How to Use Regular Expressions (RegEx) on Linux

Last Updated : 18 Jul, 2021

Regexps are acronyms for regular expressions. Regular expressions are special characters or sets of characters that help us to search for data and match the complex pattern. Regexps are most commonly used with the Linux commands:- grep, sed, tr, vi.

The following are some basic regular expressions:

Sr. no.	Symbol	Description
1.	.	It is called a wild card character, It matches any one character other than the new line.
2.	^	It matches the start of the string.
3.	$	It matches the end of the string.
4.	*	It matches up to zero or more occurrences i.e. any number of times of the character of the string.
5.	\	It is used for escape following character.
6.	()	It is used to match or search for a set of regular expressions.
7.	?	It matches exactly one character in the string or stream.

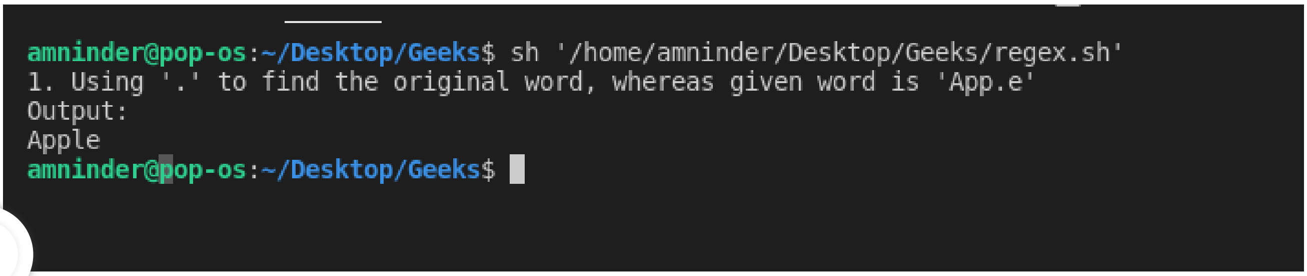
1. Using "." (dot) to match strings.

Using "." we can find a string if we do not know the exact string, or we just remember only the start and end of the string, we can use "." As a missing character, and it will fill that missing character. Let's see an example for better understanding:' This file contains the fruit's name, and we are going to use regular expressions on this file.

Script:

```
#!/bin/sh

# Basic Regular Expression

# 1. Using "." to match strings.

# loading the text file

fruits_file=`cat fruit.txt | grep App.e`

# here the original (answer) word will be Apple,

# but because we don't know the spelling of the Apple,

# we will put a dot (.) in that place.

echo "1. Using '.' to find the original word, whereas given word is 'App.e'"

# displaying output

echo "Output:"

echo "$fruits_file"
```

Output:



```
amninder@pop-os:~/Desktop/Geeks$ sh '/home/amninder/Desktop/Geeks/regex.sh'
1. Using '.' to find the original word, whereas given word is 'App.e'
Output:
Apple
amninder@pop-os:~/Desktop/Geeks$ 
```

## 2. Using "^" (caret) to match the beginning of the string

Using "^", we can find all the strings that start with the given character. Let's see an example for a better understanding. Here we are trying to find all the fruit names that start with the letter B:

Script:

#!/bin/sh

# Basic Regular Expression

# 2. Using "^" (caret) to match the beginning of the string

# loading the text file

fruits_file=`cat fruit.txt | grep ^B`

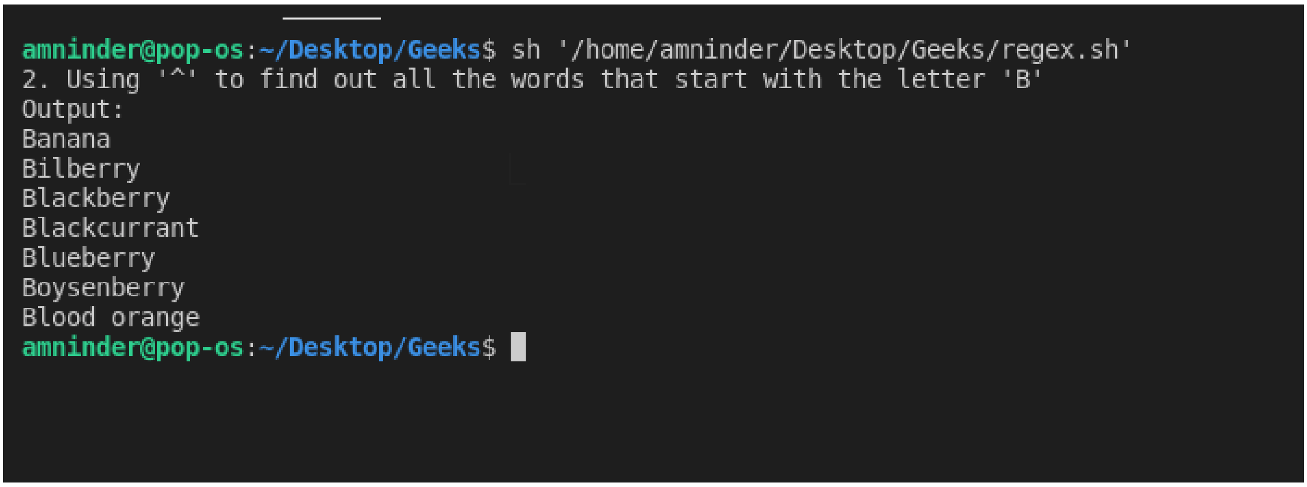
echo "2. Using '^' to find out all the words that start with the letter 'B'"

# displaying output

echo "Output:"

echo "$fruits_file"

Output:



```
amninder@pop-os:~/Desktop/Geeks$ sh '/home/amninder/Desktop/Geeks/regex.sh'
2. Using '^' to find out all the words that start with the letter 'B'
Output:
Banana
Bilberry
Blackberry
Blackcurrant
Blueberry
Boysenberry
Blood orange
amninder@pop-os:~/Desktop/Geeks$ 
```

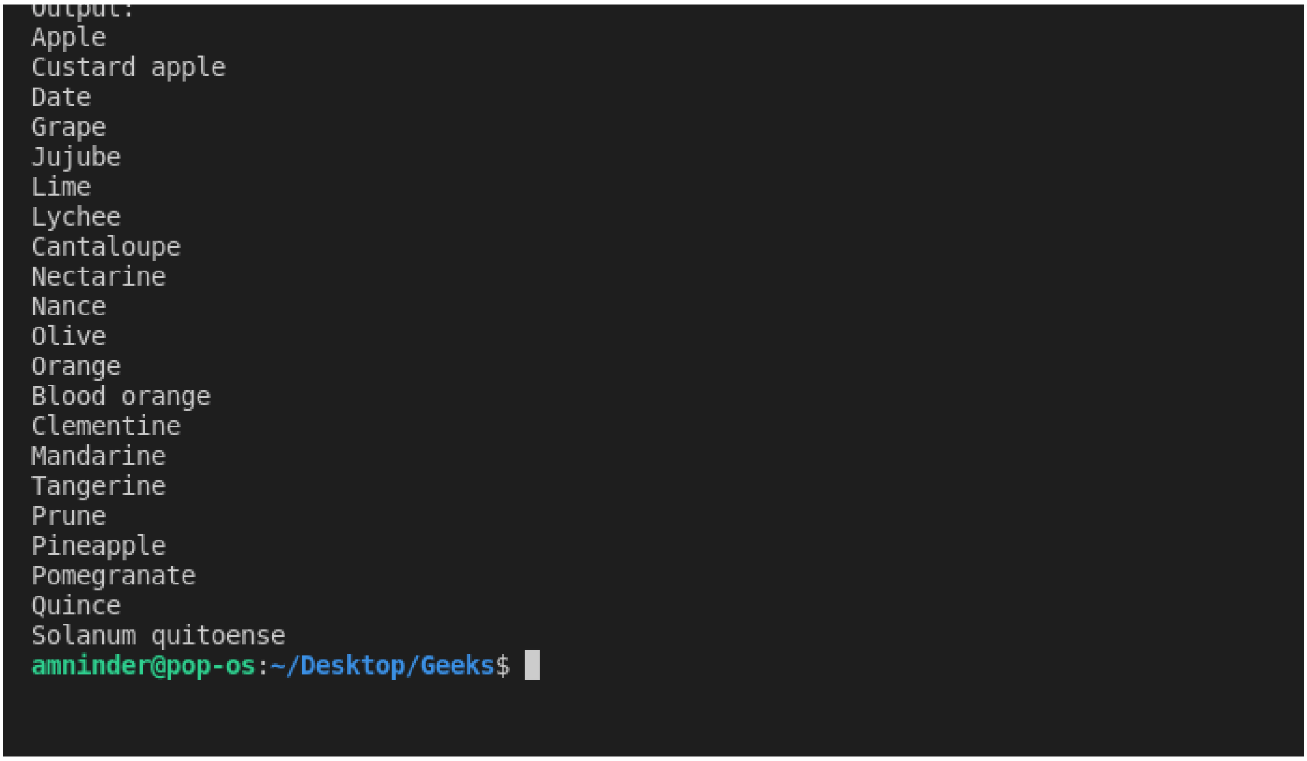
Output 2.

Using “$” we can find all the strings that end with the given character. Let’s see an example for a better understanding. Here we are trying to find all the fruit’s names that end with the letter e:

script:

```
#!/bin/sh

# Basic Regular Expression

# 3. Using “$” (dollar) to match the ending of the string

# loading the text file

fruits_file=`cat fruit.txt | grep e$`

echo “3. Using ‘$’ to find out all the words that end with the letter ‘e'”

# displaying output

echo “Output:”

echo “$fruits_file”
```

Output:



```
Output:
Apple
Custard apple
Date
Grape
Jujube
Lime
Lychee
Cantaloupe
Nectarine
Nance
Olive
Orange
Blood orange
Clementine
Mandarine
Tangerine
Prune
Pineapple
Pomegranate
Quince
Solanum quitoense
amninder@pop-os:~/Desktop/Geeks$
```

Output 3.

4. Using "*" (an asterisk) to find any number of repetitions of a string

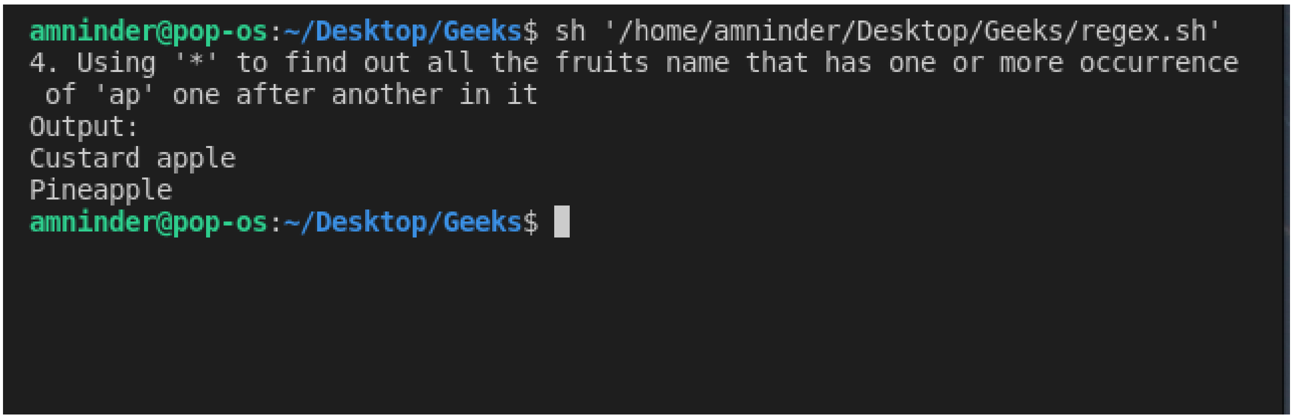
Using "*", we can match up to zero or more occurrences of the character of the string. Let's see an example for a better understanding. Here we are trying to find all the fruit's names that

has one or more occurrences of 'ap' one after another in it.

Script:

```
#!/bin/sh

# Basic Regular Expression

# 4. Using "*" to find any number of repetition of a string

# loading the text file

fruits_file=`cat fruit.txt | grep ap*le`

echo "4. Using '*' to find out all the fruits name that has one or more occurrence of 'ap' one after another in it"

# displaying output
```

Output:



```
amninder@pop-os:~/Desktop/Geeks$ sh '/home/amninder/Desktop/Geeks/regex.sh'
4. Using '*' to find out all the fruits name that has one or more occurrence
 of 'ap' one after another in it
Output:
Custard apple
Pineapple
amninder@pop-os:~/Desktop/Geeks$ 
```

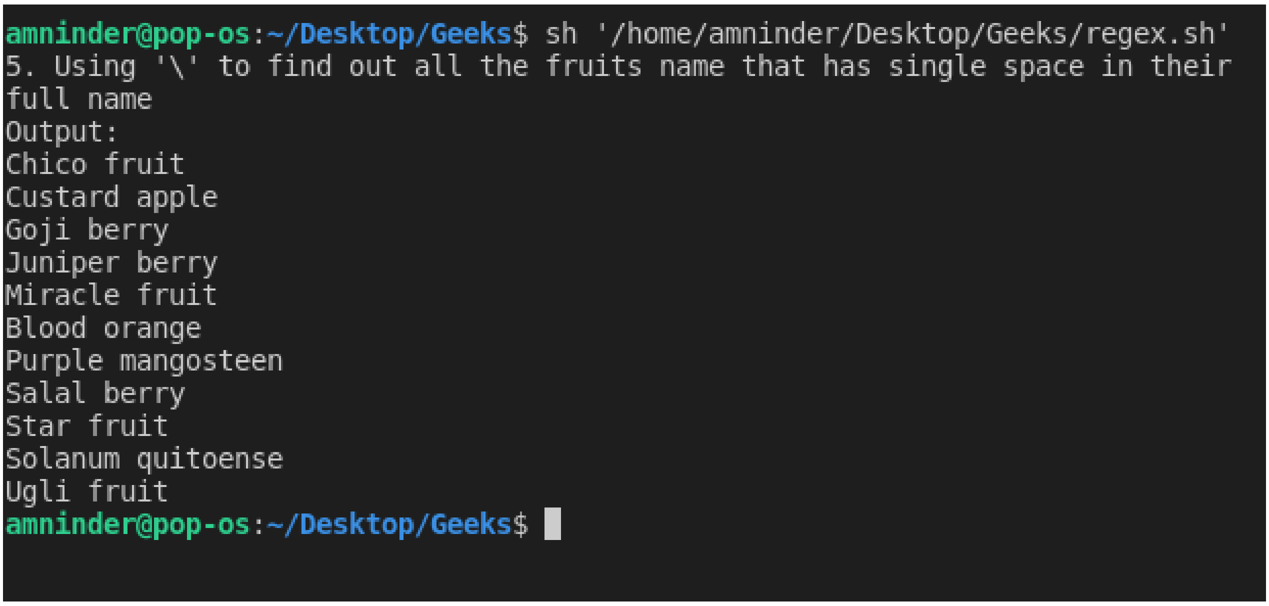
Output 4.

5. Using "\" (a backslash) to match the special symbol

Using "\" with special symbols like whitespace (" "), newline("\n"), we can find strings from the file. Let's see an example for a better understanding. Here we are trying to find all the fruit's names that have space in their full names.

Script:

```
#!/bin/sh

# Basic Regular Expression

# 5. Using "\" to match the special symbol

# loading the text file

fruits_file=`cat fruit.txt | grep  "\ "`

echo "5. Using '\' to find out all the fruits name that has single space in their full name"

# displaying output

echo "Output:"

echo "$fruits_file"
```



```
amninder@pop-os:~/Desktop/Geeks$ sh '/home/amninder/Desktop/Geeks/regex.sh'
5. Using '\' to find out all the fruits name that has single space in their
full name
Output:
Chico fruit
Custard apple
Goji berry
Juniper berry
Miracle fruit
Blood orange
Purple mangosteen
Salal berry
Star fruit
Solanum quitoense
Ugli fruit
amninder@pop-os:~/Desktop/Geeks$ 
```

### 6. Using "()" (braces) to match the group of regexp.

Using "()", we can find matched strings with the pattern in the "()". Let's see an example for a better understanding. Here we are trying to find all the fruit's names that have space in their full name.

Script:

#!/bin/sh

# Basic Regular Expression

# 6. Using "()" (braces) to match the group of regexp.

# loading the text file

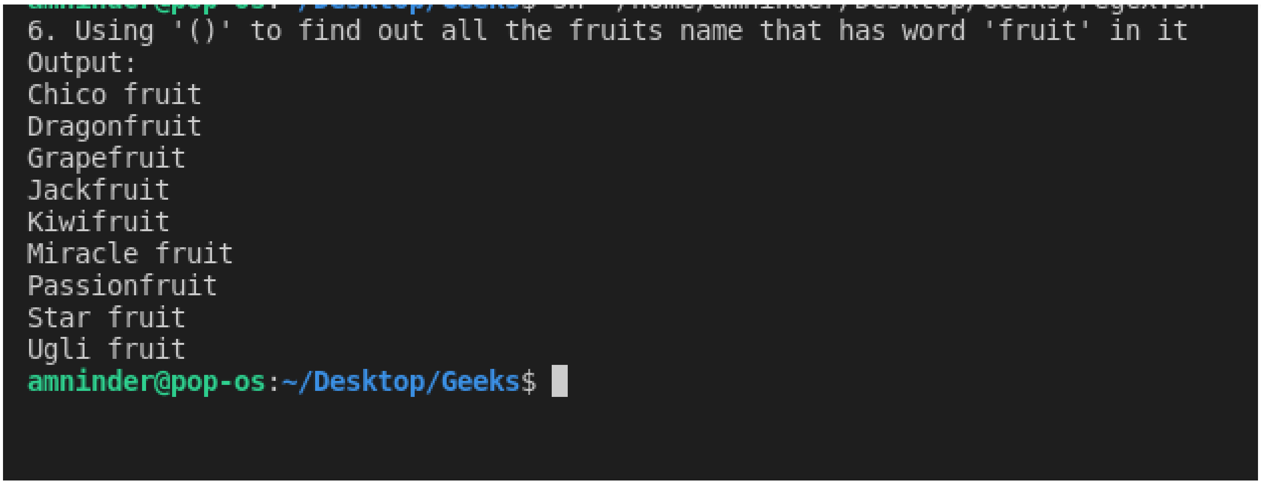
fruits_file=`cat fruit.txt | grep -E "(fruit)"`

echo "6. Using '()' to find out all the fruits name that has word 'fruit' in it"

# displaying output

echo "Output:"

echo "$fruits_file"



```
6. Using '()' to find out all the fruits name that has word 'fruit' in it
Output:
Chico fruit
Dragonfruit
Grapefruit
Jackfruit
Kiwifruit
Miracle fruit
Passionfruit
Star fruit
Ugli fruit
amninder@pop-os:~/Desktop/Geeks$
```

Output 6.

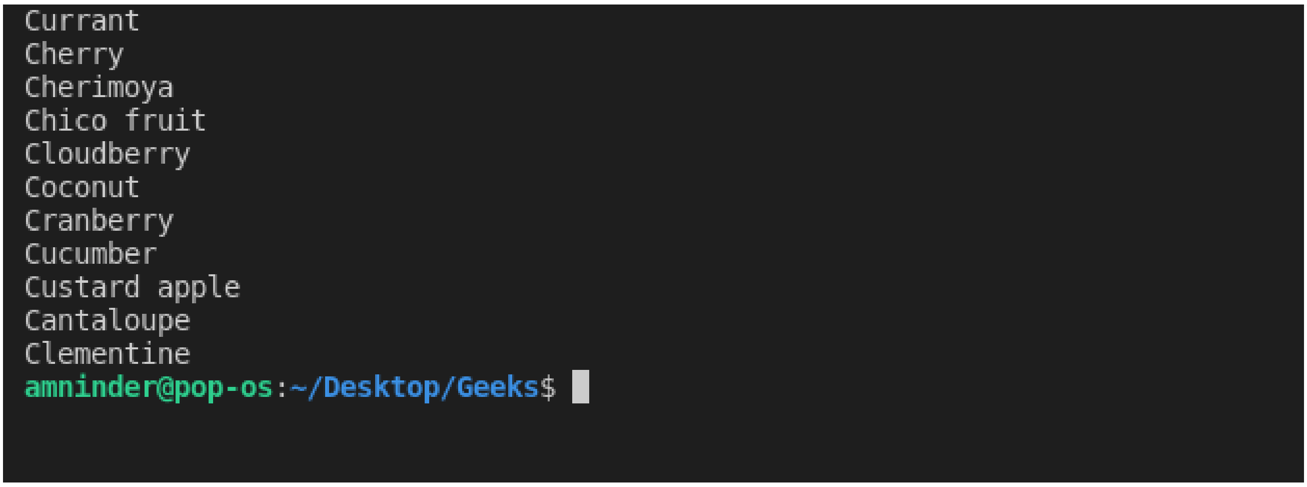
7. Using "?"(question mark) to find all the matching characters

Using "?", we can match 0 or 1 repetitions of the preceding. For example, if we do something like this: ab? It will match either 'a' or 'ab'. Let's see another example for better understanding. Here we are trying to find all the fruit's names that have the character 'Ch' in them.

Script:

```
#!/bin/sh

# Basic Regular Expression

# 7. Using "?"(question mark) to match the

# loading the text file

fruits_file=`cat fruit.txt | grep -E Ch?`

echo "7. Using '?' to find out all the fruits name that has 'Ch' in it"

# displaying output

echo "Output:"

echo "$fruits_file"
```

tput:



```
Currant
Cherry
Cherimoya
Chico fruit
Cloudberry
Coconut
Cranberry
Cucumber
Custard apple
Cantaloupe
Clementine
amninder@pop-os:~/Desktop/Geeks$ 
```

Note: The “<< //// ////” can be used for the purpose of multiline comment.

Like 3

Previous Next