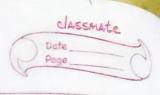
N	sonat Scope of a variable. Différence between them is the enample	
17	Local variable -> Those variables whose so noce to	
	Local variable -> Those variables whose so scope is within the function where it is declared	
	0	
	Eplobal voorfable - These voorfables com be accessed globally in the entire paragram.	
	In the entire paragram.	0 • 5
	Local voorfable	Global variables
0	The local keyword is used to	· No Keyword is used
	declare the local voorfable	
•	I + 93 declared Phoside the	· It can be declared any
	function	where in the program
	0	
	It doesn't perovide data shooting	· It perovede data shaving.
e	These are stored on the	· These variables are stoned
	Stack	in the fixed memorylocation
•		by the compiler.
	Passing parameters Pis necessary	Parameters passing is not necessary
-	grantage value Ps storted if	· Lego is stored by a family
	not initialized	not initialized
	I modification or modified	· If modified in one function that modification will be
	in one function that	visible in whole program.
	doesn't reflect in other	Visited 1 Control proj
	function	
1	0	
1		Andrew and the state of the sta



Example for creating local variable.

function hello () ?

local x = "RVCE"

echo "I am studying in x= \$x"

J. Ma

echo "I am studying in x = \$x"

Output.

I am studying in KVCE I am studying in X =

Example for creating a Global variable.

function LS3() {

X="Linux shell scripting" echo "full form of LS3 is X=\$x"

3

LS3 echo" full form of LS3?x=\$x"

full form of L83 is X=Linux shell southing full form of L33 is X=Linux shell swipting.

Explain all the perogramming constructs in shell perogramming with examples.

Loop constructs are for and while looping constructs allow a program to execute a command or sequence of commands several time

For hoop

The for loop executes a sequence of commands once for each member of a list

Syntan: -

for variable

in a list-of-value

do

command-1

command-2

last - command

Format of the for loop construct For each "terration of the loop, the next member of the list is assigned to the variable given in the for Clause References to that voorlable may be made anywhere In the commands within the do dause. It is easier to read a shell perogramming if the looping constructs are visually clear Because the shell ignores spaces at the beginning of the lines, each section of a command can be indented as it was in the above

format. Each do has to have a corresponding done

at the and of the loop when the command have been enecuted for the last value in the list, the program will execute the new line below done . If there is no line the perogram will end Create a perogram that well move files to another directory. I cat mv. file echo Please type in the diesectory path dead bath for file In memo! memo a memo 3 mv Stele \$path 1. \$file While hoop The other loop construct , the while loop, uses two groups of commands . It will continue enecuting the bequence of commands in the second group, the do as long as the fenal command in the first group the while list, neturn a status of meaning the stevements after the do can be executed. Syntox: while Command\_1 last - command Command -1 last - command done

& cat fele enter name while do gread v echo \$x77 fele done - \$ cat entor. hame echo 'please type in each person's name and then a echo "please end the lest of names with a CTRL-d" while neadx echo \$x >> xfele Cho xfele contains the following names: After the I loop is completed, the paragram enecutes the commad below the done Conditional Constructs: if and case The if command tells the shell program to execute the then sequence of command only if the final command in the of command list is successful The of construct ends with the keyword for Synton: Command -1

last - (ommand

Classmate

Classmate

Command-1:
last-command

fi

She cat search

echo Type in the word and the filename

eneed word file

if greep sword spile

then echo sword is in spile

fi

-> 11 -- then -- else

If The if... then construction can also issue an alternate set of commands with else, when the if command sequence is false when the effection of sequence is false. It has the general format syntom:

If I'm command-1

last-command then command-1

last- (ommand else (ommand-1

last - command

8

I cat search echo Type in the word and the felename. I lead word file grep & word & file The Vhull echo \$ word & san & file echo & word is NOT in stile Case - - esac Systax: -Case word pattern -1) Command-line -1 last - command - line puttorn -2) command - line -1 last - command - line \*) Command 1 last - command

classaute

Example: echo if you have a TTY HHAD type in H420 & cat set term echo if you have a TTY SHID type in 5410 echo ef you have a TTY 5420 type in 5420 nead steam case \$ term Pn 4420) TERM=TY 11 5H10) TERM-85 5H20) TERM =T7 echo not a correct terminal type esac enport TERM echo end of program Boreak - The bereak command unconditionally stop the eneutron of any loop in which it is encounted and goes to the nent command after the done of on esac statemen. go immediatly to the next iteration of a while in loss without eneating the nemarining comment fn boog