UNIT - II & III :-

1) Explain load testing on websites?

→ Load testing is meant to test the system by constantly and steadily increasing the load on the system until it reaches the threshold limit. It is a subset of Performance testing.

→ Load testing can be easily done by employing any of the suitable automation tools available in the market.

→ Load testing is also famous by names like volume testing and endurance testing.

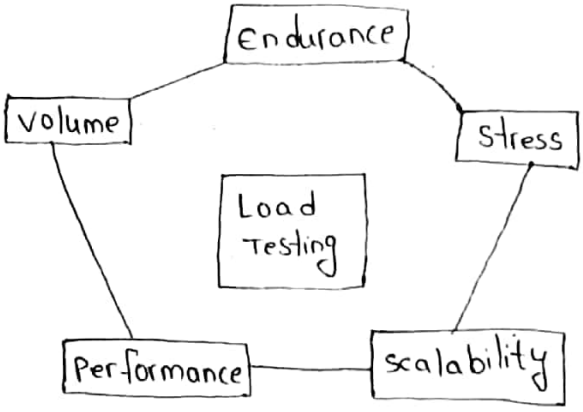


Fig:- web Load Testing.

---

2) What is the importance of Perl warnings?

→ Perl warnings are those in which Perl checks the quality of the code that you have produced.

Mandatory warings highlight problems in the lexical analysis stage.

→ Optional warnings highlight cases of possible anomaly.

```
use warnings;  # it is same as importing "all".
no warnings;  # it is same as unimporting "all".
use warnings::register;
if (warnings::enabled()) {
    warnings::warn("any warning");}
```

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

```
if(warnings::enabled("void"))
{
    warnings::warn("void", "any warning");
}
```

3.) Explain the arguments for Perl Interpreter?

→ Perl Interpreter, it contains or compiles the Prgrm internally into a Parse tree and then executes it immediately. Perl is commonly known as an interpreted language, but this is not strictly true.

→ perl cmnd line arguments stored in the special array called @ARGV. The array @ARGV contains the command-line arguments intended for the script.

→ $#ARGV is generally the no. of arguments minus one, because $ARGV[0] is the first argument, not the program's command name itself.

4.) What are Prefix dereferencer?

→ Using a particular prefix when you dereference a variable, they are called Prefix dereferencer.

1.) $ - Scalar variables
2.) % - Hash variables
3.) @ - Arrays
4.) & - subroutines
5.) Typeglobs - myvar stands for @myvar, %myvar.

5.) Explain "grep" function in Perl?

→ The Perl grep() function is a filter that runs a regular expression on each element of an array and returns only the elements that evalute as true.

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

→ using regular expressions can be extremely powerful and complex.

→ the grep() functions uses the syntax:-

@List = grep(Expression, @array).

→ Using Grep() -function to Return True Expressions.

```
@myNames = ('Jacob', 'Michael', 'Joshua', 'Mathew',
            'Alexander', 'Andrew');
@grepNames = grep(/^A/, @myNames);
```

→ Reversing the Expression in a Grep() Functions.

```
@myNames = ('Jacob', 'Michael', 'Joshua', 'Mathew',
            'Alexander', 'Andrew');
@grepNames = grep(!/^A/, @myNames);
```

6.) How can arrays be tied in Perl?

→ this module provides methods for array-tying classes.

→ See perTie for a list of the functions required in order to tie an array to a package.

→ the basic Tie::Array package provides stub DESTROY and EXTEND methods that do nothing, stub DELETE and EXISTS method that croak() if the delete() or exists() buit-ins are ever called on the tied array, and implementations of PUSH, POP, SHIFT, UNSHIFT, SPLICE and CLEAR in terms of basic FETCH, STORE, FETCHSIZE, STORESIZE.

7.) How many ways can we express string in Perl?

→ Many. For example 'this is a string' can be expressed as:-

→ 'this is a string'

→ "this is a string"

→ qq/this is a string like double-quote string/

→ qq^this is a string like double-quoted string^

→ q/this is a string/.

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

→ q1 this is a string &

→ q(this is a string).

---

8) What is Perl one-liner?

→ Perl-one liners are Powertools that get a lot of work done with a single cmnd. You can use them on Unix command-line.

→ There are many pages about Perl one-liners. this one is only interested in Perl one-liners in the context of regular expression matching, replacing & splitting.

Ex:- `Perl -pi -e 's/you/me/g' file`

---

9) How do you connect to database in Perl?

Step1:- Use the following Perl code to connect to MYSQL and select a database. Replace USERNAME with your username, PASSWORD with your password, and DBNAME with the database name:

```
use DBI;
$myconnection= DBI -> connect("DBI:mysql:DBNAME:
                              localhost",
                "USERNAME","PASSWORD");
```

Step2:- After the code connects to MYSQL and selects the database, you can run SQL queries and perform other operations.

Ex:-

```
$query =$myconnection-> Prepare("SELECT
          lastname FROM employees");
$result = $query -> execute();
```

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

10) Discuss about Last, next and redo commands?

Next:- The Perl next stmnt starts the next iteration of the loop. You can Provide a LABEL with next stmnt where LABEL is the label for a loop. A next stmnt can be used inside a nested loop.

syn:- `next [LABEL];`

Last:- The last cmnd is like the break stmnt in c; It immediately exists the loop in question.

syn:-
```
LINE: while (<STDIN>)
  {
  last LINE if /^$/;
    ...
  }
```

redo:- The redo cmnd restarts the loop block without evaluting the conditional again. You can provide a LABEL with redo stmnt where LABEL ~~is~~ is the label for a loop.

syn:- `redo [LABEL]`

---

11) what is a filehandle and write the syntax to create filehandle in Perl?

→ File handling is the most important part in any prgrmming language. A file handle is an internal perl structure that associates with a file name.

→ Perl file handling is important as it is helpful in-accessing file such as text-files, log-files or configuration -files.

→ It is capable of creating, reading, opening and closing a file.

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

Mode:-		Explanation:-
"<"	-	Read only Mode.
">"	-	creates file
">>"	-	creates, file, Appends to the file.
"+<"	-	Reads & writes but does NOT -create.
"+>"	-	creates file, clears, Reads and writes.
"+>>"	-	creates file, Reads and Appends.

12.) what is typeglob in perl?

→ perl uses a special type called, a typeglob to hold an entire symbol table entry. the type prefix of a typeglob is a * because it represents all types.

syn:-
```
$fh = *STDOUT;
$fh = \*STDOUT;
$fh = *STDOUT{IO};
```

13.) Define array of arrays in perl with an example?

→ In perl, array is a special type of variable. the array is used to store the list of values and each object of the list is termed as an element.

→ Elements can either be a number, string, & any type of scalar data including another variable.

0	1	2	3	4	5	6
40	55	63	17	22	68	89

← Array indices.

Array Length = 7
First index = 0
Last index = 6

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

→ Ex:-
```
@number = (50, 70, 46);
@names = ("Geeks", "For", "Geeks");
```

14.) Define complex data structures in Perl?

→ complex data structures fall in two different broad categories: positional, or list-like and Associative, or key-value pair like, according to how you access its first-level elements.

Ex:- say [[1, 2] [3, 4]], [[5, 6] [7, 8]]]]. deepmap (*.elems);

O/P:- [[1 1 [1 1]] [1 1 [1 1]]]

Ex:-
```
class SortedArray is Array
    {
        method iterator()
        {
            self.sort.iterator
        }
    };
    my @thing := SortedArray.new([3,2,1,4]);
    .say for @thing;  # output: <<1 2 3 4>>
```

15.) List out the file status checks available in Perl?

Refer answer : 11 and 12.

16.) Define eval function & syntax in Perl?

→ The eval() function evalutes or executes an argument. If the argument is an expression, eval() function evalutes the expression. If the argument is one or more Javascript stmnts, eval() executes the statements.

syntax:-
```
eval EXPR
eval BLOCK
```

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

17.) Define ref function & syntax in Perl?

→ A Perl reference is a scalar datatype that holds the location of another value which could be scalar, arrays, or hashes. You can construct the lists containing references to other lists, which can contain references to hashes, and so on.

Syn:-
=
ex:-
=

```
$scalarref = \$foo;
$arrayref  = \@ARGV;
$hashref   = \%ENV;
$coderef   = \&handler;
$globref   = \*foo;
```

or

Syn:-
=

ref EXPR
ref

18.) How can you create a class in Perl?

→ A class is a blue print or template of similar objects.

→ A class is a method for creating an object.

Ex:- phone is an instance of Product class.

Product
- serial: string - name: string - Price: float
+get_serial() +set_serial() +get_name() +set_name() +get_price() +set_price() +to_string()

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

Ex:-
```
sub new
{
    my $class = shift;
    my $self = { _firstName => shift,
                 -lastName => shift,
                 -ssn      => shift,
               };
  Print "First Name is $self->{_firstName}\n";
  Print "LastName is $self->{_lastName}\n";
  bless $self, $class;
  return $self;
```

19.) Explain about Perl constructor?

→ A constructor is a function which uses bless function inside it and also returns a reference to something that has the class name associated with it. It means that constructor is a special kind of function which has two specialities.

1.) It uses bless function.
2.) It returns a reference to something that has the class name associated with it.

20.) Explain about Perl destructor?

Peral automatically calls the destructor method when all the references to an object go out of scope. Destructor methods are of use if the class creates thr- eads or temporary files that need to be cleaned up if the object gets destroyed. Perl contains a special method name, 'DESTROY' for the destructor, that must be used at the time of destructor declaration.

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner

St. Mary's Group of Institutions
CHEBROLU
522 212
Guntur

syntax:-

```
sub DESTROY
{
  # DEFINE Destructors
   my $self = shift;
   Printf "constructor Destroyed : P"
}
```

once this snippet is added to existing file Area.pm.

By:-

R. Kartheek,
CSE-Department,
SMGI - Guntur.

GMKSOFTWARE.BLOGSPOT.COM

Scanned by CamScanner