https://pastebin.com/raw/Z5Meyptz

Perl chop():

````````````

The Perl chop() function removes last character from a string regardless of what that character is.

#!/usr/bin/perl

$a = "abc";

chop($a);

print "$a\n"

# perl script.pl

ab

e.g:

````

#!/usr/bin/perl

print "Enter a value: ";

chomp($a = <>);

chop($a);

print("Out: ", $a)

# perl 1.pl

Enter command: pwd

OUTPUT: /root

#!/usr/bin/perl

print "Enter command: ";

$a = <>;

chomp($a)

print ("Output: ", `$a`);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Some adv features in PERL:

``````````````````````````

=> use of Pragmas:

```````````````````

\* Its highly recommanded to use pragmas in top of the program. This force you to declare all your variables and causes PERL to complain/warning when it sees a variable which is not declared or misspelled.

\* It also checks the variable scopes in different blocks of a long code.

use strict;

use warnings; OR #!/usr/bin/perl -w

EG:

```

#!/usr/bin/perl

print "$test\n";

# perl script.pl

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

WITH WARNING PRAGMA:

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

#!/usr/bin/perl

use warnings;

print "$test\n";

# perl script.pl

Name "main::test" used only once: possible typo at script.pl line 3.

Use of uninitialized value $test in concatenation (.) or string at script.pl line 3.

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

WITH STRICT PRAGMA:

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

#!/usr/bin/perl

use strict;

print "$test\n";

# perl script.pl

Global symbol "$test" requires explicit package name (did you forget to declare "my $test"?) at script.pl line 4.

Execution of script.pl aborted due to compilation errors.

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

BOTH STRICT & WARNINGS PRAGAMA:

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

#!/usr/bin/perl

use strict;

use warnings;

print "$test\n";

# perl script.pl

Global symbol "$test" requires explicit package name (did you forget to declare "my $test"?) at script.pl line 4.

Execution of script.pl aborted due to compilation errors.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

my - Local variable

````````````````````

- Suppose we have a variable having some value but we want to use a different value of the same variable only in one block of code.

#!/usr/bin/perl

$a = 5;

print "Value: $a\n";

{

$a = 6;

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

}

print "Value of a outside the block is $a\n";

# perl script.pl

Value: 5

Value of a is 6

Value of a outside the block is 6

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

#!/usr/bin/perl

use strict;

use warnings;

$a = 5;

print "Value: $a\n";

{

my $a = 6;

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

}

print "Value of a outside the block is $a\n";

# perl script.pl

Value: 5

Value of a is 6

Value of a outside the block is 5

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

#!/usr/bin/perl

use strict;

use warnings;

my $a = 5;

print "Value: $a\n";

{

my $a = 6;

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

}

print "Value of a outside the block is $a\n";

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_