

Command Substitution

Command substitution is the mechanism by which the shell performs a given set of commands and then substitutes their output in the place of the commands.

Syntax

```
`command`  
`command1 | command2`  
`command1 ; command2`
```

i.e.

```
#!/bin/sh  
  
DATE=`date`  
echo "Date is $DATE"  
  
USERS=`who | wc -l`  
echo "Logged in user are $USERS"  
  
UP=`date ; uptime`  
echo "Uptime is $UP"
```

Variable Substitution

Variable substitution enables the shell programmer to manipulate the value of a variable based on its state.

Sr.No.	Form & Description
1	\${var} Substitute the value of <i>var</i> .
2	\${var:-word} If <i>var</i> is null or unset, <i>word</i> is substituted for var . The value of <i>var</i> does not change.
3	\${var:=word}

	If <i>var</i> is null or unset, <i>var</i> is set to the value of word .
4	\${var:?message} If <i>var</i> is null or unset, <i>message</i> is printed to standard error. This checks that variables are set correctly.
5	\${var:+word} If <i>var</i> is set, <i>word</i> is substituted for <i>var</i> . The value of <i>var</i> does not change.

i.e.

```
#!/bin/sh

echo ${var:-"Variable is not set"}
echo "1 - Value of var is ${var}"

echo ${var:="Variable is not set"}
echo "2 - Value of var is ${var}"

unset var
echo ${var:+ "This is default value"}
echo "3 - Value of var is $var"

var="10"
echo ${var:+ "This is default value"}
echo "4 - Value of var is $var"

echo ${var:? "Print this message"}
echo "5 - Value of var is ${var}"
```

Input Redirection from file

Command < file