

CSCI 3308 Software Development Methods and Tools

Instructor: David Graham

Lab 2 - Regular Expression

TA: Yawen Zhang

Lab 2 - Regular Expressions

Objectives

- ❖ Use regular expressions with common Unix commands
- ❖ Practice using some useful Unix commands
- ❖ Practice creating and running bash shell scripts
- ❖ Practice using pipes

Lab Link

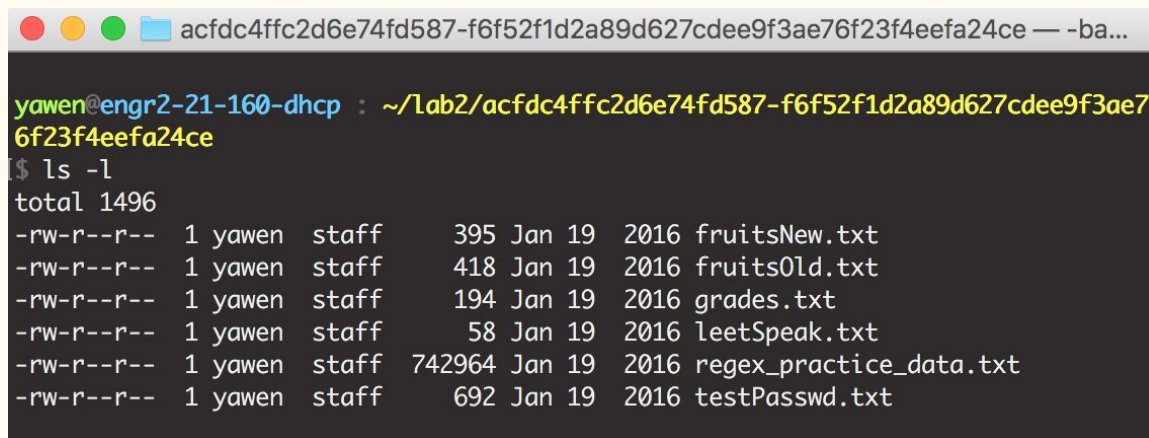
- ❖ <http://www.lousymedia.com/csci-3308/labs/lab-2>

Preparation: download practice files (step 1)

```
curl -L  
https://gist.github.com/dgrah  
am/acfdc4ffc2d6e74fd587/ar  
chive/f6f52f1d2a89d627cdee  
9f3ae76f23f4eefa24ce.zip >  
lab2.zip
```

```
unzip lab2.zip -d lab2
```

```
cd  
lab2/acfdc4ffc2d6e74fd587-f  
6f52f1d2a89d627cdee9f3ae7  
6f23f4eefa24ce
```

A terminal window with a dark background and light text. The title bar shows a folder icon and the path 'acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce' followed by a prompt character and '-ba...'. The terminal content shows a user 'yawn' at a host 'engr2-21-160-dhcp' in the directory '~/Lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce'. The user has entered the command 'ls -l', and the output shows a directory listing with permissions, file sizes, dates, and filenames.

```
acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce — -ba...  
yawn@engr2-21-160-dhcp : ~/Lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae7  
6f23f4eefa24ce  
$ ls -l  
total 1496  
-rw-r--r--  1 yawn  staff    395 Jan 19  2016 fruitsNew.txt  
-rw-r--r--  1 yawn  staff    418 Jan 19  2016 fruitsOld.txt  
-rw-r--r--  1 yawn  staff    194 Jan 19  2016 grades.txt  
-rw-r--r--  1 yawn  staff     58 Jan 19  2016 leetSpeak.txt  
-rw-r--r--  1 yawn  staff 742964 Jan 19  2016 regex_practice_data.txt  
-rw-r--r--  1 yawn  staff    692 Jan 19  2016 testPasswd.txt
```

Practice Unix Commands

❖ **diff** : `diff file1 file2`

❖ **wc** : `wc -l file1`

❖ **cut** : `cut -d : -f 3 file1`

❖ **pipe** : `cut -d : -f 3 file1 | sort > file2`

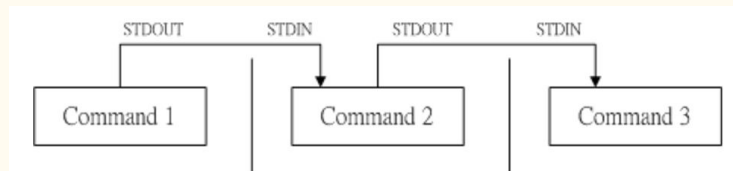
❖ **sed** : `sed s/yourname/myname/g file1`

❖ **awk** : `awk 'NR > 1{print $1}' file1`

❖ **grep** : `grep -c '^[0-9]' file1`

General Unix command format:

`command -option1 argument -option2 argument ...`



awk (step 7)

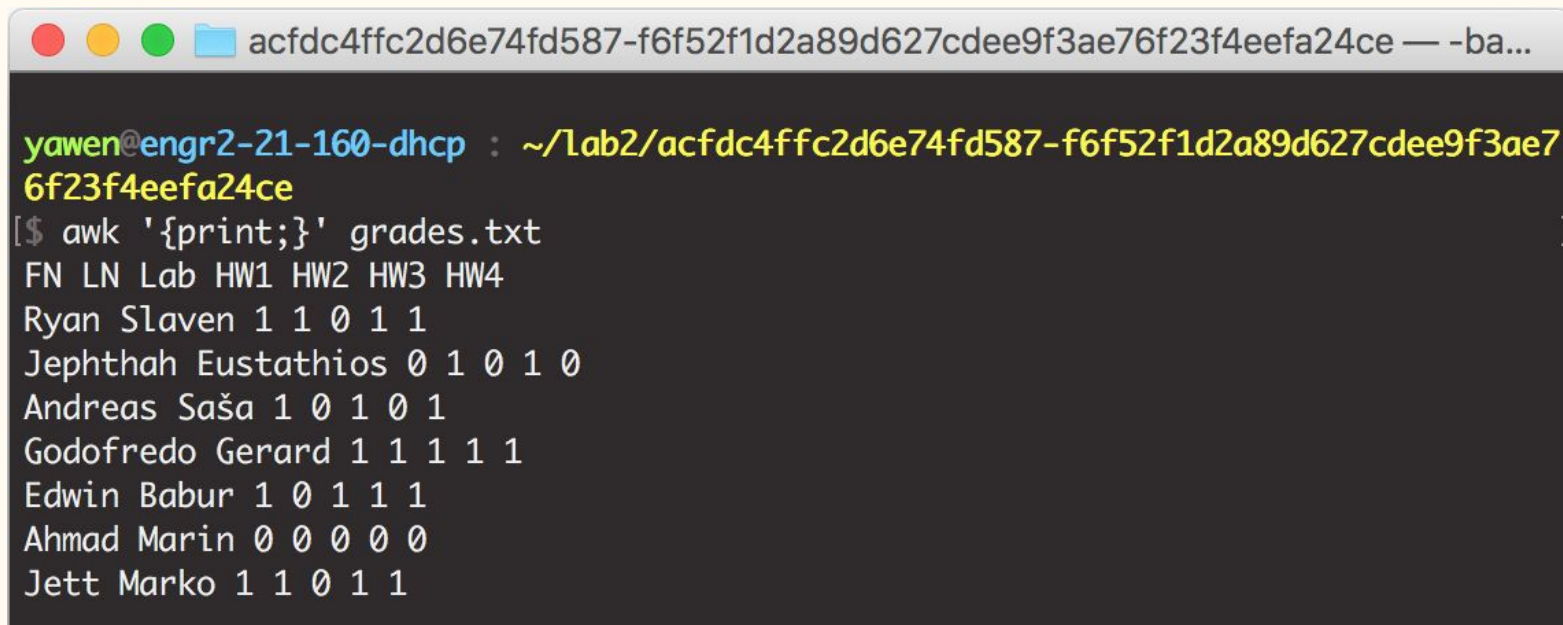
- ❖ a little more difficult, but very powerful text processing command
- ❖ work with lines in a file
- ❖ build-in variables: NR, NF (number of fields), FS (field separator, space by default)

Syntax:

```
awk '/search pattern1/ {Actions}  
    /search pattern2/ {Actions}' file
```

awk (step 7)

- ❖ awk Example 1. Default behavior of awk
- ❖ command: **awk '{print;}' grades.txt**

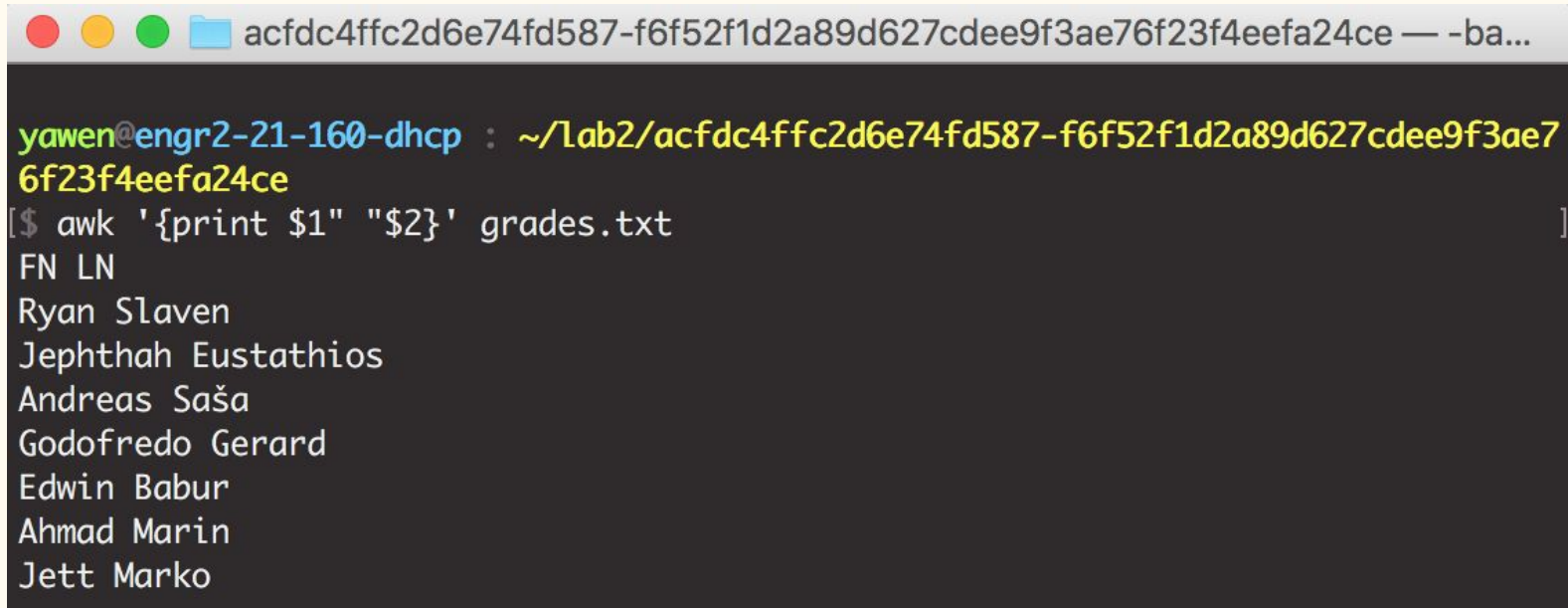


A terminal window with a title bar showing a file path and window controls. The prompt is `yawen@engr2-21-160-dhcp : ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce`. The command `awk '{print;}' grades.txt` has been executed, displaying the contents of the file `grades.txt`.

```
yawen@engr2-21-160-dhcp : ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce  
[$ awk '{print;}' grades.txt]  
FN LN Lab HW1 HW2 HW3 HW4  
Ryan Slaven 1 1 0 1 1  
Jephthah Eustathios 0 1 0 1 0  
Andreas Saša 1 0 1 0 1  
Godofredo Gerard 1 1 1 1 1  
Edwin Babur 1 0 1 1 1  
Ahmad Marin 0 0 0 0 0  
Jett Marko 1 1 0 1 1
```

awk (step 7)

- ❖ awk Example 2. Print only specific field
- ❖ command: **awk '{print \$1" "\$2}' grades.txt**

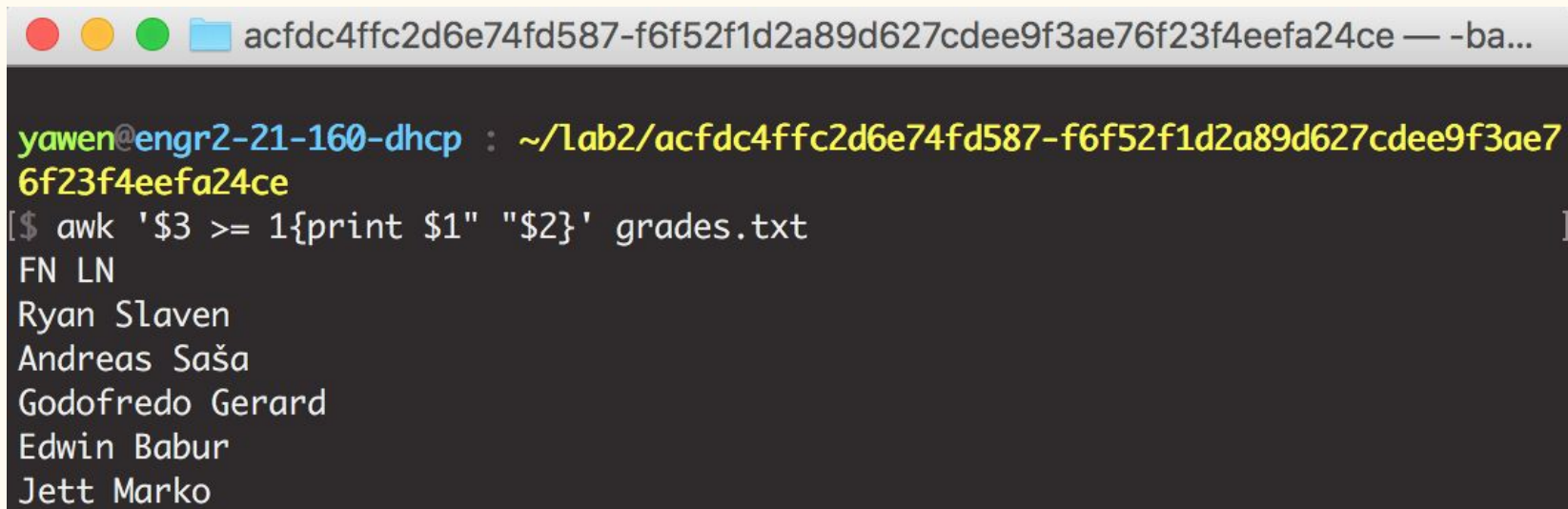


A terminal window with a dark background and light-colored text. The window title bar shows three colored circles (red, yellow, green) and a folder icon, followed by the text 'acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce — -ba...'. The prompt is 'yawen@engr2-21-160-dhcp : ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce'. The command entered is '\$ awk '{print \$1" "\$2}' grades.txt'. The output is a list of names: 'FN LN', 'Ryan Slaven', 'Jephthah Eustathios', 'Andreas Saša', 'Godofredo Gerard', 'Edwin Babur', 'Ahmad Marin', and 'Jett Marko'.

```
acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce — -ba...  
yawen@engr2-21-160-dhcp : ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce  
[$ awk '{print $1" "$2}' grades.txt]  
FN LN  
Ryan Slaven  
Jephthah Eustathios  
Andreas Saša  
Godofredo Gerard  
Edwin Babur  
Ahmad Marin  
Jett Marko
```

awk (step 7)

- ❖ awk Example 3. Print the lines which matches with the pattern
- ❖ command: **awk '\$3 >= 1{print \$1" "\$2}' grades.txt**



```
yawen@engr2-21-160-dhcp : ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce
[$ awk '$3 >= 1{print $1" "$2}' grades.txt
FN LN
Ryan Slaven
Andreas Saša
Godofredo Gerard
Edwin Babur
Jett Marko
```


awk (step 7)

- ❖ awk Example 4. Initialization and Final Action (BEGIN + END)
- ❖ command: **awk 'BEGIN{count = 0}\$3 == 1{count++}END{print "number of students with lab grad equal to 1: "count}' grades.txt**

Syntax:

```
BEGIN { Actions }
```

```
{ACTION} # Action for everyline in a file
```

```
END { Actions }
```

is for comments in Awk

```
acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce — -ba...

yawen@engr2-21-160-dhcp : ~/Lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce
[$ awk 'BEGIN{count = 0}$3 == 1{count++}END{print "number of students with lab grad equal to 1: "count}' grades.txt
number of students with lab grad equal to 1: 5

yawen@engr2-21-160-dhcp : ~/Lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce
[$ awk '$3 == 1' grades.txt
Ryan Slaven 1 1 0 1 1
Andreas Saša 1 0 1 0 1
Godofredo Gerard 1 1 1 1 1
Edwin Babur 1 0 1 1 1
Jett Marko 1 1 0 1 1
```

awk (step 7)

- ❖ awk Example 5. Using build-in variables
- ❖ command: **awk 'NR > 1{print \$1" "\$2}' grades.txt**



A terminal window with a title bar containing window control buttons and a file path. The prompt shows the user is yawen@engr2-21-160-dhcp in the directory ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce. The command awk 'NR > 1{print \$1" "\$2}' grades.txt has been executed, resulting in a list of names.

```
acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce — -ba...  
  
yawen@engr2-21-160-dhcp : ~/lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce  
[$ awk 'NR > 1{print $1" "$2}' grades.txt  
Ryan Slaven  
Jephthah Eustathios  
Andreas Saša  
Godofredo Gerard  
Edwin Babur  
Ahmad Marin  
Jett Marko
```

Basic Syntax in RE (step 8)

abc... Letters
123... Digits
\d Any Digit
\D Any Non-digit character
\w Any Alphanumeric character
\W Any Non-alphanumeric character
\s Any Whitespace
\S Any Non-whitespace character

[abc] Only a, b, or c
[^abc] Not a, b, nor c
[a-z] Characters a to z
[0-9] Numbers 0 to 9

{m} m Repetitions
{m,n} m to n Repetitions

. Any Character (one)
***** Zero or more repetitions
+ One or more repetitions
? Optional character

\. Period

^a Starts with a
a\$ Ends with a (the end of a string)

(abc) Capture Group
(a(bc)) Capture Sub-group
(abc|def) Matches abc or def

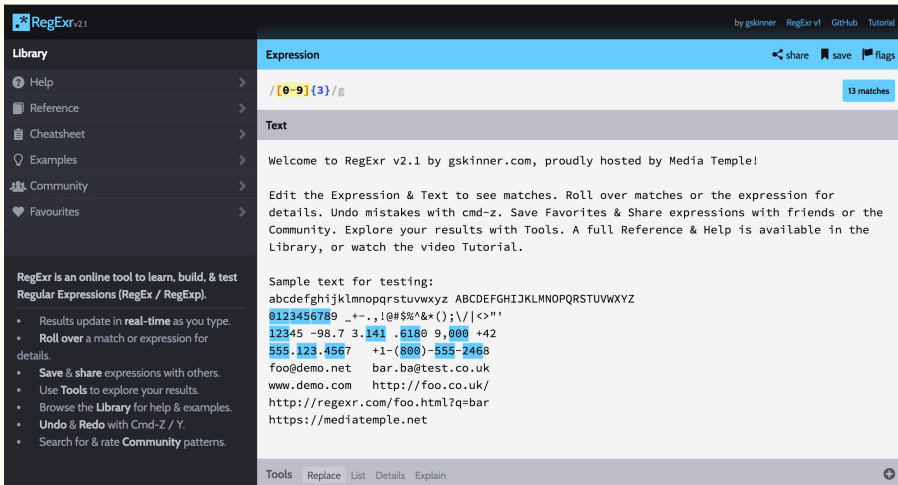
❖ On-line testing with your regular expression commands

<http://regexpr.com/>

Basic Syntax in RE (step 8)

With grep command: **[0-9]\{3\}**

Testing with RE: **[0-9]\{3\}**



RegExr v2.1

Library

- Help
- Reference
- Cheatsheet
- Examples
- Community
- Favourites

RegExr is an online tool to learn, build, & test Regular Expressions (RegEx / RegExp).

- Results update in **real-time** as you type.
- Roll over** a match or expression for details.
- Save & share** expressions with others.
- Use **Tools** to explore your results.
- Browse the **Library** for help & examples.
- Undo & Redo** with Cmd-Z / Y.
- Search for & rate **Community** patterns.

Expression: `[0-9]\{3\}` 13 matches

Text

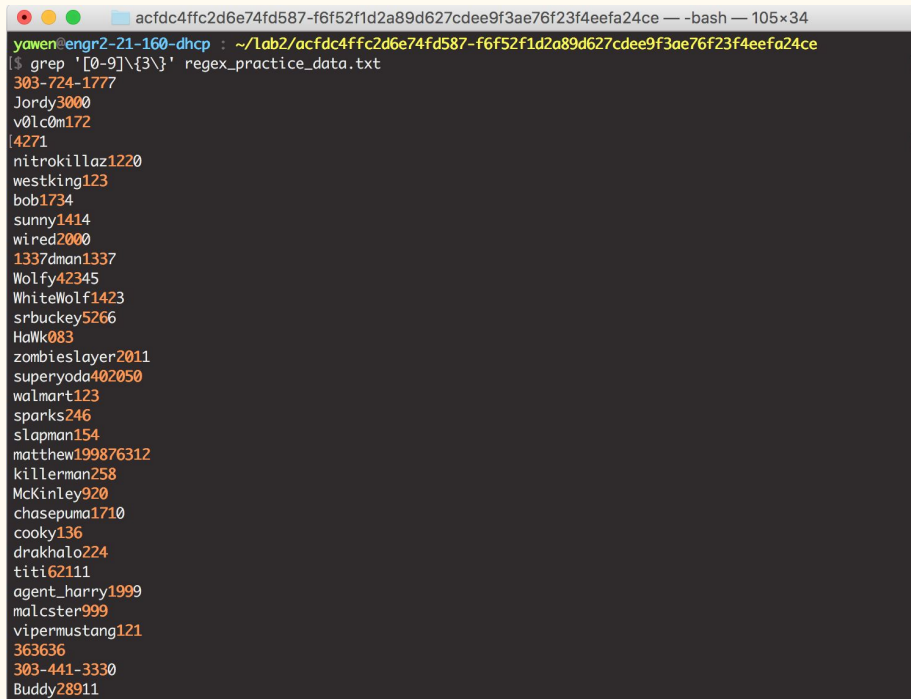
Welcome to RegExr v2.1 by gskinner.com, proudly hosted by Media Temple!

Edit the Expression & Text to see matches. Roll over matches or the expression for details. Undo mistakes with cmd-z. Save Favorites & Share expressions with friends or the Community. Explore your results with Tools. A full Reference & Help is available in the Library, or watch the video Tutorial.

Sample text for testing:

```
abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789 _+-.!@#%&*(){}|<>"'
12345 -98.7 3. 141 .6180 9,000 +42
555.123.4567 +1-(800)-555-2468
foo@demo.net bar.ba@test.co.uk
www.demo.com http://foo.co.uk/
http://regexpr.com/foo.html?q=bar
https://mediatemplate.net
```

Tools Replace List Details Explain



```
acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce — -bash — 105x34
yawen@engr2-21-160-dhcp : ~/Lab2/acfdc4ffc2d6e74fd587-f6f52f1d2a89d627cdee9f3ae76f23f4eefa24ce
$ grep '[0-9]\{3\}' regex_practice_data.txt
303-724-1777
Jordy3000
v0lc0m172
4271
nitrokillaz1220
westking123
bob1734
sunny1414
wired2000
1337dman1337
Wolfy42345
WhiteWolf1423
sruckey5266
HaWk083
zombieslayer2011
superyoda402050
walmart123
sparks246
slapman154
matthew199876312
killerman258
McKinley920
chasepuma1710
cooky136
drakhalo224
titi62111
agent_harry1999
malcster999
vipermustang121
363636
303-441-3330
Buddy28911
```

Lab Task

- ❖ Task: **step 1 - 8**
- ❖ For submission, prepare a .txt file (you may try editing it with **vim** or **any text editor** you like)
- ❖ Save all the commands you use for **step 2 - 8** in the .txt file
- ❖ When you **finish**, show me your **.txt file** and get grade for the lab

Search the Internet !

when you get puzzled