Let us first create test.txt file that will be used in all the examples mentioned below.

$cat test.txt

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

Append Lines Using Sed Command

Sed provides the command “a” which appends a line after every line with the address or pattern.

Syntax:

#sed 'ADDRESS a\

Line which you want to append' filename

#sed '/PATTERN/ a\

Line which you want to append' filename

Sed Append Example 1. Add a line after the 3rd line of the file.

Add the line “Cool gadgets and websites” after the 3rd line. sed “a” command inserts the line after match.

$ **sed '3 a\**

**> Cool gadgets and websites' test.txt**

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

**Cool gadgets and websites**

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

Sed Append Example 2. Append a line after every line matching the pattern

The below sed command will add the line “Linux Scripting” after every line that matches the pattern “Sysadmin”.

**$ sed '/Sysadmin/a \**

**> Linux Scripting' test.txt**

Linux Sysadmin

**Linux Scripting**

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

**Linux Scripting**

Sed Append Example 3. Append a line at the end of the file

The following example, appends the line “Website Design” at the end of the file.

**$ sed '$ a\**

**> Website Design' test.txt**

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

**Website Design**

Insert Lines Using Sed Command

Sed command “i” is used to insert a line before every line with the range or pattern.

Syntax:

#sed 'ADDRESS i\

Line which you want to insert' filename

#sed '/PATTERN/ i\

Line which you want to insert' filename

Sed Insert Example 1. Add a line before the 4th line of the line.

Add a line “Cool gadgets and websites” before 4th line. “a” command inserts the line after match whereas “i” inserts before match.

**$ sed '4 i\**

**> Cool gadgets and websites' test.txt**

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

**Cool gadgets and websites**

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

Sed Insert Example 2. Insert a line before every line with the pattern

The below sed command will add a line “Linux Scripting” before every line that matches with the pattern called ‘Sysadmin”.

**$ sed '/Sysadmin/i \**

**> Linux Scripting' test.txt**

**Linux Scripting**

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

**Linux Scripting**

Windows- Sysadmin, reboot etc.

Sed Insert Example 3. Insert a line before the last line of the file.

Append a line “Website Design” before the last line of the file.

**$ sed '$ i\**

**> Website Design' test.txt**

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

**Website Design**

Windows- Sysadmin, reboot etc.

Replace Lines Using Sed Command

“c” command in sed is used to replace every line matches with the pattern or ranges with the new given line.

Syntax:

#sed 'ADDRESS c\

new line' filename

#sed '/PATTERN/ c\

new line' filename

Sed Replace Example 1. Replace a first line of the file

The below command replaces the first line of the file with the “The Geek Stuff”.

**$ sed '1 c\**

**> The Geek Stuff' test.txt**

**The Geek Stuff**

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

Sed Replace Example 2. Replace a line which matches the pattern

Replace everyline which has a pattern “Linux Sysadmin” to “Linux Sysadmin – Scripting”.

**$ sed '/Linux Sysadmin/c \**

**> Linux Sysadmin - Scripting' test.txt**

**Linux Sysadmin - Scripting**

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

Windows- Sysadmin, reboot etc.

Sed Replace Example 3. Replace the last line of the file

Sed command given below replaces the last line of the file with “Last Line of the file”.

**$ sed '$ c\**

**> Last line of the file' test.txt**

Linux Sysadmin

Databases - Oracle, mySQL etc.

Security (Firewall, Network, Online Security etc)

Storage in Linux

Productivity (Too many technologies to explore, not much time available)

**Last line of the file**

Print Line Numbers Using Sed Command

“=” is a command in sed to print the current line number to the standard output.

Syntax:

#sed '=' filename

The above send command syntax prints line number in the first line and the original line from the file in the next line .

sed ‘=’ command accepts only one address, so if you want to print line number for a range of lines, you must use the curly braces.

Syntax:

# sed -n '/PATTERN/,/PATTERN/ {

=

p

}' filename

Sed Line Number Example 1. Find the line number which contains the pattern

The below sed command prints the line number for which matches with the pattern “Databases”

**$ sed -n '/Databases/=' test.txt**

2

Sed Line Number Example 2. Printing Range of line numbers

Print the line numbers for the lines matches from the pattern “Oracle” to “Productivity”.

**$ sed -n '/Oracle/,/Productivity/{**

**> =**

**> p**

**> }' test.txt**

2

Databases - Oracle, mySQL etc.

3

Security (Firewall, Network, Online Security etc)

4

Storage in Linux

5

Productivity (Too many technologies to explore, not much time available)

Sed Line Number Example 3. Print the total number of lines in a file

Line number of the last line of the file will be the total lines in a file. Pattern $ specifies the last line of the file.

**$ sed -n '$=' test.txt**

6

**Deleting lines from a particular file :** SED command can also be used for deleting lines from a particular file. SED command is used for performing deletion operation without even opening the file

Examples:  
**1. To Delete a particular line say n in this example**

1. Syntax:
2. $ sed 'nd' filename.txt
3. Example:
4. $ sed '5d' filename.txt

**2. To Delete a last line**

Syntax:

$ sed '$d' filename.txt

**3. To Delete line from range x to y**

Syntax:

$ sed 'x,yd' filename.txt

Example:

$ sed '3,6d' filename.txt

1. **To Delete from nth to last line**

Syntax:

$ sed 'nth,$d' filename.txt

Example:

$ sed '12,$d' filename.txt

**6. To Delete pattern matching line**

Syntax:

$ sed '/pattern/d' filename.txt

Example:

$ sed '/abc/d' filename.txt

**To save changes**

**Sed –i or use redirection**

**Like**

**Sed –i ‘/abc/d’ test.txt**

**How to Declare Array in Shell Scripting?**  
We can declare an array in a shell script in different ways.

1. Indirect Declaration  
In Indirect declaration, We assigned a value in a particular index of Array Variable. No need to first declare.

ARRAYNAME[INDEXNR]=value

2. Explicit Declaration  
In Explicit Declaration, First We declare array then assigned the values.

declare -a ARRAYNAME

3. Compound Assignment  
In Compount Assignment, We declare array with a bunch of values. We can add other values later too.

ARRAYNAME=(value1 value2 .... valueN)

or  
[indexnumber=]string

ARRAYNAME=([1]=10 [2]=20 [3]=30)

**To Print Array Value in Shell Script?**

To Print All elements

[@] & [\*] means All elements of Array.

echo ${ARRAYNAME[\*]}

|  |
| --- |
| #! /bin/bash    # To declare static Array  arr=(prakhar ankit 1 rishabh manish abhinav)    # To print all elements of array  echo ${arr[@]}  echo ${arr[\*]}  echo ${arr[@]:0}  echo ${arr[\*]:0} |

Output:

prakhar ankit 1 rishabh manish abhinav

prakhar ankit 1 rishabh manish abhinav

prakhar ankit 1 rishabh manish abhinav

prakhar ankit 1 rishabh manish abhinav

To Print first element

|  |
| --- |
| # To print first element  echo ${arr[0]}  echo ${arr} |

Output:

prakhar

prakhar

To Print Selected index element

echo ${ARRAYNAME[INDEXNR]}

|  |
| --- |
| # To print particular element  echo ${arr[3]}  echo ${arr[1]} |

Output:

rishabh

ankit

To print elements from a particular index

echo ${ARRAYNAME[WHICH\_ELEMENT]:STARTING\_INDEX}

|  |
| --- |
| # To print elements from a particular index  echo ${arr[@]:0}  echo ${arr[@]:1}  echo ${arr[@]:2}  echo ${arr[0]:1} |

Output:

prakhar ankit 1 rishabh manish abhinav

ankit 1 rishabh manish abhinav

1 rishabh manish abhinav

prakhar

To print elements in range

echo ${ARRAYNAME[WHICH\_ELEMENT]:STARTING\_INDEX:COUNT\_ELEMENT}

|  |
| --- |
| # To print elements in range  echo ${arr[@]:1:4}  echo ${arr[@]:2:3}  echo ${arr[0]:1:3} |

Output:

ankit 1 rishabh manish

1 rishabh manish

rak

**To count Length of in Array**

To count the length of a particular element in Array.  
Use #(hash) to print length of particular element

|  |
| --- |
| # Length of Particular element  echo ${#arr[0]}  echo ${#arr} |

Output:

7

7

To count length of Array.

|  |
| --- |
| # Size of an Array  echo ${#arr[@]}  echo ${#arr[\*]} |

Output:

6

6

**To Search in Array**

rr[@] : All Array Elements.  
/Search\_using\_Regular\_Expression/ : Search in Array

Search Returns 1 if it found the pattern else it return zero. It does not alter the original array elements.

|  |
| --- |
| # Search in Array  echo ${arr[@]/\*[aA]\*/} |

Output:

1

**To Search & Replace in Array**

//Search\_using\_Regular\_Expression/Replace : Search & Replace

Search & Replace does not change in Original Value of Array Element. It just returned the new value. So you can store this value in same or different variable.

|  |
| --- |
| # Replacing Substring Temporary  echo ${arr[@]//a/A}  echo ${arr[@]}  echo ${arr[0]//r/R} |

Output:

prAkhAr Ankit 1 rishAbh mAnish AbhinAv

prakhar ankit 1 rishabh manish abhinav

RakhaR

**To delete Array Variable in Shell Script?**

To delete index-1 element

unset ARRAYNAME[1]

To delete the whole Array

unset ARRAYNAME

|  |
| --- |
| #! /bin/bash  # To declare static Array  arr=(prakhar ankit 1 rishabh manish abhinav)    # To print all elements of array  echo ${arr[@]}        # prakhar ankit 1 rishabh manish abhinav  echo ${arr[\*]}        # prakhar ankit 1 rishabh manish abhinav  echo ${arr[@]:0}    # prakhar ankit 1 rishabh manish abhinav  echo ${arr[\*]:0}    # prakhar ankit 1 rishabh manish abhinav    # To print first element  echo ${arr[0]}        # prakhar  echo ${arr}            # prakhar    # To print particular element  echo ${arr[3]}        # rishabh  echo ${arr[1]}        # ankit    # To print elements from a particular index  echo ${arr[@]:0}    # prakhar ankit 1 rishabh manish abhinav  echo ${arr[@]:1}    # ankit 1 rishabh manish abhinav  echo ${arr[@]:2}    # 1 rishabh manish abhinav  echo ${arr[0]:1}    # rakhar    # To print elements in range  echo ${arr[@]:1:4}    # ankit 1 rishabh manish  echo ${arr[@]:2:3}    # 1 rishabh manish  echo ${arr[0]:1:3}    # rak    # Length of Particular element  echo ${#arr[0]}        # 7  echo ${#arr}        # 7    # Size of an Array  echo ${#arr[@]}        # 6  echo ${#arr[\*]}        # 6    # Search in Array  echo ${arr[@]/\*[aA]\*/}    # 1    # Replacing Substring Temporary  echo ${arr[@]//a/A}        # prAkhAr Ankit 1 rishAbh mAnish AbhinAv  echo ${arr[@]}            # prakhar ankit 1 rishabh manish abhinav  echo ${arr[0]//r/R}        # pRakhaR |

Output:

prakhar ankit 1 rishabh manish abhinav

prakhar ankit 1 rishabh manish abhinav

prakhar ankit 1 rishabh manish abhinav

prakhar ankit 1 rishabh manish abhinav

prakhar

prakhar

rishabh

ankit

prakhar ankit 1 rishabh manish abhinav

ankit 1 rishabh manish abhinav

1 rishabh manish abhinav

rakhar

ankit 1 rishabh manish

1 rishabh manish

rak

7

7

6

6

1

prAkhAr Ankit 1 rishAbh mAnish AbhinAv

prakhar ankit 1 rishabh manish abhinav

pRakhaR