays)
* Examples:
  + Copy and rename:
  + cp cats.txt cat.txt
  + Copy into a folder:
  + cp cats.txt cats/cat.txt

**🗑️ rm – Remove Files**

* rm filename deletes a file.
  + Example:
  + rm cats/cats.txt
* rmdir folder deletes an **empty folder**.
* For folders with files:
* rm -rf foldername
  + -r = recursive (removes all inside)
  + -f = force (no confirmation prompts)
* Even if a folder looks empty, ls -la may show hidden files like . and ...
  + Use rm -rf to handle these cases.

**🛠️ Final Summary – Key Commands & Safety Tips**

| **Command** | **Function** | **Example** |
| --- | --- | --- |
| mv | Move or rename files | mv file1 folder/, mv old.txt new.txt |
| cp | Copy files | cp file1 file2 |
| rm | Delete files | rm file1.txt |
| rmdir | Remove empty directories | rmdir emptyfolder |
| rm -rf | Remove directories with contents | rm -rf folder |

**✅ Best Practices Before Using Destructive Commands**

* Use ls to **preview wildcard matches**:
  + ls b\* → make sure it lists only what you intend to affect.
* Avoid accidental data loss:
  + Always double-check wildcard usage like \*, ?
* Rename, move, or delete **cautiously** — especially in bulk.

**🎯 CompTIA A+ 220-1102 Relevance**

**Exam Objective 1.8** expects you to:

* Use Linux command-line tools (mv, cp, rm)
* Understand file/directory navigation and manipulation
* Recognize and properly use **wildcards** and options like -r, -f
* Apply safety practices before executing commands that alter file structures

Would you like a printable **Linux command cheat sheet** or a **wildcard practice quiz**? I can generate one based on this lesson!