

Curso 1º SMR

Módulo: SOM

Sesión 4: Programación shell: case

¿Qué pretendemos conseguir?

Conocer la sentencia case.

¿Qué necesitamos?

Haber conseguido los objetivos planteados en la sesión 2.
Conocer la cláusula if estudiada en la sesión 3.

Desarrollo de la sesión

case nos va a permitir comprobar una expresión y dependiendo del valor que obtengamos realizar unas cosas u otras.

La sintaxis de case es la siguiente:

```
case expresión in
    patron1) bloque de instrucciones ;;
    patron2) bloque de instrucciones ;;
    .
    .
    .
    *) bloque de instrucciones;;
esac
```

expresión puede ser una variable o constante.

Patron Es el valor que toma en el momento de ejecutar el programa expresion.

Los dos ; que aparecen al final de los bloques de intrucciones van a hacer que no se ejecuten el resto de bloques.

El patrón *), que no es imprescindible, se ejecutará cuando no se cumpla ninguno de los patrones anteriores.

Vamos a poner un ejemplo y lo vamos a traducir a lenguaje natural:

```
variable1='X'
case $variable1 in
    'A')echo "la variable es A";;
    'B')echo "la variable es B";;
    .
    .
    .
```

```

.
'X')echo "la variable es X";;
'Y')echo "la variable es Y";;
'Z')echo "la variable es Z";;
*)echo "probablemente la letra sea minúscula";;
esac

```

Ejercicio1: crea el script primercase.sh que contenga el script anterior. Observa que los puntos debes cambiarlos por las letras del abecedario que faltan en mayúsculas. ¿Qué ocurre con este programa?

Ejercicio2: Quita los dos puntos y coma del bloque de instrucciones de la Y hasta esac. Ejecuta el script ¿Qué ha pasado ahora?

Bien en lenguaje natural el script anterior vendría a decir lo siguiente:

- la variable 1 contiene X.
- Comprueba que contiene la variable1.
- Busca, hasta esac, un patrón que sea igual al contenido de la variable.
- Si lo has encontrado, ejecuta las instrucciones hasta los dos puntos y coma.
- Si no lo encuentras y existe el asterisco, ejecuta el bloque de código del patrón asterisco.

Dentro de los bloques delimitados por ;; pueden introducirse if, case y cualquier otra sentencia de control que necesitemos.

Ejercicio3: En el script anterior elimina la línea de patrón X. Modifica el patrón Y y pon lo siguiente:

```
'X' | 'Y' ) echo "es X o Y";;
```

Ejecuta ahora el programa y describe lo que ocurre.

Podemos hacer que el mismo bloque de código sirva para más de un patrón con |. En el ejemplo anterior se cumplirá el patrón si es X o es Y. Puede utilizarse para más de dos valores.

Ejercicio4: Elabora un script que pida una letra e indique si es mayúscula o minúscula. (nota: los patrones pueden contener metacaracteres [], ?. Repasa la sesión 2 de Linux si no lo recuerdas).

Ejercicio5: Crea el script PIEDRAPAPELTIJERA.sh que pida al usuario un P, piedra, una A, papel y un T, tijera y muestre en pantalla la opción que ha pulsado.

Ejercicio6: Realiza el script áreas. En él se mostrará un menú indicando el área que queremos calcular: cuadrado,

triángulo, círculo. A continuación solicitará los datos pertinentes y mostrará el resultado en pantalla. (área de cuadrado lado x lado, área de triángulo base x altura dividido entre dos, área del círculo pi por radio al cuadrado).

Ejercicio7: Queremos hacer un programa al que se le pase como parámetro posicional un número. El programa debe mostrar en pantalla el nombre del mes. Si no existiera la correspondencia entre el número y el mes debe indicarlo.

Ejercicio8: Realiza un script que pida la categoría profesional de un empleado y las horas que ha trabajado. Los empleados cobrarán según la siguiente tabla:

Cat A → 50 € la hora

Cat B → 40 € la hora

Cat C → 30 € la hora

Cat D → 15 € la hora.

El programa debe mostrar el sueldo de ese empleado.

Ejercicio9: Elabora un programa OPCIONES que solicite una elección entre tres opciones a, b y c, y muestre el mensaje "primera", "segunda", "tercera" o elección incorrecta según el caso. Utiliza la estructura case.

Ejercicio10: Crea el programa MENULINUX.sh. Mostrará las siguientes opciones:

A- Borra la pantalla.

B- Listado del directorio.

C- fecha del sistema

Al elegir el usuario la opción realizará la acción especificada. Ten en cuenta que serán válidas tanto las minúsculas como las mayúsculas.

Ejercicio11: Desarrolla el programa calcula.sh. El programa pedirá dos números y mostrará a continuación un menú para realizar las siguientes operaciones:

A- Divide A entre B

B- Multiplica A entre B

C- Suma A y B

D- Resta A de B.

En la primera opción debes comprobar que B no sea cero. De ser así mostrará un mensaje en pantalla.