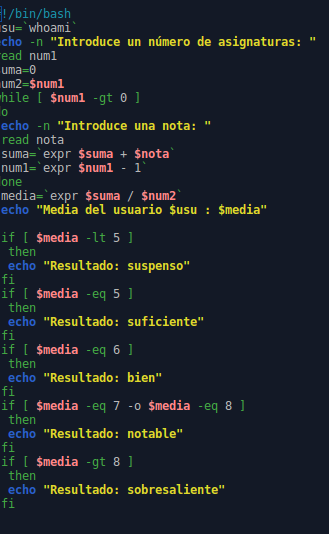
**1. Preguntar al usuario por un número de asignaturas. Luego preguntar la nota de cada asignatura. Mostrar la siguiente información:**

**Media del usuario *nombre*: *media***

**Resultado: *nota***

**Donde *nombre* es el nombre del usuario, *media* es la nota media calculada y *nota* es la calificación textual (ej. aprobado o notable, etc…)**

#!/bin/bash

usu=`whoami`

echo -n "Introduce un número de asignaturas: "

read num1

suma=0

num2=$num1

while [ $num1 -gt 0 ]

do

echo -n "Introduce una nota: "

read nota

suma=`expr $suma + $nota`

num1=`expr $num1 - 1`

done

media=`expr $suma / $num2`

echo "Media del usuario $usu : $media"

if [ $media -lt 5 ]

then

echo "Resultado: suspenso"

fi

if [ $media -eq 5 ]

then

echo "Resultado: suficiente"

fi

if [ $media -eq 6 ]

then

echo "Resultado: bien"

fi

if [ $media -eq 7 -o $media -eq 8 ]

then

echo "Resultado: notable"

fi

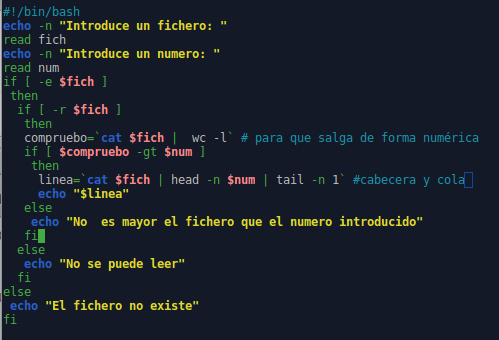
if [ $media -gt 8 ]

then

echo "Resultado: sobresaliente"

fi

**2. Preguntar al usuario por un fichero. Preguntar al usuario por un número. Comprobar que el fichero existe y se puede leer. Comprobar que tiene más líneas que el número indicado por el usuario. Si todo es correcto mostrar la línea correspondiente al número introducido por el usuario.**



#!/bin/bash

echo -n "Introduce un fichero: "

read fich

echo -n "Introduce un numero: "

read num

if [ -e $fich ]

then

if [ -r $fich ]

then

compruebo=`cat $fich | wc -l` # para que salga de forma numérica

if [ $compruebo -gt $num ]

then

linea=`cat $fich | head -n $num | tail -n 1` #cabecera y cola

echo "$linea"

else

echo "No es mayor el fichero que el numero introducido"

fi

else

echo "No se puede leer"

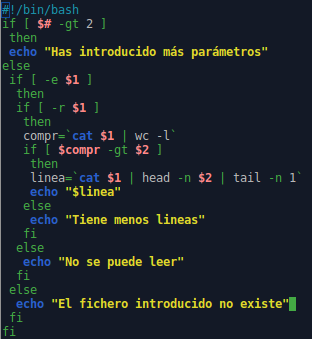
fi

else

echo "El fichero no existe"

fi

**3. Repetir el ejercicio 2. pero aceptando los valores como parámetros. Comprobar que el número de parámetros es el correcto.**

#!/bin/bash

if [ $# -gt 2 ]

then

echo "Has introducido más parámetros"

else

if [ -e $1 ]

then

if [ -r $1 ]

then

compr=`cat $1 | wc -l`

if [ $compr -gt $2 ]

then

linea=`cat $1 | head -n $2 | tail -n 1`

echo "$linea"

else

echo "Tiene menos lineas"

fi

else

echo "No se puede leer"

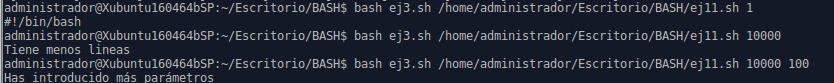
fi

else

echo "El fichero introducido no existe"

fi

fi



**4. Crear un fichero llamado canciones.txt con la siguiente estructura: Autor;Canción**

**Mostrar al usuario el siguiente menú:**

**1. Autor**

**2. Canción**

**3. Buscar**

**4. Salir**

**Si elige la opción 1, el programa deberá mostrar todos los autores.**

**Si elige la 2, el programa deberá mostrar todas las canciones.**

**Si elige la 3, el programa deberá preguntar por Autor y Canción. En caso de elegir Autor, preguntar cuál y mostrar todas sus canciones. En caso de elegir Canción, preguntar cuál y mostrar su autor.**

**El programa deberá mostrar el menú hasta que se elija la opción 4, que terminará el script.**

#!/bin/bash

echo "MENUU"

echo "Opción 1: Autor"

echo "Opción 2: Cancion"

echo "Opción 3: Buscar"

echo "Opción 4: Salir"

echo -n "Selecciona una opción introduciendo el número correspondiente:"

read opt

while [ $opt != "4" ]

do

if [ $opt = "1" ]

then

echo "Los autores son: "

cat /home/administrador/Escritorio/BASH/canciones.txt | cut -d ";" -f "1"

fi

if [ $opt = "2" ]

then

echo "Las canciones son: "

cat /home/administrador/Escritorio/BASH/canciones.txt | cut -d ";" -f "2"

fi

if [ $opt = "3" ]

then

echo -n "Selecciona autor introduciendo el numero 1 y cancion introduciendo el numero 2: "

read num

if [ $num = "1" ]

then

echo -n "Introduce el nombre del autor en minus: "

read autor

echo "Estas son sus canciones: "

cat /home/administrador/Escritorio/BASH/canciones.txt | grep -e "$autor" | cut -d ";" -f "2"

else

echo -n "Introduce la cancion en minus: "

read canc

echo "Este es el autor: "

cat /home/administrador/Escritorio/BASH/canciones.txt | grep -e "$canc" | cut -d ";" -f "1"

fi

fi

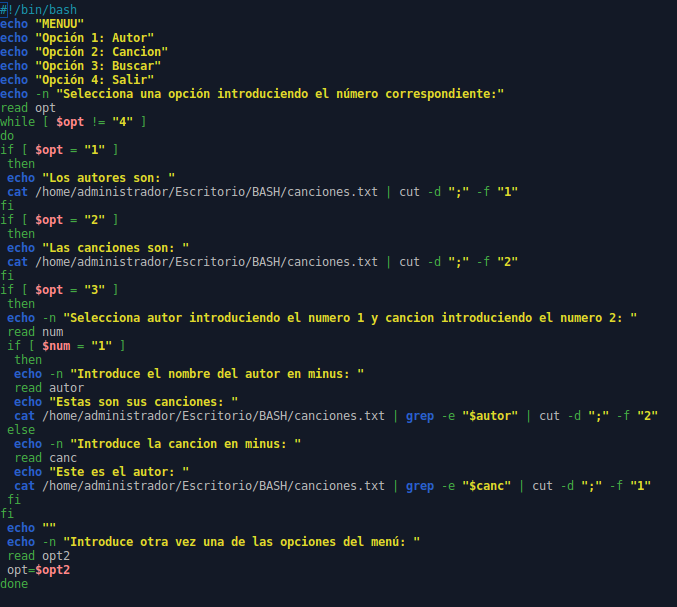
echo ""

echo -n "Introduce otra vez una de las opciones del menú: "

read opt2

opt=$opt2

done



**PARA MOSTRAR EL MENÚ TODO EL RATO O LO VUELVES A METER AL FINAL DEL WHILE O OTRA OPCION MEJOR ES DEFINIR LA VARIABLE OPT=0 PARA QUE ENTRE EN EL WHILE LA PRIMERA VEZ Y YA LUEGO Q TE MUESTRE TODOS LOS ECHO DENTRO DEL WHILE.**

**5. Crear un script que reciba como parámetro el nombre de un programa y que muestre el número de PID del proceso correspondiente en el caso de estarse ejecutando.**

<https://ubuntulife.wordpress.com/2009/05/17/pidof-conocer-el-pid-de-un-proceso/>



#!/bin/bash

ps -ef | grep -e “$1” | cut -d " " -f "2"

6. Programa que pregunte al usuario por su nombre y las horas trabajadas por cada día de la semana. Si un día trabaja más de 8 horas, las horas restantes se consideran horas extras. Mostrar la siguiente información:

*Nombre*

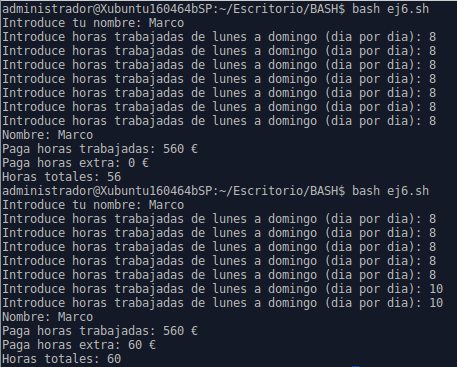
Paga horas trabajadas: ##

Paga horas extras: ##

Sueldo: ##

Horas totales: ##

Donde *Nombre* es el nombre del usuario, la paga de las horas trabajadas son las horas normales que se cobran a 10€ la hora. La paga de las horas extras es el rédito correspondiente a las horas extras contando que estas se cobran a 15€ la hora. El sueldo es la suma de ambas pagas y las horas totales es la suma de las horas normales más las horas extras.



#!/bin/bash

echo -n "Introduce tu nombre: "

read nom

sem=7

trab=0

extra=0

while [ $sem -gt 0 ]

do

echo -n "Introduce horas trabajadas de lunes a domingo (dia por dia): "

read hor

if [ $hor -le 8 ]

then

trab=`expr $trab + $hor`

else

otr=`expr $hor - 8`

extra=`expr $extra + $otr`

trab=`expr $trab + 8`

fi

sem=`expr $sem - 1`

done

total=`expr $trab + $extra`

trab=`expr $trab \\* 10`

extra=`expr $extra \\* 15`

echo "Nombre: $nom"

echo "Paga horas trabajadas: $trab €"

echo "Paga horas extra: $extra €"

echo "Horas totales: $total"

