Using Wildcards and Brace Expansion

1. Asterisk Wildcard (*)

Definition: The asterisk (*) wildcard matches any number of characters, including no characters, in a file or directory name.

Examples:

- List all files in a directory regardless of name or extension:

```
ls *
```

- List all hidden files in a directory(excluding the parent directory):

```
ls .??*
```

- list files that start with "pizza" and end in any extension:

```
ls pizza*
```

2. Question Mark Wildcard (?)

Definition: The question mark (?) wildcard matches exactly one character in a filename or directory name.

Examples:

- List files with names that are exactly 4 characters long and have a `.exe` extension:

```
ls ????.exe
```

- List files that start with "a" followed by a single character and end in `.log`:

```
ls a?.log
```

- List files with extensions that are exactly 3 characters long:

```
ls *.???
```

3. Bracket Wildcard ([])

Definition: The bracket ([]) wildcard matches any one of the characters within the brackets in the file or directory name. You can also specify ranges within the brackets.

Examples:

- List files that contain "a", "b" or any number and end in `.txt`:

```
ls *[ab0-9]*.txt
```

- List files that start with "a" and cannot contain numbers before the extension:

```
ls a*[!0-9].*
```

- List files that cannot start with uppercase letters and end in `.sh`:

```
ls [!A-Z]*.sh
```

- List files that contain punctuation sign:

```
ls *[[:punct:]]*.sh
```

4. Brace Expansion ({})

Definition: Brace expansion allows you to generate a series of names by specifying comma-separated values or ranges within braces ({}). It's commonly used to create multiple files or directories at once.

Examples:

- Create multiple files with different extensions in one command:

```
touch file.{txt,doc,pdf}
```

- Generate a sequence of numbered directories:

```
mkdir dir_{1..3}
```

- List files with names that start with either "fileA", "fileB", or "fileC":

```
ls file{A,B,C}.txt
```

Table Wildcards

Wildcard	Definition	Example
*	matches 0 to any number of characters	ls ~/Downloads/*.png
?	matches 1 character	ls ~/Downloads/f?ll.sh
[]	matches 1 character from a set	ls ~/Downloads/f[0-9]ll.sh