GREP vs. EGREP vs. FGREP

Search for a variety of text fragments through the most powerful commands of terminal i.e. grep, egrep and fgrep.

grep

Definition:

The command `grep` stands for **"global regular expression print"**, and is used to search for specified text patterns in files or program outputs.

Syntax:

```
grep [option(s)] pattern [file(s)]
```

Options:

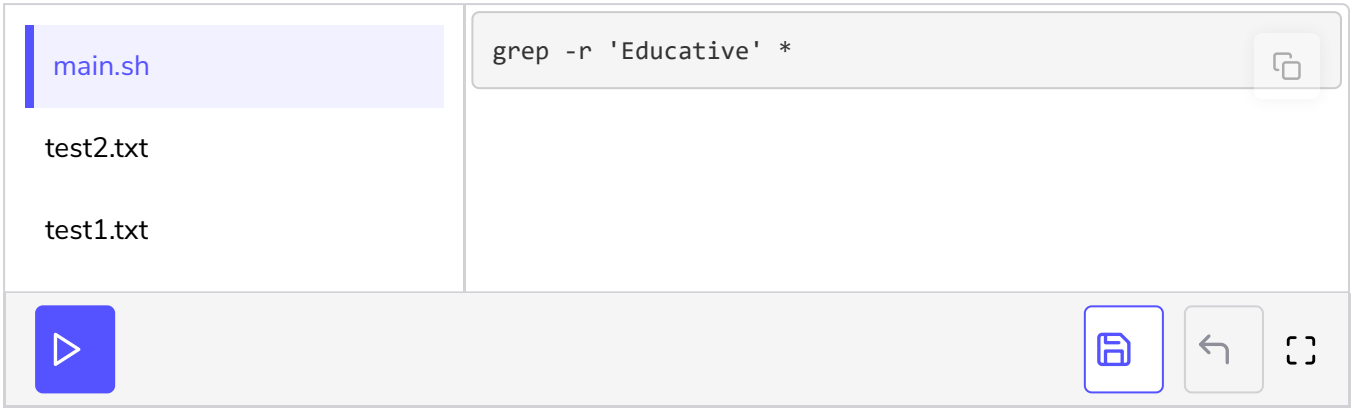
Option	Description
-E (extended regexp)	Causes `grep` to behave like `egrep`.
-F (fixed strings)	Causes `grep` to behave like `fgrep`.
-G (basic regexp)	Causes `grep`, `egrep`, or `fgrep` to behave like the standard **grep** utility.
-r	To search recursively through an entire directory tree (i.e., a directory and all levels of subdirectories within it)
-I	Process a binary file as if it did not contain matching data

	contain matching data.
-c	To report the number of times that the pattern has been matched for each file and to not display the actual lines.
-n	To precede each line of output with the number of the line in the text file from which it was obtained.
-v	It matches only those lines that do not contain the given pattern.
-w	To select only those lines that contain an entire word or phrase that matches the specified pattern.
-x	To select only those lines that match exactly the specified pattern.
-l	To not return the lines containing matches but to only return only the names of the files that contain matches.
-L	It is the opposite of the -l option (and analogous to the -v option) i.e. it will cause grep to return only the names of files that do not contain the specified pattern.

Example:

- This would search all files in the current directory and in all of its subdirectories, for every line containing the string “Educative”:

```
grep -r 'Educative' *
```

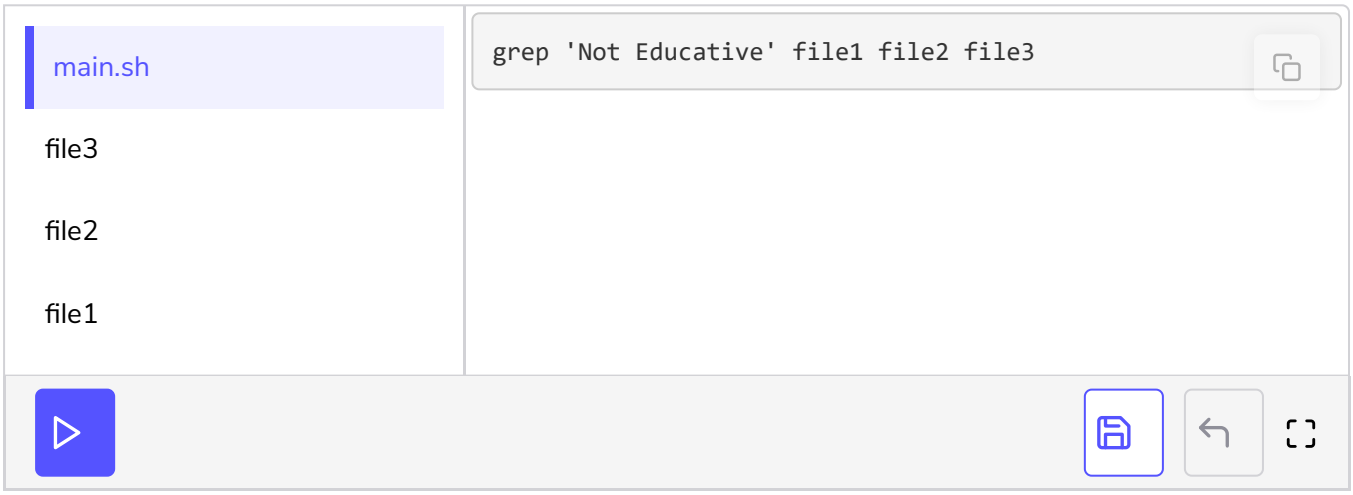


- To search for a word in some respective files:

```
grep Educative file1 file2 file3
```

- `grep` can be used to search for a sequence of strings:

```
grep 'Search commands' file1 file2 file3
```



- Using `grep` to find files based on content:

main.sh

```
find . -type f -print | xargs grep "for"
```

file5.txt

file4.txt

file3.txt

file2.txt

file1.txt

egrep

Definition:

The command `egrep` stands for **“extended global regular expression print”**. It is used for searching particular patterns and is same as `grep` with an `-E` option:

```
grep -E is same as egrep
```

Syntax:

```
egrep [option(s)] pattern [file(s)]
```

Options:

`egrep` accepts same options as `grep` except `-E` and `-F`.

Examples:

- Identifying every line containing a specific string:

To identify every line containing the string “prog” in **file.txt**, execute the following command. Here, `-n` shows the line numbers along with the results:

main.sh

```
egrep -n 'prog' file.txt
```

file.txt

Regex Examples

- Finding lines with specific number of vowels:

To find lines with **2** vowels:

main.sh

file.txt

```
egrep '[aeiou]{2,}' file.txt
```

- Finding lines with specific characters in them and those characters don't come at the end of line:

To find the lines that have **'in'** in them and these lines do not end up on **'in'**:

main.sh

file.txt

```
programmed
system
file
heirarchy
progress
intent
output
program
wings
insight
grin
```

- Finding each line with some sequences of characters:

To find the lines having "pro" or "in":

main.sh

file.txt

```
egrep -n 'pro|in' file.txt
```

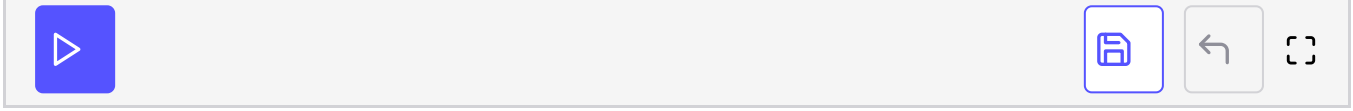
- Finding number of lines with some particular character at the end:

To find the number of lines in **file.txt** with the letter ‘t’ in the end:

main.sh

```
egrep -c 't$' file.txt
```

file.txt



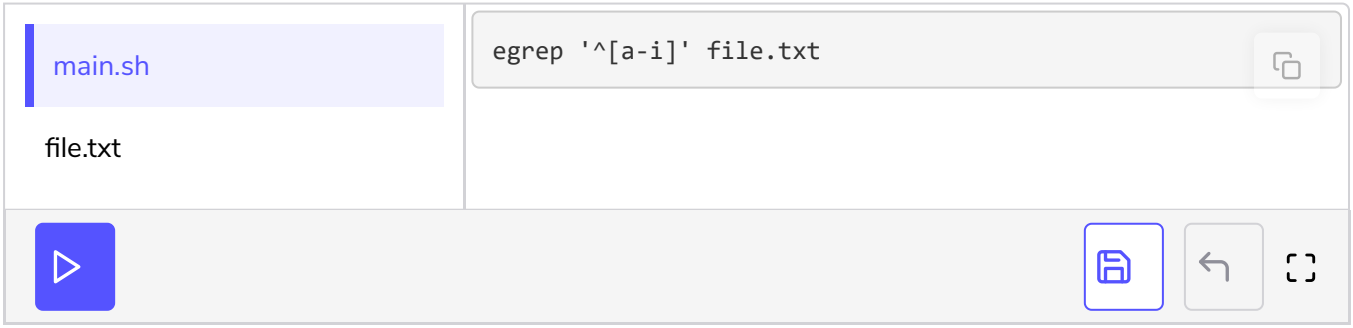
- Finding lines beginning with some specific characters:

To find the lines that begin with letters ranging from **‘c’** to **‘i’**:

main.sh

```
egrep '^[a-i]' file.txt
```

file.txt



fgrep

Definition:

`fgrep` is used to interpret pattern as a list of **fixed strings** (the whole string is interpreted literally), separated by new lines, Hence, regular expressions can’t be used.

Example:

To find the string *“Banana”* along with the line number where it’s found:

main.sh

```
fgrep -n "Banana" file.txt
```

file.txt