Important Shell Flags

Write down important flags with meaning and commands with example usages in shell scripting

Here’s a list of important flags with their meanings and example usages in shell scripting:

---

1. File Test Operators (Flags)

Flag	Meaning	Example
`-e`	Check if a file exists	`if [ -e file.txt ]; then echo "Exists"; fi`
`-f`	Check if it's a regular file	`if [ -f file.txt ]; then echo "File exists"; fi`
`-d`	Check if it's a directory	`if [ -d mydir ]; then echo "Directory exists"; fi`
`-r`	Check if the file is readable	`if [ -r file.txt ]; then echo "Readable"; fi`
`-w`	Check if the file is writable	`if [ -w file.txt ]; then echo "Writable"; fi`
`-x`	Check if the file is executable	`if [ -x script.sh ]; then echo "Executable"; fi`
`-s`	Check if the file is non-empty	`if [ -s file.txt ]; then echo "Not empty"; fi`
`-L`	Check if it's a symbolic link	`if [ -L link ]; then echo "It's a symlink"; fi`

---

2. String Test Operators

Flag	Meaning	Example
`-z`	Check if string is empty	`if [ -z "$var" ]; then echo "Empty"; fi`
`-n`	Check if string is non-empty	`if [ -n "$var" ]; then echo "Non-empty"; fi`
`=`	String equality	`if [ "$str1" = "$str2" ]; then echo "Equal"; fi`
`!=`	String inequality	`if [ "$str1" != "$str2" ]; then echo "Not Equal"; fi`

3. Integer Comparison Operators

Operator	Meaning	Example
`-eq`	Equal to	`if [ "$num1" -eq "$num2" ]; then echo "Equal"; fi`
`-ne`	Not equal to	`if [ "$num1" -ne "$num2" ]; then echo "Not Equal"; fi`
`-lt`	Less than	`if [ "$num1" -lt "$num2" ]; then echo "Less"; fi`
`-le`	Less than or equal to	`if [ "$num1" -le "$num2" ]; then echo "Less or Equal"; fi`
`-gt`	Greater than	`if [ "$num1" -gt "$num2" ]; then echo "Greater"; fi`
`-ge`	Greater than or equal to	`if [ "$num1" -ge "$num2" ]; then echo "Greater or Equal"; fi`

4. Important Shell Commands

1. Echo

Prints a message to the console.

```sh
echo "Hello, World!"
```

2. Read

Takes user input.

```sh
echo "Enter your name:"
read name
echo "Hello, $name!"
```

3. Variables

Assign and use variables.

```sh
name="Alice"
echo "My name is $name"
```

4. Conditional Statements

If-Else

```sh
num=10
if [ $num -gt 5 ]; then
    echo "Number is greater than 5"
else
    echo "Number is less than or equal to 5"
fi
```

Case Statement

```sh
echo "Enter a choice: start/stop"
read action
case $action in
    start) echo "Starting..." ;;
    stop) echo "Stopping..." ;;
    *) echo "Invalid choice" ;;
esac
```

5. Loops

For Loop

```sh
for i in 1 2 3 4 5; do
    echo "Number: $i"
done
```

While Loop

```sh
count=1
while [ $count -le 5 ]; do
    echo "Count: $count"
    ((count++))
done
```

6. Functions

```sh
greet() {
    echo "Hello, $1!"
}
greet "Alice"
```

7. Command Substitution

```sh
today=$(date)
echo "Today's date is $today"
```

---

5. Process Management

Command	Meaning	Example
`ps`	Show running processes	`ps aux`
`kill`	Terminate a process	`kill 1234`
`bg`	Resume a job in the background	`bg %1`
`fg`	Bring a background job to the foreground	`fg %1`
`jobs`	List background jobs	`jobs`

---

6. File Management

Command	Meaning	Example
`ls`	List files	`ls -l`
`cp`	Copy file	`cp file1 file2`
`mv`	Move/Rename file	`mv oldname newname`
`rm`	Remove file	`rm file.txt`
`mkdir`	Create directory	`mkdir newdir`
`rmdir`	Remove empty directory	`rmdir olddir`

---

7. Redirections & Pipes

Symbol	Meaning	Example
`>`	Redirect output to a file	`echo "Hello" > file.txt`
`>>`	Append output to a file	`echo "World" >> file.txt`
`<`	Input redirection	`wc -l < file.txt`
`	`	Pipe output

---

8. Permissions & Ownership

Command	Meaning	Example
`chmod`	Change permissions	`chmod 755 script.sh`
`chown`	Change file owner	`chown user:group file.txt`
`umask`	Set default permissions	`umask 022`

---

9. Environment Variables

Command	Meaning	Example
`export`	Set environment variable	`export PATH=$PATH:/new/path`
`env`	Show all environment variables	`env`
`unset`	Remove an environment variable	`unset VAR_NAME`

---

10. Networking Commands

Command	Meaning	Example
`ping`	Check network connectivity	`ping google.com`
`curl`	Fetch a web page	`curl https://example.com`
`wget`	Download file	`wget http://example.com/file.zip`
`netstat`	Show network status	`netstat -tulnp`

This list covers most commonly used shell scripting flags, commands, and their practical examples. Let me know if you need further details! 🚀