Unix / Linux Shell - The case...esac Statement

You can use multiple **if...elif** statements to perform a multiway branch. However, this is not always the best solution, especially when all of the branches depend on the value of a single variable.

Shell supports **case...esac** statement which handles exactly this situation, and it does so more efficiently than repeated if...elif statements.

Syntax

The basic syntax of the **case...esac** statement is to give an expression to evaluate and to execute several different statements based on the value of the expression.

The interpreter checks each case against the value of the expression until a match is found. If nothing matches, a default condition will be used.

```
case word/pattern in
   pattern1)
      Statement(s) to be executed if pattern1 matches
      ;;
   pattern2)
      Statement(s) to be executed if pattern2 matches
      ;;
   pattern3)
      Statement(s) to be executed if pattern3 matches
      ;;
   *)
     Default condition to be executed
     ;;
esac
```

```
#check input character is digit,vowel or consonant
#!/bin/bash

 echo -n  "Enter  alphanumeric character: "
read ch
case $ch in
     [0-9])echo you choose from 0 to 9
          ;;
     [aeiouAEIOU])echo you choose vowels
          ;;
     [bcdfghjklmnpqrstvwxyz])echo you choose consonants
          ;;
     *)echo combined or special  characters

 esac
```

Unix / Linux Shell - The case...esac Statement

Output:
debasis@LAPTOP-H3N6JCNE:~$ sh caseEx2.sh
Enter alphanumeric character: r
you choose cosonants
debasis@LAPTOP-H3N6JCNE:~$ sh caseEx2.sh
Enter alphanumeric character: 9
you choose from 0 to 9

```
#check whether given input is digits,vowels,consonants and other characters...
#!/bin/bash
ans='y'
while [ $ans = 'y' ]
do
        echo "enter your choice: "
        read ch
        case $ch in
                [0-9])echo digits;;
                [aeiou])echo vowels;;
                [b-d,f-h,j-n,p-t,v-z])echo consonants;;
                *)echo "other ..."
        esac
echo -n "Do U want to continue:?(y/n): "
read ans
done
```

```
#check whether given input is digits,vowels,consonants and other characters...
#!/bin/bash
ans='y'
while [ $ans = 'y' ]
do
        echo "enter your choice: "
        read ch
        case $ch in
                [0-9])echo digits;;
                [aeiou])echo vowels;;
                [b-d,f-h,j-n,p-t,v-z])echo consonants;;
                *)echo "other ..."
        esac
echo -n "Do U want to continue:?(y/n): "
read ans
ans=${ans,,} #convert upper to smaller
done
```

Unix / Linux Shell - The case...esac Statement

```
debasis@LAPTOP-H3N6JCNE:~/student$ chmod 777 sw.sh
debasis@LAPTOP-H3N6JCNE:~/student$ ./sw.sh
enter your choice:
1
digits
Do U want to continue:?(y/n): Y
enter your choice:
b
consonants
Do U want to continue:?(y/n): n
```

```
#MENU DRIVEN CALCULATOR
#!/bin/bash

ans='y'
while [ $ans = 'y' ]
do     echo "1.Addition."
       echo "2.Subtraction."
       echo "3.Multiplication."
       echo "4.Division."
       echo "5.Remainder."
       echo "6.Square root."
       echo "7.power."
       echo "0.To Exit."

       read -p "Enter your choice: " ch

       case $ch in
            1)echo -n "Enter two numbers: "
            read a b
            sum=`echo $a + $b | bc`
            echo "sum= " $sum;;

           2)echo -n "Enter two numbers: "
            read a b
            sub=`expr $a - $b`
            echo $sub;;
                    3)echo -n "Enter two numbers: "
            read a b
            m=`expr $a \* $b`
            echo "Multiplication: " $m;;

           4)echo -n "Enter two numbers: "
            read a b
            div=`echo "scale=2;$a / $b" | bc`
            echo "Division= "$div;;
```

```
            5)echo -n "Enter two numbers: "
             read a b
             rem=`echo "scale=2;$a % $b" |bc`
             echo "Remainder: " $rem;;

             6)echo -n "Enter number: "
             read num
             sq=`echo "scale=2;sqrt($num)" | bc`
             echo $sq;;

             7)echo -n "Enter two values to find power: "
              read a b
              pw=`echo "scale=2;$a ^ $b" | bc`
              echo $pw;;
                        0)exit;;

             *)echo "Invalid choice."

esac

echo -n "Do U want to continue:(y/n): "
read ans
done
echo "Job Over."
Output:
debasis@LAPTOP-H3N6JCNE:~$ sh caseExCal.sh
1.Addition.
2.Subtraction.
3.Multiplication.
4.Division.
5.Remainder.
6.Square root.
7.power.
0.To Exit.
Enter your choice: 7
Enter two values to find power: 25.50 2
650.25

Do U want to continue:(y/n): y
1.Addition.
2.Subtraction.
3.Multiplication.
4.Division.
5.Remainder.
6.Square root.
7.power.
0.To Exit.
Enter your choice: 4
Enter two numbers: 50.50 5
Division= 10.10
Do U want to continue:(y/n): n
```

```
debasis@LAPTOP-H3N6JCNE:~/student$ vi uplow.sh
```

```
#Convert lower to uppr and upper to lower
#!/bin/bash
echo "Enter name in lower letter: "
read name
echo
name=${name^^} #convert lower to upper to lower
echo "After converting from lower to upper: " $name
echo
name=${name,,} #convert upper to lower
echo "after converting upper to lower: " $name
```

```
debasis@LAPTOP-H3N6JCNE:~/student$ ./uplow.sh
Enter name in lower letter:
debasis

After converting from lower to upper:  DEBASIS

after converting upper to lower:  debasis
```