**EXPERIMENT 11**

**AIM**

Experiment to study basics of Perl script.

a) Write a perl script to compute the power of a given number

b) Write a perl script to check a number is prime or not

**THEORY**

Perl stands for Practical Extraction and Report Language. It is used for network administration, Automation, GUI, Databases, etc.

Can be executed from command line and creating a script file

Perl -e ‘ print”Hello” ’ -e enables the perl interpreter

Properties:

Extension pl

Case sensitive

Loosely typed

$ scalar type

@ array

% hash

Various symbols:

. concatenation

x repetition

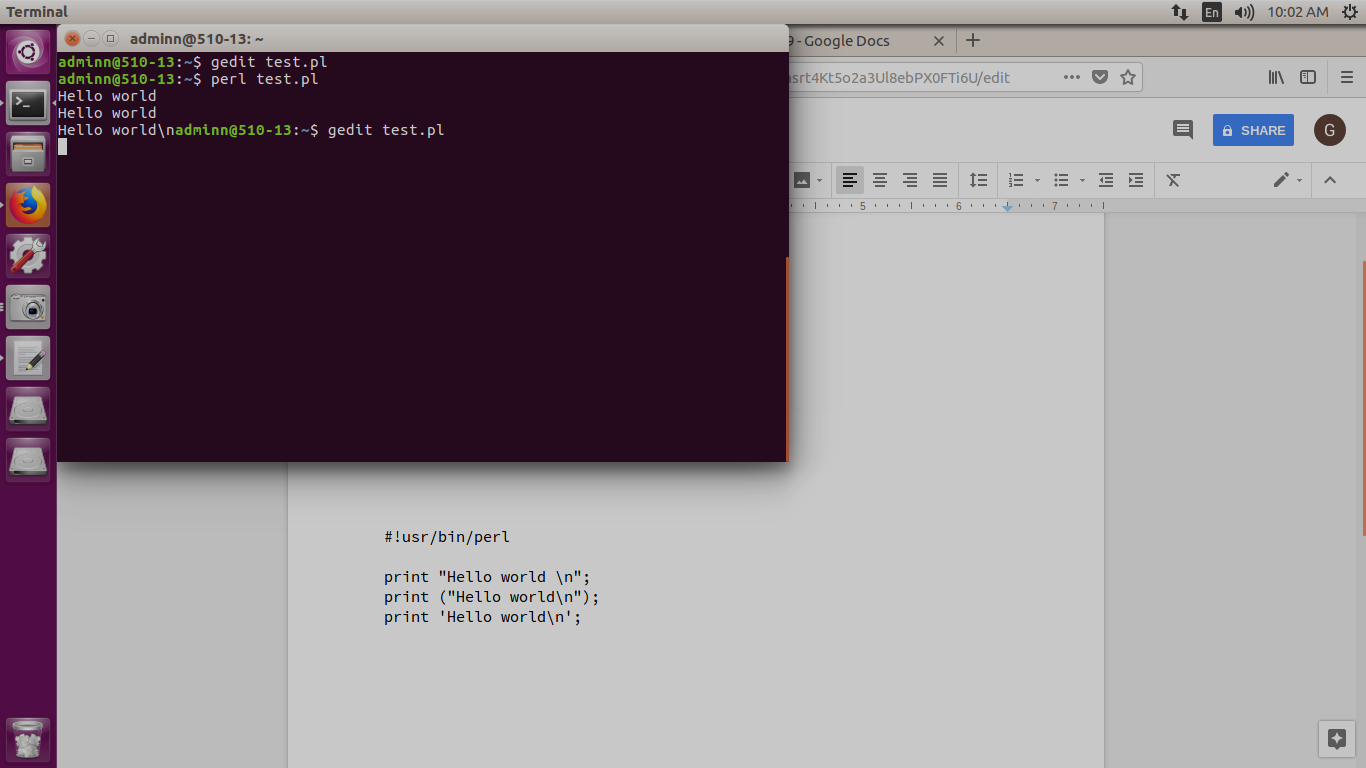
.. range

#!usr/bin/perl

print "Hello world \n";

print ("Hello world\n");

print 'Hello world\n';



#!usr/bin/perl

$a=10;

$b=20;

print "a=$a, b=$b\n";

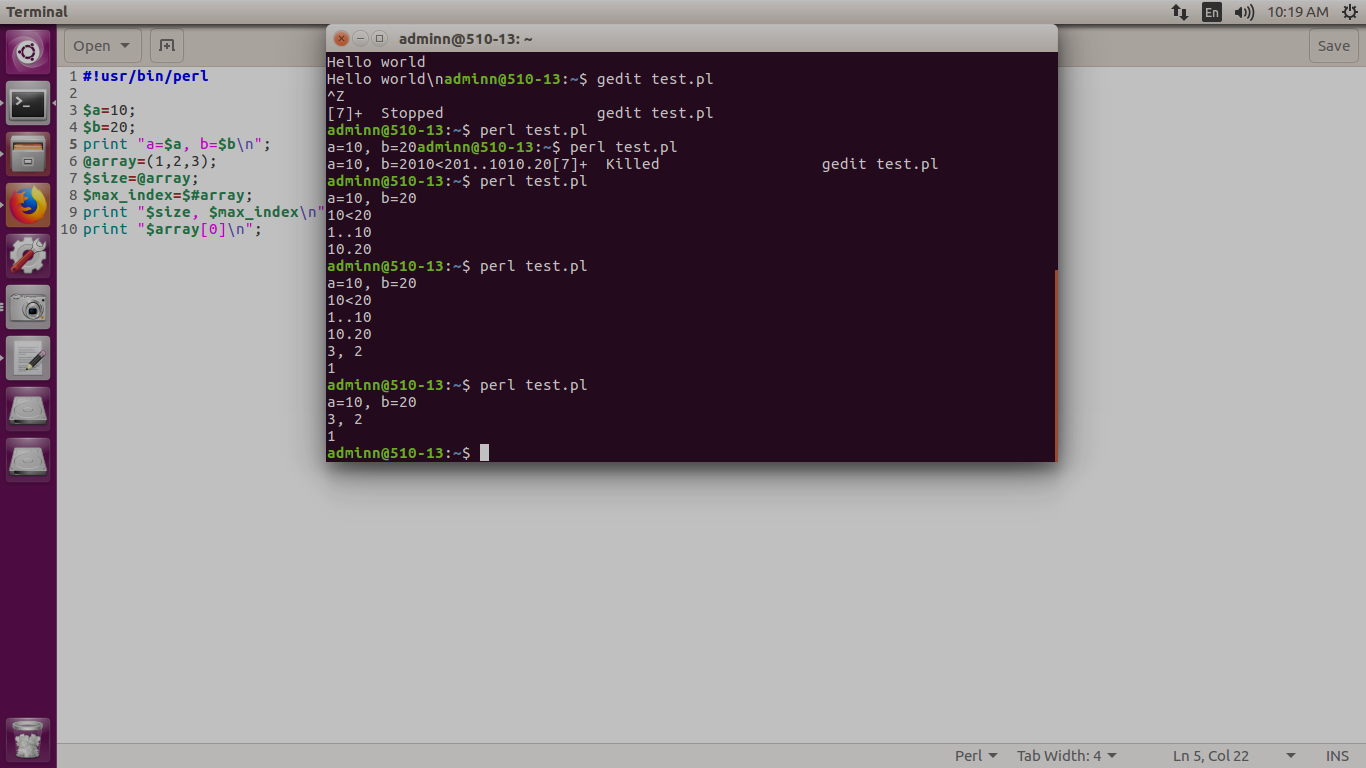
@array=(1,2,3);

$size=@array;

$max\_index=$#array;

print "$size, $max\_index\n";

print "$array[0]\n";



#!usr/bin/perl

$a="abc";

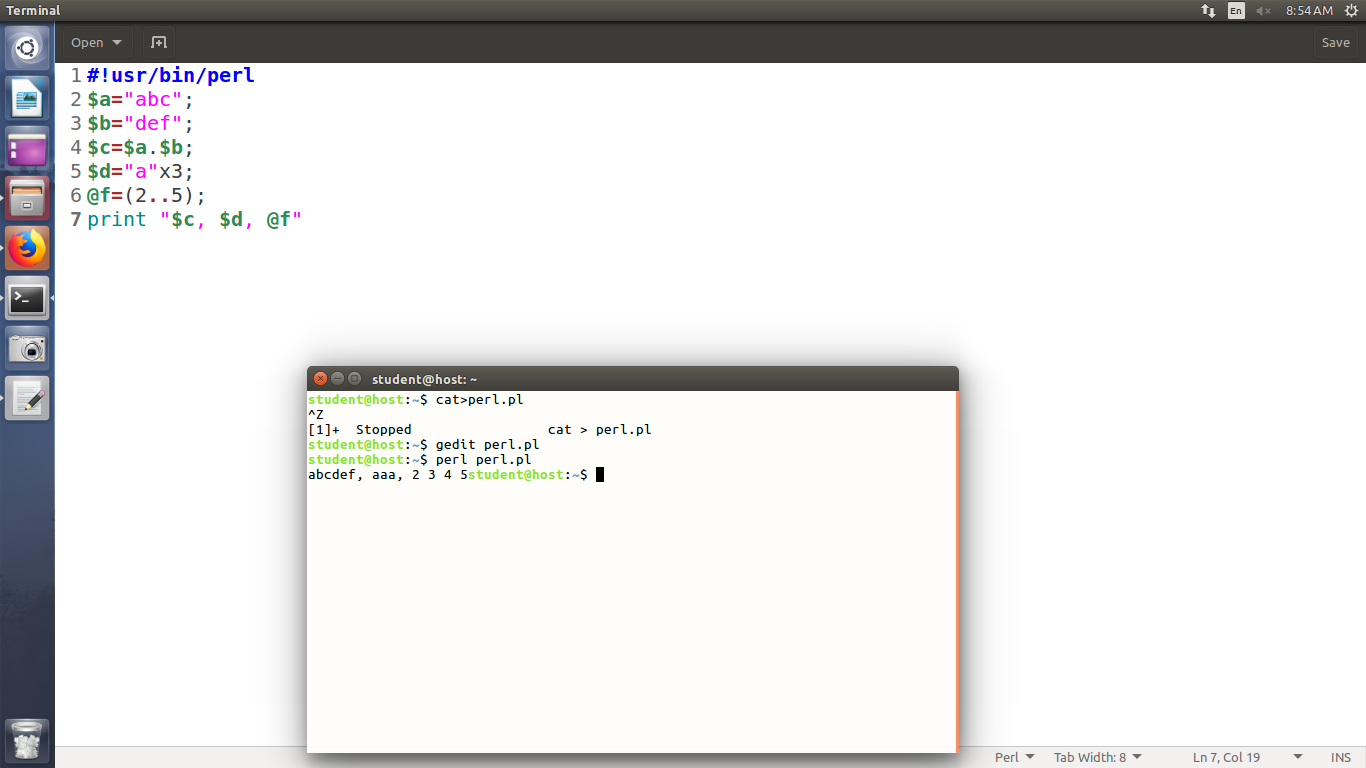
$b="def";

$c=$a.$b;

$d="a"x3;

@f=(2..5);

print "$c, $d, @f"



*if(condn)*

*{*

*}*

*if(condn)*

*{*

*}*

*else{ }*

*if(condn)*

*{*

*}*

*elseif(condn){*

*}*

*else{ }*

*unless(condn)*

*{*

*}*

*else*

*{*

*}*

*Identifiers:scalar, array, hash*

*While*

*until*

*For*

*Foreach*

*Do while*

UNLESS:

#!usr/bin/perl

print "Enter a number:";

$a=<STDIN>;

unless($a%2==0)

{

print "odd\n";

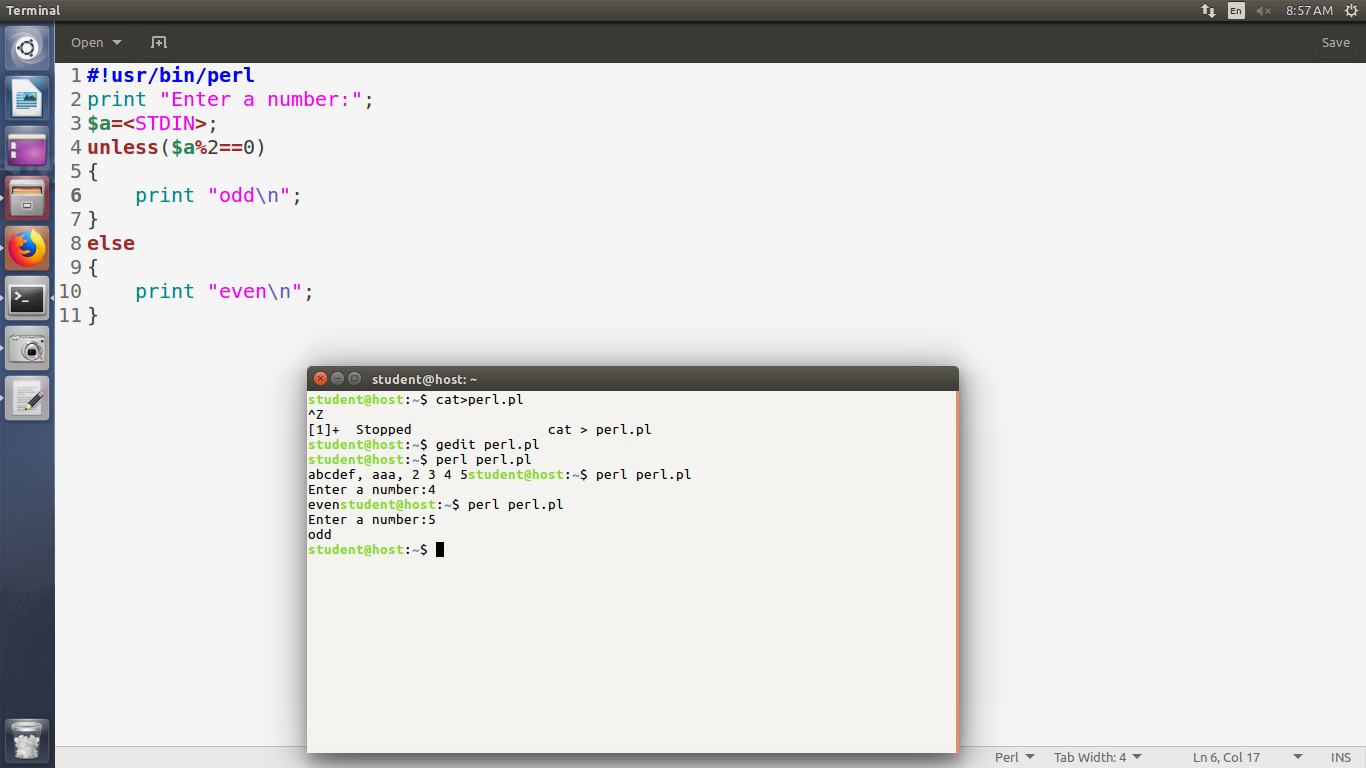
}

else

{

print "even\n";

}



IF:

#!usr/bin/perl

print "Enter a number:";

$a=<STDIN>;

if($a%2==0)

{

print "Even number\n";

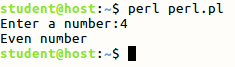
}

else

{

print "Odd number\n";

}



FOREACH:

#!usr/bin/perl

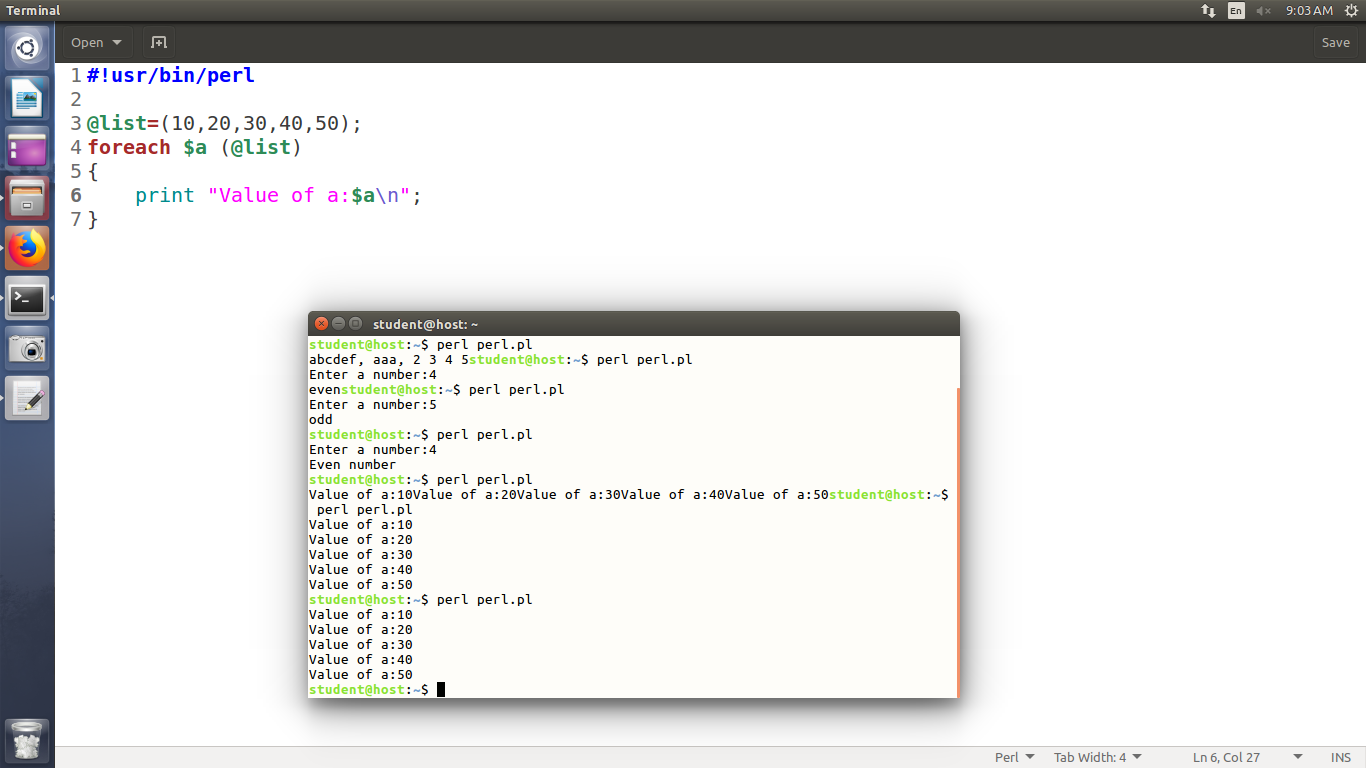
@list=(10,20,30,40,50);

foreach $a (@list)

{

print "Value of a:$a\n";

}



*Push*

*Pop*

*Shift*

*Unshift*

*Slice*

#!usr/bin/perl

@list=(10,50,40,20,30);

foreach $a (@list)

{

print "Value of a:$a\n";

}

push(@list, 60);

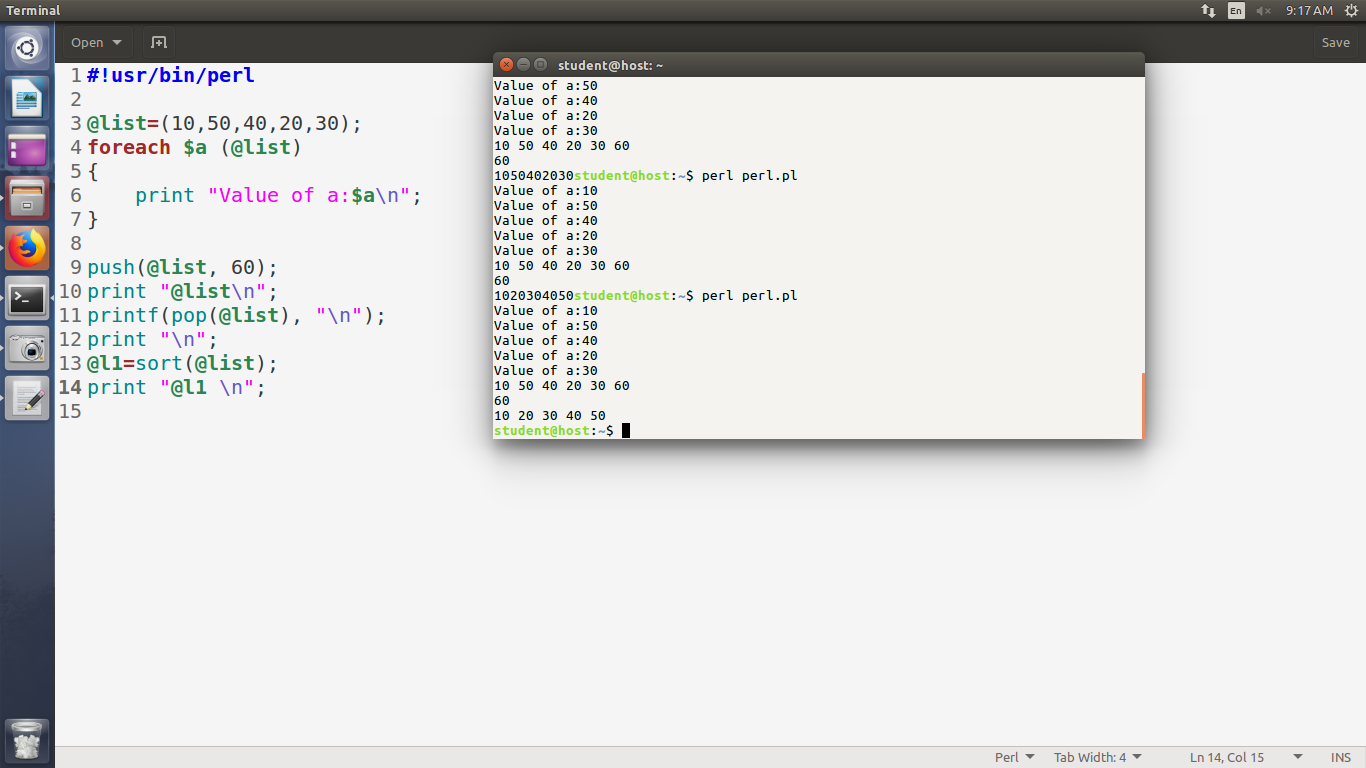
print "@list\n";

printf(pop(@list), "\n");

print "\n";

@l1=sort(@list);

print "@l1 \n";



FUNCTIONS:

#!usr/bin/perl

sub Hello

{

print "Hello\n";

}

sub Average

{

$h=scalar(@\_);

$sum=0;

foreach $num (@\_)

{

$sum+=$num;

}

$avg=$sum/$h;

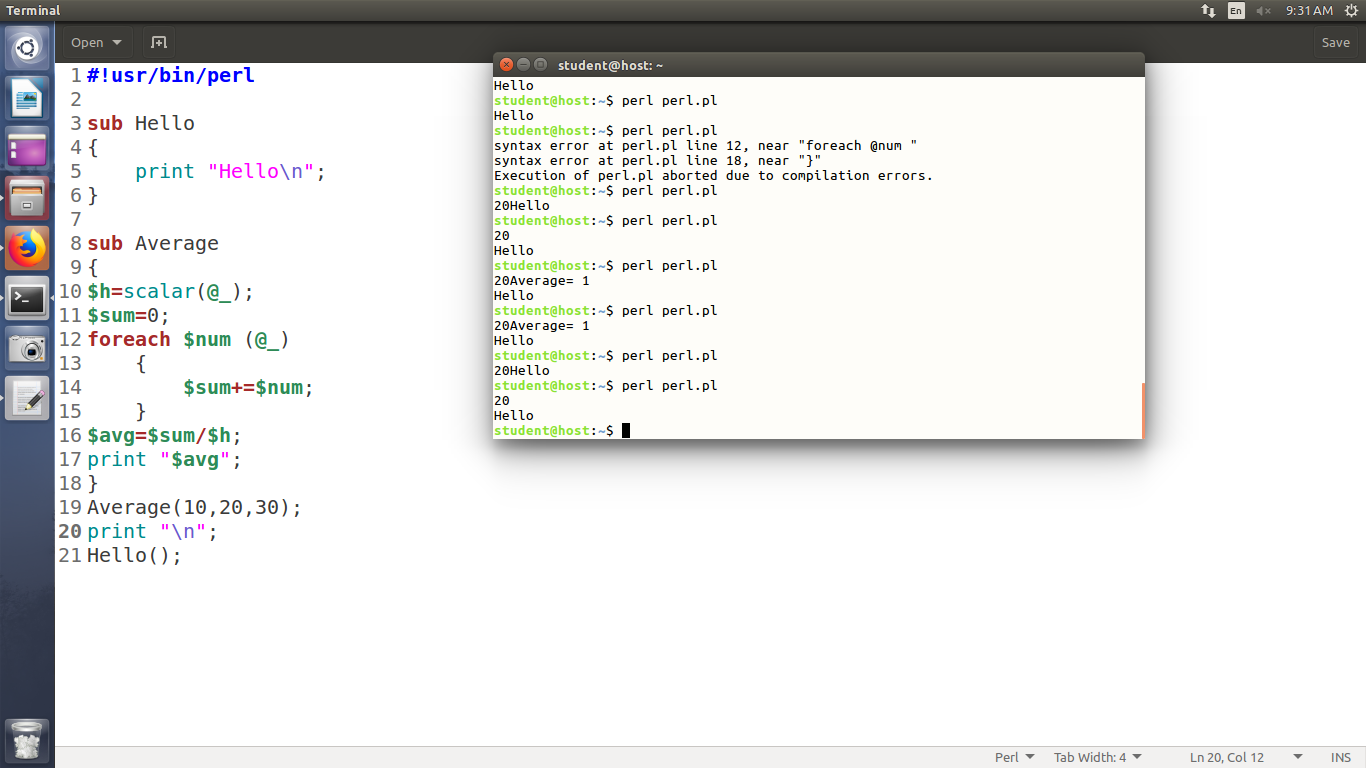
print "$avg";

}

Average(10,20,30);

print "\n";

Hello();



#!usr/bin/perl

sub Hello

{

print "Hello\n";

}

sub Average

{

$h=scalar(@\_);

$sum=0;

foreach $num (@\_)

{

$sum+=$num;

}

$avg=$sum/$h;

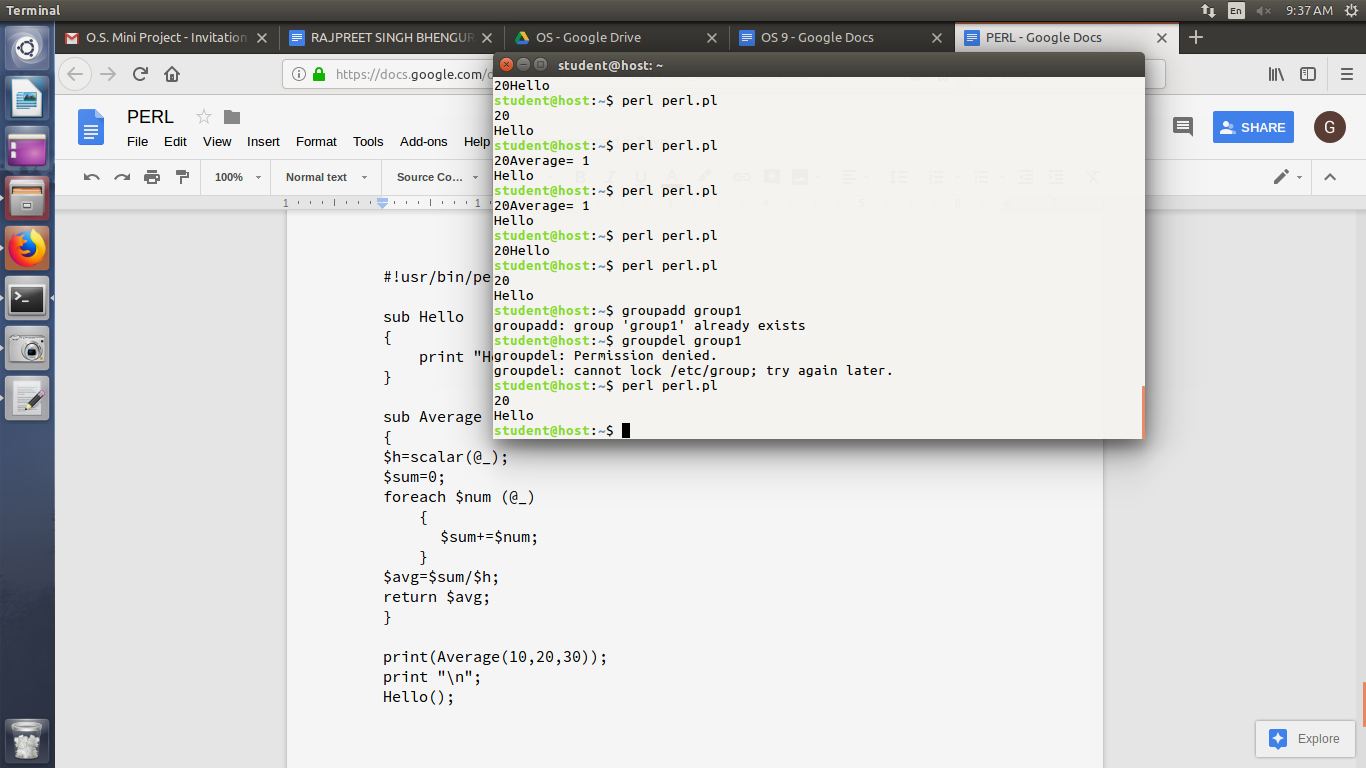
return $avg;

}

print(Average(10,20,30));

print "\n";

Hello();



REPORT GENERATION:

*@ start of the field*

*< left justified*

*> right justified*

*@##### numeric field holder*

*@\* multiline field holder*

#!usr/bin/perl

format EMPLOYEE =

===================================

@<<<<<<<<<<<<<<<<<<<<<< @<<

$name, $age

@#####.##

$salary

===================================

.

select(STDOUT);

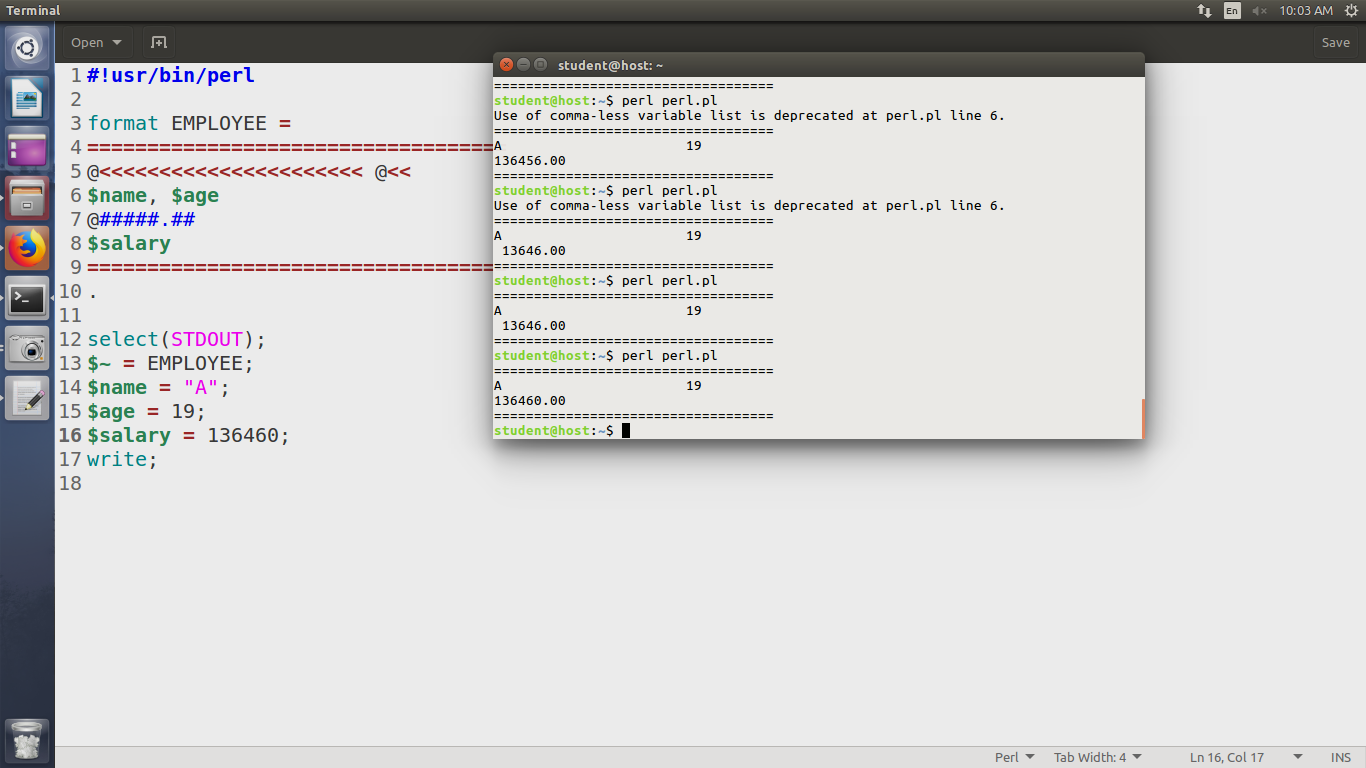
$~ = EMPLOYEE;

$name = "A";

$age = 19;

$salary = 136460;

write;



sub pow

{

print "Enter the base : ";

$x = <STDIN>;

print "Enter the power : ";

$y = <STDIN>;

$i=1;

$pow=1;

while($i<=$y)

{

$pow=$pow\*$x;

$i=$i+1;

}

print "The power is $pow";

}

pow();