Ejercicio 5.1.

Utilizando una variable que contenga el valor entero 365 y otra que guarde el número del día actual del año en curso, realice la misma operación del ejemplo anterior usando cada una de las diversas formas de cálculo comentadas hasta el momento, es decir, utilizando expr, $((...)) y $[...].

```
juanka1995@juanka-hp:~$ dias=365
juanka1995@juanka-hp:~$ dia_actual=`date +%j`
juanka1995@juanka-hp:~$ expr \( $dias - $dia_actual \) / 7
6
```

```
juanka1995@juanka-hp:~$ dias=365
juanka1995@juanka-hp:~$ dia_actual=`date +%j`
juanka1995@juanka-hp:~$ echo $(((($dias-$dia_actual)/7))
6
```

```
juanka1995@juanka-hp:~$ echo $[($dias-$dia_actual)/7]
6
```

Ejercicio 5.2.

Realice las siguientes operaciones para conocer el funcionamiento del operador de incremento como sufijo y como prefijo. Razone el resultado obtenido en cada una de ellas:

```
juanka1995@juanka-hp:~$ v=1
juanka1995@juanka-hp:~$ echo $v
1
juanka1995@juanka-hp:~$ echo $((v++))
1
juanka1995@juanka-hp:~$ echo $v
2
juanka1995@juanka-hp:~$ echo $((++v))
3
juanka1995@juanka-hp:~$ echo $v
3
juanka1995@juanka-hp:~$
```

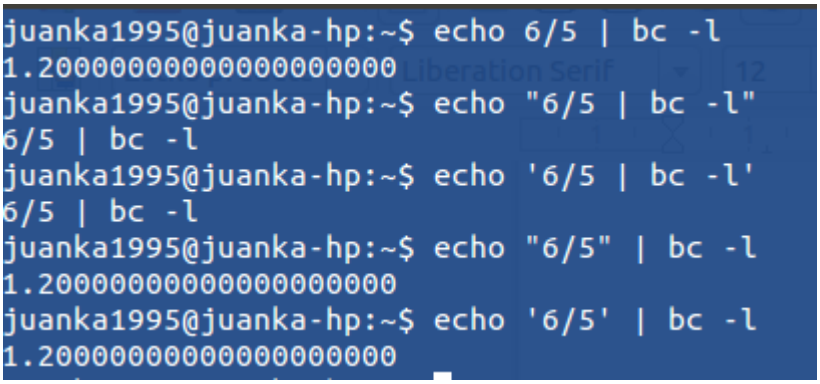
Ejercicio 5.3.

Utilizando el operador de división, ponga un caso concreto donde se aprecie que la asignación abreviada es equivalente a la asignación completa, es decir, que x/=y equivale a x=x/y.

```
juanka1995@juanka-hp:~$ a=6
juanka1995@juanka-hp:~$ b=3
juanka1995@juanka-hp:~$ let a/=b
juanka1995@juanka-hp:~$ echo $a
2
juanka1995@juanka-hp:~$ a=6
juanka1995@juanka-hp:~$ b=3
juanka1995@juanka-hp:~$ let a=a/b
juanka1995@juanka-hp:~$ echo $a
2
```

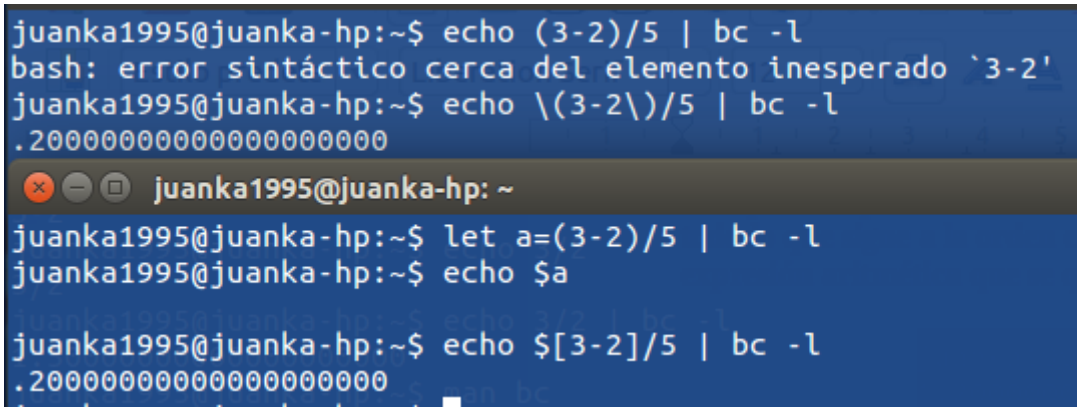
Ejercicio 5.4.

Compruebe qué ocurre si en el ejemplo anterior utiliza comillas dobles o simples para acotar todo lo que sigue a la orden echo. ¿Qué sucede si se acota entre comillas dobles solamente la expresión aritmética que se quiere calcular?, ¿y si se usan comillas simples?



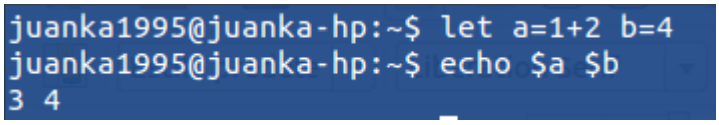
Ejercicio 5.5.

Calcule con decimales el resultado de la expresión aritmética (3-2)/5. Escriba todas las expresiones que haya probado hasta dar con una respuesta válida. Utilizando una solución válida, compruebe qué sucede cuando la expresión aritmética se acota entre comillas dobles; ¿qué ocurre si se usan comillas simples?, ¿y si se ponen apóstrofos inversos?



Ejercicio 5.7.

Con la orden let es posible realizar asignaciones múltiples y utilizar operadores que nosotros no hemos mencionado anteriormente. Ponga un ejemplo de asignación múltiple y, por otra parte, copie en un archivo el orden en el que se evalúan los operadores que admite. Apóyese a través de la ayuda que ofrece help let.



Ejercicio 5.8.

Haciendo uso de las órdenes conocidas hasta el momento, construya un guion que admita dos parámetros, que compare por separado si el primer parámetro que se le pasa es igual al segundo, o es menor, o es mayor, y que informe tanto del valor de cada uno de los parámetros como del resultado de cada una de las evaluaciones mostrando un 0 o un 1 según corresponda.

Ejercicio_5_8

```
#!/bin/bash
echo -e "0 es falso y 1 es verdadero\n"
echo "El numero $1 es MAYOR que $2 = $[$1>$2]"
echo "El numero $1 es MENOR que $2 = $[$1<$2]"
echo "El numero $1 es IGUAL que $2 = $[$1==$2]"
```

```
juanka1995@juanka-hp:~$ ./Ejercicio_5_8 14 14
0 es falso y 1 es verdadero

El numero 14 es MAYOR que 14 = 0
El numero 14 es MENOR que 14 = 0
El numero 14 es IGUAL que 14 = 1
```

Ejercicio 5.9.

Usando test, construya un guion que admita como parámetro un nombre de archivo y realice las siguientes acciones: asignar a una variable el resultado de comprobar si el archivo dado como parámetro es plano y tiene permiso de ejecución sobre él; asignar a otra variable el resultado de comprobar si el archivo es un enlace simbólico; mostrar el valor de las dos variables anteriores con un mensaje que aclare su significado. Pruebe el guion ejecutándolo con /bin/cat y también con /bin/rnano.

Ejercicio_5_9

```
1 #!/bin/bash
2 if [ $# -eq 1 ]
3         then
4                 plano=`test -f $1 && test -x $1;echo $?`
5                 enlace=`test -h $1;echo $?`
6         echo -e "$plano . Si muestra 0 es que es un fichero plano y tiene permisos de ejecucion"
7         echo -e "$enlace . Si muestra 0 es que es un enlace simbolico"
8 fi
```

Ejercicio 5.11.

Responda a los siguientes apartados:

1. **Razone qué hace la siguiente orden: if test -f ./sesion5.pdf ; then printf “El archivo ./sesion5.pdf existe\n”; fi**
2. **Añada los cambios necesarios en la orden anterior para que también muestre un mensaje de aviso en caso de no existir el archivo. (Recuerde que, para escribir de forma legible una orden que ocupe más de una línea, puede utilizar el carácter “\” como final de cada línea que no sea la última.)**
3. **Sobre la solución anterior, añada un bloque elif para que, cuando no exista el archivo ./sesion5.pdf, compruebe si el archivo /bin es un directorio. Ponga los mensajes adecuados para conocer el resultado en cada caso posible.**
4. **Usando como base la solución del apartado anterior, construya un guion que sea capaz de hacer lo mismo pero admitiendo como parámetros la ruta relativa del primer archivo a buscar y la ruta absoluta del segundo. Pruébelo con los dos archivos del apartado anterior.**

ejercicio

```
#!/bin/bash

if [ $# == 2 ]; then

        if test -f $1 ; then
                printf "El archivo $1 existe\n"
        elif test -d ./$2 ; then
                printf "El archivo $1 no existe\n"
                printf "El directorio $2 si existe.\n"
        else
                printf "El directorio $2 no existe.\n"
        fi

else

        printf "No has introducido la cantidad de parametros correctos\n";

fi
```

Ejercicio 5.12.

Construya un guion que admita como argumento el nombre de un archivo o directorio y que permita saber si somos el propietario del archivo y si tenemos permiso de lectura sobre él.

Ejercicio_5_12

```
#!/bin/bash

if [ $# == 1 ]; then

        if test -d $1 || test -f $1 ; then
                #Comprueba si el usuario conectado es el propietario del fichero/directorio introducido
                if test -O $1  ; then
                        echo -e "Eres propietario del fichero o directorio $1"
                        #Comprueba si el propietario del fichero/directorio tiene permisos de lectura sobre este
                        if test -r $1; then
                                echo "Tienes permisos de lectura sobre $1"
                        else
                                echo "No tienes permisos de lectura sobre $1"
                        fi
                else
                        echo -e "No eres propietario del fichero o directorio $1"
                fi
        else
                echo -e "Introduce un fichero o directorio valido"
        fi
else
        echo -e "Introduce como maximo un parametro"
fi
```

Ejercicio 5.13.

Escriba un guion que calcule si el número de días que faltan hasta fin de año es múltiplo de cinco o no, y que comunique el resultado de la evaluación. Modifique el guion anterior para que admita la opción -h de manera que, al ejecutarlo con esa opción, muestre información de para qué sirve el guion y cómo debe ejecutarse.

El siguiente guion de ejemplo se puede utilizar para borrar el archivo temporal que se le dé como argumento. Si rm devuelve 0, se muestra el mensaje de confirmación del borrado; en caso contrario, se muestra el código de error. Como se puede apreciar, hemos utilizado la variable $LINENO que indica la línea actualmente en ejecución dentro del guion.

```bash
#!/bin/bash

let dias_restantes=365-`date +%j`

if [ "$1" == "-h" ]; then
        echo -e "Intrucciones del guion:\n"
        echo "Este guion va a calcular los dias que quedan para fin de año y va a mostrar un mensaje diciendo si el numero de dias es multiplo de 5 o no."
        echo "Si ejecutamos el nombre del fichero con la opcion -h accederemos a la ayuda."

elif [ `echo $[dias_restantes%5]` == 0 ]; then
        echo "Quedan $dias_restantes dias para fin de año y es multiplo de 5"
else
        echo "Quedan $dias_restantes dias para fin de año pero no es multiplo de 5"
fi
```