Data file as below.

plym fury 1970 73 2500

chevy malibu 1999 60 3000

ford mustang 1965 45 10000

volvo s80 1998 102 9850

ford thundbd 2003 15 10500

chevy malibu 2000 50 3500

bmw 325i 1985 115 450

honda accord 2001 30 6000

ford taurus 2004 10 17000

toyota rav4 2002 180 750

chevy impala 1985 85 1550

ford explor 2003 25 9500

**Requirement**

1. **Write an awk expression that produces the following output.**

1999 malibu $3000

1965 mustang $10000

2003 thundbd $10500

2000 malibu $3500

2004 taurus $17000

awk '$2 ~ /^[tm]/ {print $3, $2, "$"$NF}' cars

1. **Find all cars whose year ends with 5.**

**Output should be like.**

1965 ford $10000

1985 bmw $450

1985 chevy $1550

awk '$3 ~ /[5]/ {print $3, $1, "$"$NF}' cars

1. **Find all cars which are made in 1985**

bmw     325i    1985    115     450

chevy   impala  1985    85      1550

awk '$3 ~ /1985/ {print $1, $2, $3, $4 "$"$NF}' cars

1. **Find all cars whose price is less than or equal to  3000**

awk '$NF <=3000 {print $1, $2, $3, $4 "$"$NF}' cars

1. **Find all cars whose price is greater than 3000**

plym    fury    1970    73      2500

chevy   malibu  1999    60      3000

bmw     325i    1985    115     450

toyota  rav4    2002    180     750

chevy   impala  1985    85      1550

awk '$NF > 3000 {print $1, $2, $3, $4 "$"$NF}' cars

1. **Find all cars greater than 2000 and less than 9000**

plym    fury    1970    73      2500

chevy   malibu  1999    60      3000

chevy   malibu  2000    50      3500

bmw     325i    1985    115     450

honda   accord  2001    30      6000

toyota  rav4    2002    180     750

awk '$NF > 2000 && $NF < 9000 {print $1, $2, $3, $4 "$"$NF}' cars

**Range operator. You can use , to find a range of cars. But remember range will go through to the groups. Second example print it.**

Find all cars between volvo and bmw

volvo   s80     1998    102     9850

ford    thundbd 2003    15      10500

chevy   malibu  2000    50      3500

bmw     325i    1985    115     450

awk '/volvo/,/bmw/ {print $1, $2, $3, $4 "$"$NF}' cars

Exercise:

**Print the model below.  Put the program in pr\_header file.**

Make    Model   Year    Miles   Price

plym    fury    1970    73      2500

chevy   malibu  1999    60      3000

ford    mustang 1965    45      10000

volvo   s80     1998    102     9850

awk -f pr\_hearder cars

Do another one which prints the file below.

Make    Model   Year    Miles   Price

---------------------------------------------------

plym    fury    1970    73      2500

chevy   malibu  1999    60      3000

ford    mustang 1965    45      10000

volvo   s80     1998    102     9850

awk -f pr\_hearder1 cars

**Exercise: Print the length of line followed by line in cars file**

21 bmw  325i    1985    115     450

22 plym fury    1970    73      2500

23 volvo        s80     1998    102     9850

24 ford explorer  2003    25      9500

24 toyota       rav4    2002    180     750

25 chevy        impala  1985    85      1550

25 chevy        malibu  1999    60      3000

25 chevy        malibu  2000    50      3500

25 ford taurus  2004    10      17000

25 honda        accord  2001    30      6000

26 ford mustang 1965    45      10000

26 ford thundbd 2003    15      10500

awk '{print length($0), $0}' cars

**Exercise:**

Write a file called spear\_demo which prints the output below. Ply is changed to plymouth and chevy is changed to chevrolet.

plymouth fury 1970 73 2500

chevrolet malibu 1999 60 3000

ford    mustang 1965    45      10000

volvo   s80     1998    102     9850

ford    thundbd 2003    15      10500

chevrolet malibu 2000 50 3500

bmw     325i    1985    115     450

honda   accord  2001    30      6000

ford    taurus  2004    10      17000

toyota  rav4    2002    180     750

chevrolet impala 1985 85 1550

ford    explor  2003    25      9500

awk -f spear\_demo cars

**Produce a report as follows**

Miles

Make       Model       Year      (000)      Price

-------------------------------------------------------------------

plymouth   fury        1970      73     $  2500.00

chevrolet  malibu      1999      60     $  3000.00

ford       mustang     1965      45     $ 10000.00

volvo      s80         1998     102     $  9850.00

ford       thundbd     2003      15     $ 10500.00

chevrolet  malibu      2000      50     $  3500.00

bmw        325i        1985     115     $   450.00

honda      accord      2001      30     $  6000.00

ford       taurus      2004      10     $ 17000.00

toyota     rav4        2002     180     $   750.00

chevrolet  impala      1985      85     $  1550.00

ford       explor      2003      25     $  9500.00

**Exercice**

1. Print the next available uid. In the line below. 8 is the uid and it is third field

You need to run awk against /etc/passwd.

mail:x:8:10:mail:/var/mail:/usr/sbin/nologin

awk -F”:” ‘$3 ~’/^9/’ {print $0}’ /etc/passwd

1. Print the cars report. Remember each field should be separated by tab. Instead of price at the end you need to print inexpensive, expensive , please ask.

Inexpensive is <= 5000, greater than 5000 and less than 10000 expensive

 >  10000 “please ask”

plym       fury        1970       73    inexpensive

chevy      malibu      1999       60    inexpensive

ford       mustang     1965       45    expensive

volvo      s80         1998      102    please ask

ford       thundbd     2003       15    expensive

chevy      malibu      2000       50    inexpensive

bmw        325i        1985      115    inexpensive

honda      accord      2001       30    please ask

ford       taurus      2004       10    expensive

toyota     rav4        2002      180    inexpensive

chevy      impala      1985       85    inexpensive

ford       explorer      2003       25    please ask

awk -f carsreport cars

**Exercise:**

Write a shell script that takes 2 variables . First variable is the name of the car and second is the summary of the count.

./manuf.sh 1 car

bmw                1

chevy               3

ford                4

honda               1

plym                1

toyota              1

volvo               1

./manuf.sh

awk ' { arr[$1]++ } END {for(i in arr) {print i , arr[i] }} ' cars

./manuf.sh 3 cars

1965                1

1970                1

1985                2

1998                1

1999                1

2001                1

2002                1

2003                2

2004                1

./manuf1.sh

awk ' { arr[$3]++ } END {for(i in arr) {print i , arr[i] }} ' cars

**Write a script like word\_usage which counts the number of words in a file Very difficult**

**Write a script that prints a report as follows on the following cars.**

**Hint** use the () to get the data and clever use of NR

Report for Wed Jan 31, 2018

2500    plym

3000    chevy

10000   ford

9850    volvo

10500   ford

3500    chevy

450     bmw

6000    honda

17000   ford

750     toyota

1550    chevy

9500    ford

./report1.sh

Don’t do it for now.

**Produce the report from the numbers file.**

cat numbers

10      20      30.3    40.5

20      30      45.7    66.1

30      xyz     50      70

40      75      107.2   55.6

50      20      30.3    40.5

60      30      45.O    66.170

70      1134.7  50      70

80      75      107.2   55.6

90      176     30.3    40.5

100     1027.45 45.7    66.1

110     123     50      57a.5

120     75      107.2   55.6

Produce a report as follows.

./tally

Record 3 skipped:

        30      xyz     50      70

Record 6 skipped:

        60      30      45.O    66.1

Record 11 skipped:

        110     123     50      57a.5

$ cat tally.out

     10.00     20.00     30.30     40.50

     20.00     30.00     45.70     66.10

     40.00     75.00    107.20     55.60

     50.00     20.00     30.30     40.50

     70.00   1134.70     50.00     70.00

     80.00     75.00    107.20     55.60

     90.00    176.00     30.30     40.50

    100.00   1027.45     45.70     66.10

    120.00     75.00    107.20     55.60

   -------   -------   -------   -------

    580.00   2633.15    553.90    490.50

        Grand Total 4257.55

 > operator adds to the file.

cat passwd. Uid is the third field. Password is second field.

bill::102:100:ext 123:/home/bill:/bin/bash

roy:x:104:100:ext 475:/home/roy:/bin/bash

tom:x:105:100:ext 476:/home/tom:/bin/bash

lynn:x:166:100:ext 500:/home/lynn:/bin/bash

mark:x:107:100:ext 112:/home/mark:/bin/bash

sales:x:108:100:ext 102:/m/market:/bin/bash

anne:x:109:100:ext 355:/home/anne:/bin/bash

toni::164:100:ext 357:/home/toni:/bin/bash

ginny:x:115:100:ext 109:/home/ginny:/bin/bash

chuck:x:116:100:ext 146:/home/chuck:/bin/bash

neil:x:164:100:ext 159:/home/neil:/bin/bash

rmi:x:118:100:ext 178:/home/rmi:/bin/bash

vern:x:119:100:ext 201:/home/vern:/bin/bash

bob:x:120:100:ext 227:/home/bob:/bin/bash

janet:x:122:100:ext 229:/home/janet:/bin/bash

maggie:x:124:100:ext 244:/home/maggie:/bin/bash

dan::126:100::/home/dan:/bin/bash

dave:x:108:100:ext 427:/home/dave:/bin/bash

mary:x:129:100:ext 303:/home/mary:/bin/bash

bill (ext 123) has no password.

toni (ext 357) has no password.

neil has the same UID as toni : UID = 164

dan has no password.

dave has the same UID as sales : UID = 108

awk -f passwd\_check passwd

* Need to work out UID part

**Produce a report as below . Don’t do it for now.**

./list\_cars (The output after doing necessary work with awk should be written to a temporary file). The temporary file needs to be delete  on signals 1, 2, 15 and

There should be no errors when removing. Should print script aborted and exit with an error code of 1.

Price range (for example, 5000 7500):3000 8000

Miles     Make       Model       Year    (000)         Price

--------------------------------------------------

chevrolet  malibu      1999       60    $  3000.00

chevrolet  malibu      2000       50    $  3500.00

honda      accord      2001       30    $  6000.00

 ./list\_cars

Price range (for example, 5000 7500):0 2000

Miles       Make       Model       Year    (000)         Price

--------------------------------------------------

bmw        325i        1985      115    $   450.00

toyota     rav4        2002      180    $   750.00

chevrolet  impala      1985       85    $  1550.00

 ./list\_cars

Price range (for example, 5000 7500):15000 100000

MilesMake       Model       Year    (000)         Price

--------------------------------------------------

ford       taurus      2004       10    $ 17000.00

**Exercises**

1. Write a gawk program that numbers each line in a file and sends its output to standard output.

awk '{print NR , $0}' cars

2. Write a gawk program that displays the number of characters in the first field followed by the first field and sends its output to standard output. length($1),

awk '{print length($1) , $1}' cars

3. Write a gawk program that uses the cars file (page 645), displays all cars priced at more than $5,000, and sends its output to standard output.

awk ' $NF >5000 {print $1,$2,$3,$4, $NF}' cars

4. Use awk to determine how many lines in /etc/services contain the string Mail. Verify your answer using grep.

awk '/mail/ {print NR}' /etc/services | wc -l

grep mail /etc/services | wc -l

6. Write a gawk (not awk or mawk) program named net\_list that reads from the rfc-retrieval.txt file on www.rfc-editor.org

displays a the last word on each line in all uppercase letters.

curl -o rfc-retrieval.txt [www.rfc-editor.org](http://www.rfc-editor.org)

sed -e 's/<[^>]\*>//g' rfc-retrieval.txt | gawk '{print toupper($NF)}'