

MÒDUL 0374 – Administració avançada de sistemes operatius

Administració de sistemes operatius

Activitat pràctica 1
Bash scripting

RA7: Utilitza llenguatges de guions en sistemes operatius, descrivint la seva aplicació i administrant serveis del sistema operatiu.

Observacions

Responen en un color diferenciador.

L'objectiu d'aquesta activitat és la de definir scripts en bash (Linux) que permetin automatitzar i operar sistemes Linux de forma de desatesa, controlant els errors que es produeixin. El codi cal que estigui documentat de forma correcta.

Els punts que es treballaran en aquesta activitat són:

- Variables d'entorn
- Pas de paràmetres a funcions
- Estructures iteratives
- Lectura i escriptura de/a fitxers

RECOMANABLE: Per cada exercici, cal posar el script generat i l'execució del mateix on es vegi el resultat. Servirà d'apunts per a vosaltres.

Generar un fitxer .sh per cada exercici de nom exerciciX.sh on la X és el número d'exercici.

Variables d'entorn

1.	Substituint nom_d'usuari pel valor de la variable \$USER, escriviu una ordre que mostri la frase següent: "El \$USER: nom d'usuari està llegint el llibre " _el llibre que estigueu llegint o que hàgiu llegit_"
El script:	

GNU nano 7.2

exercici1.sh

```
##El meu primer script
#Substitueix nom_usuari pel valor de la variable $USER
#Daniel Roblas
#!/bin/bash

echo "El " $USER "esta lleguint el llibre Harry Potter"
```

La comprovació:

```
isard@ubuntu-24:~$ ./exercici1.sh
El isard esta lleguint el llibre Harry Potter
isard@ubuntu-24:~$
```

2.

Creeu un un script que mostri:

Hola món, son les %hora% del dia %DiaMesAny% i soc l'usuari %nomusuari%

Script:

GNU nano 7.2

exercici2.sh

```
##Daniel Roblas
#Script que mostra el següent text:
#!/bin/bash

hora=$(date +%H:%M)

dia=$(date +%d/%m/%Y)

echo "Hola món, son les" $hora "del dia" $dia "i soc l'usuari" $USER
```

Comprovació:

```
isard@ubuntu-24:~$ ./exercici2.sh
Hola món, son les 10:17 del dia 08/01/2026 i soc l'usuari isard
isard@ubuntu-24:~$
```

Creeu un directori ~/scripts per guardar els vostres programes, moveu el fitxer generat al punt anterior al directori ~/scripts i afegiu el directori al contingut de la variable PATH amb la següent comanda:

3.

```
mkdir ~/scripts
mv ~/llibre.sh ~/scripts
export PATH="$PATH:~/scripts"
```

Cal provar que ara l'execució de l'script funciona sense haver d'accedir al directori.

Script:

```
GNU nano 7.2
#Crear un directori /scripts i moure un fitxer per afegir-li despres un path
#Daniel Roblas

mkdir ~/scripts
mv exercici2.sh ~/scripts

export PATH="$PATH:$HOME/scripts"
```

Comprovació (l'error es perquè el directori scripts ja l'havia creat):

```
isard@ubuntu-24:~$ ls
Baixades  Documents  Escriptori  exercici1.sh  exercici2.sh  exercici3.sh  Imatges
isard@ubuntu-24:~$ ./exercici3.sh
mkdir: no s'ha pogut crear el directori «/home/isard/scripts»: El fitxer ja existeix
isard@ubuntu-24:~$ ls
Baixades  Documents  Escriptori  exercici1.sh  exercici3.sh  Imatges  Música  Plantilles
isard@ubuntu-24:~$ cd /scripts
bash: cd: /scripts: El fitxer o directori no existeix
isard@ubuntu-24:~$ cd ../isard/scripts
isard@ubuntu-24:~/scripts$ ls
exercici2.sh
isard@ubuntu-24:~/scripts$
```

Execució del script '[exercici2.sh](#)' sense estar en la carpeta /scripts:

```
isard@ubuntu-24:~$ pwd
/home/isard
isard@ubuntu-24:~$ ls
Baixades  Documents  Escriptori  exercici1.sh  exercici3.sh  Imatges
isard@ubuntu-24:~$ . exercici2.sh
Hola món, son les 10:02 del dia 09/01/2026 i soc l'usuari isard
isard@ubuntu-24:~$
```

Pas de paràmetres

És molt habitual que els scripts rebin paràmetres que utilitzaran per a aplicar les diferents accions sobre els paràmetres rebuts.

4.	<p>Fer un script que rebí 3 paràmetres a l'hora d'executar: blau , verd, vermell La sortida del programa ha de ser la següent similar a la següent: Nom de l'script: ./exercici3.sh Valor del primer paràmetre de l'script: blau Valor del segon paràmetre de l'script: verd Valor del tercer paràmetre de l'script: vermell</p>
----	--

Scripts:

```
GNU nano 7.2 e)
#Aquest script printa per pantalla els arguments posats a l'hora d'executar l'escript
#Daniel Roblas
#!/bin/bash

echo "Nom del script" $0
echo "El primer parametre de l'escript es" $1
echo "El segon parametre de l'escript es" $2
echo "El tercer parametre de l'escript es" $3
```

Comprovació:

```
isard@ubuntu-24:~$ ./exercici4.sh blau verd vermell
Nom del script ./exercici4.sh
El primer parametre de l'escript es blau
El segon parametre de l'escript es verd
El tercer parametre de l'escript es vermell
isard@ubuntu-24:~$
```

5.	<p>Fer un script que us demani el vostre nom al terminal , l'imprimeixi per pantalla , generant la sortida següent:</p> <p>Com et dius: ? Carles Hola Carles</p>
----	--

Script:

GNU nano 7.2

```
#Aquest script pregunta el teu nom i respon amb ell
#Daniel Roblas

#!/bin/bash
echo "Com et dius?"
read nom

echo "Hola" $nom
```

Comprovació:

```
isard@ubuntu-24:~$ nano exercici5.sh
isard@ubuntu-24:~$ . exercici5.sh
Com et dius?
Daniel
Hola Daniel
isard@ubuntu-24:~$
```

6. Fer un script que demani 2 nombres i mostri la seva suma, resta, multiplicació i divisió.

Script:

```
GNU nano 7.2 exercici6.sh *
#Aquest script demana dos nombres i printa la seva suma, resta, multiplicació i divisió
#Daniel Roblas

read -p "Introdueix un numero ..." N1
read -p "Introdueix un altre numero" N2

let "N3 = N1 + N2"
let "N4 = N1 - N2"
let "N5 = N1 * N2"
let "N6 = N1 / N2"

echo "La suma del primer numero mes la del segon es igual a" $N3
echo "La resta del primer numero menys la del segon es igual a" $N4
echo "La multiplicació del primer numero per el segon es igual a" $N5
echo "La divisió del primer numero entre el segon es igual a" $N6
```

Comrpovació:

```
isard@ubuntu-24:~$ nano exercici6.sh
isard@ubuntu-24:~$ . exercici6.sh
Introdueix un numero ...20
Introdueix un altre numero2
La suma del primer numero mes la del segon es igual a 22
La resta del primer numero menys la del segon es igual a 18
La multiplicació del primer numero per el segon es igual a 40
La divisió del primer numero entre el segon es igual a 10
isard@ubuntu-24:~$
```

7.

Fer un script que indiqui si un nom donat com a paràmetre és un fitxer regular. L'script ha de controlar que se li passi només un paràmetre. Si no és així ha de mostrar un missatge d'error a l'usuari que li indiqui com cal executar el script, i retornar un codi d'error 1.

Script:

```
GNU nano 7.2 exercici7.sh
#Aquest script indica si el nom donat com a parametre es un archiu regular
#Daniel Roblas

#!/bin/bash

test -f $1
N=$?

if [ $# -ne 1 ]; then
echo "Error:Has de posar un parametre, torna a executar l'escript nomes amb un p>
elif [ $N -eq 0 ]; then
echo "El fitxer existeix!!!"
else
echo "El fitxer no existeix :("
fi
```

Comrpovació:

```
isard@ubuntu-24:~$ nano exercici7.sh
isard@ubuntu-24:~$ ./exercici7.sh
Error:Has de posar un parametre, torna a executar l'escript nomes amb un parametre
isard@ubuntu-24:~$ ./exercici7.sh hola hola
Error:Has de posar un parametre, torna a executar l'escript nomes amb un parametre
isard@ubuntu-24:~$ ./exercici7.sh hola
El fitxer existeix!!!
isard@ubuntu-24:~$ ./exercici7.sh hol
El fitxer no existeix :(
isard@ubuntu-24:~$
```

8.

Fer un script que rebí un argument de manera que:

- Si és un directori mostri el missatge "El contingut del directori <nom_directori> és:" i llisti el seu contingut.
- Si l'argument no és cap directori ha de donar un missatge informatiu.

L'script ha de controlar que se li passi només un paràmetre. Si no és així ha de mostrar un missatge d'error a l'usuari i que li indiqui com cal executar el script.

Script:

```
GNU nano 7.2 exercici8.sh
#El script rep un argument, depenen de si es un directori o no fara una cosa difer
#Daniel Roblas

#!/bin/bash

test -d $1
n=$?

if [ $# -ne 1 ]; then
echo "Has de posar nomes un argument"
exit
elif [ $n -eq 0 ]; then
echo "El contingut del directori" $1 "és:"
ls $1
else
echo "El directori no existeix o no s'ha trobat"
fi
```

Comprovació:

```
isard@ubuntu-24:~$ nano exercici8.sh
isard@ubuntu-24:~$ ./exercici8.sh 123
El directori no existeix o no s'ha trobat
isard@ubuntu-24:~$ ./exercici8.sh
Has de posar nomes un argument
isard@ubuntu-24:~$ ./exercici8.sh 1 2
Has de posar nomes un argument
isard@ubuntu-24:~$ ./exercici8.sh scripts
El contingut del directori scripts és:
exercici2.sh
isard@ubuntu-24:~$
```

Estructures de control

El llenguatge bash disposa de diferents estructures de control que ens permeten controlar el fluxe i les iteracions de les execucions.

- | | |
|----|---|
| 9. | Fer un script que demani el valor d'una nota (un nombre enter) i ens digui si la nota és una D (0, 1, 2), una C- (3, 4), una C+ (5, 6), una B (7, 8) o una A (9,10). Cal comprovar que els valors que s'informen són entre 0 i 10, sinó donar un error. |
|----|---|

Utilitzar alguna estructura de tipus **if-else**

Script:


```

isard@ubuntu-24: ~
GNU nano 7.2 exercici9.sh *
#!/bin/bash

#Aquest script pregunta la nota de un examen per a fer la conversio de el numero a una lletra
#Daniel Roblas

echo "Escriu la teva nota del 0 al 10 ..."
read n1

if [ $n1 -lt 0 ] || [ $n1 -gt 10 ]; then
echo "Error: No es un numero del 0 al 10, torna a executar l'escript"
elif [ $n1 -ge 0 ] && [ $n1 -le 2 ]; then
echo "La teva nota es una 'D'"
elif [ $n1 -ge 3 ] && [ $n1 -le 4 ]; then
echo "La teva nota es una '-C'"
elif [ $n1 -ge 5 ] && [ $n1 -le 6 ]; then
echo "La teva nota es una '+C'"
elif [ $n1 -ge 7 ] && [ $n1 -le 8 ]; then
echo "La teva nota es una 'A'"
elif [ $n1 -ge 9 ] && [ $n1 -le 10 ]; then
echo "La teva nota es una 'S'"
fi

```

Comprovació:

```

0374 RA7 Scripting bash isardVDI.com
10 de gen. 22:56
isard@ubuntu-24: ~
isard@ubuntu-24:~$ nano exercici9.sh
isard@ubuntu-24:~$ ./exercici9.sh
Escriu la teva nota del 0 al 10 ...
100
Error: No es un numero del 0 al 10, torna a executar l'escript
isard@ubuntu-24:~$ ./exercici9.sh
Escriu la teva nota del 0 al 10 ...
87
Error: No es un numero del 0 al 10, torna a executar l'escript
isard@ubuntu-24:~$ ./exercici9.sh
Escriu la teva nota del 0 al 10 ...
2
La teva nota es una 'D'
isard@ubuntu-24:~$ ./exercici9.sh
Escriu la teva nota del 0 al 10 ...
5
La teva nota es una '+C'
isard@ubuntu-24:~$ ./exercici9.sh
Escriu la teva nota del 0 al 10 ...
8
La teva nota es una 'A'
isard@ubuntu-24:~$ ./exercici9.sh
Escriu la teva nota del 0 al 10 ...
9
La teva nota es una 'S'
isard@ubuntu-24:~$

```

10. Fer el mateix script que implementat amb una escala if-else, però ara utilitzant una estructura **case**

Script:

```
GNU nano 7.2 exercici10.sh
#!/bin/bash

#Aquest script pregunta la nota de un examen per a fer la conversio de el numero a una lletra
#Daniel Roblas

echo "Escriu la teva nota del 0 al 10 ..."
read n1

case $n1 in
    [1-2]) echo "La teva nota es : 'D'";;
    [3-4]) echo "La teva nota es : '-C'";;
    [5-6]) echo "La teva nota es : '+C'";;
    [7-8]) echo "La teva nora es : 'B'";;
    9|10) echo "La teva nota es : 'A'";;
    *) echo "Error: Tens que posar un numero enter de entre el 0 i el 10"
esac
```

Comprovació:

```
0374 RA7 Scripting bash

isard@ubuntu-24:~$ nano exercici10.sh
isard@ubuntu-24:~$ ./exercici10.sh
Escriu la teva nota del 0 al 10 ...
a
Error: Tens que posar un numero enter de entre el 0 i el 10
isard@ubuntu-24:~$ ./exercici10.sh
Escriu la teva nota del 0 al 10 ...
12
Error: Tens que posar un numero enter de entre el 0 i el 10
isard@ubuntu-24:~$ ./exercici10.sh
Escriu la teva nota del 0 al 10 ...
100
Error: Tens que posar un numero enter de entre el 0 i el 10
isard@ubuntu-24:~$ ./exercici10.sh
Escriu la teva nota del 0 al 10 ...
2
La teva nota es : 'D'
isard@ubuntu-24:~$ ./exercici10.sh
Escriu la teva nota del 0 al 10 ...
6
La teva nota es : '+C'
isard@ubuntu-24:~$ ./exercici10.sh
Escriu la teva nota del 0 al 10 ...
10
La teva nota es : 'A'
isard@ubuntu-24:~$
```

Scripts de control de sistema

11.	Fer un script que calculi l'ocupació de disc de cada directori de /home. Podeu obtenir la llista d'aquests directoris executant l'ordre ls dins del directori /home. Per calcular l'ocupació del directori podeu utilitzar l'ordre du (disk usage).
Script:	

```
isard@exe
GNU nano 7.2
#!/bin/bash

# Aquest script calcula l'espai que ocupa cada directori de la home del usuari
# Daniel Roblas

# Aquí indico que de manera visual, es mostrara l'espai de cada directori de la home
echo ""
echo "-----"
echo " Ara es mostraran l'espai que ocupa cada directori de la teva home:"
echo "-----"
echo ""

# Ara utilitz el for x in a per dir que per cada directori de la home, executi la comanda
# du de manera que surti de manera llegible amb Gb, Mg,... I que surti de manera resumida
for dir in /home/*/; do
du -sh "$dir"
done

# Aquet echo es perquè la comanda es veigui mes ordenada i visual
echo ""
```

Comprovació:

```
0374 RA7 Scripting bash

isard@ubuntu-24:~$ nano exercici11.sh
isard@ubuntu-24:~$ ./exercici11.sh

-----
Ara es mostraran l'espai que ocupa cada directori de la teva home:
-----

4,0K    /home/hola/
119M    /home/isard/

isard@ubuntu-24:~$ cd /home
isard@ubuntu-24:/home$ ls
hola  isard
isard@ubuntu-24:/home$
```

12.	<p>Fer el mateix script, però rebent per paràmetre el nom dels directoris dels quals volem conèixer l'espai que ocupen al disc.</p> <p>Cal comprovar que el directori facilitat és correcte i existeix.</p>
Script:	

```
GNU nano 7.2
#!/bin/bash

# Aquest script agafa el paràmetre indicat a la hora de executar l'escript i si
# es un directori de la home, mostrara amb la comanda du, l'espai utilitzat en
# el disc
# Daniel Roblas

# Avisem a el usuari de manera clara de com s'utilitzara el script
echo ""
echo "-----"
echo " Ara es mostrara l'espai que ocupa el directori escrit com a paràmetre en el disc:"
echo "-----"
echo "(directori utilitzat = $1)"
echo ""

# Ara farem un test per veure si el paràmetre escrit es un directori o no, i si
# no es, retornarem error junt amb els possibles directoris que pot utilitzar
# el usuari com a paràmetre
test -d "/home/$1"
comprovacio=$?
if [ $comprovacio -ne 0 ]; then
echo ""
echo "Error: El directori senyalat com a paràmetre no s'ha trobat o no existeix"
echo ""
echo "Ara es mostrara quins directoris que es poden utilitzar:"
echo ""
cd /home
ls -d */
exit 1
fi

# Ara indicarem que si existeix el directori, que printi l'espai que ocupa
if [ $comprovacio -eq 0 ]; then
cd /home
du -sh $1
echo ""
fi
```

Comprovació:

```
0374 RA7 Scripting bash

isard@ubuntu-24:~$ nano exercici12.sh
isard@ubuntu-24:~$ ./exercici12.sh qwerty

-----
Ara es mostrara l'espai que ocupa el directori escrit com a parametre en el disc:
(directori utilitzat = qwerty)

Error: El directori senyalat com a parametre no s'ha trobat o no existeix
Ara es mostrara quins directoris que es poden utilitzar:

hola/ isard/
isard@ubuntu-24:~$ ./exercici12.sh hola/

-----
Ara es mostrara l'espai que ocupa el directori escrit com a parametre en el disc:
(directori utilitzat = hola/)

4,0K hola/

isard@ubuntu-24:~$ ./exercici12.sh isard/

-----
Ara es mostrara l'espai que ocupa el directori escrit com a parametre en el disc:
(directori utilitzat = isard/)

117M isard/

isard@ubuntu-24:~$
```

13.	Fer un script que mostri els usuaris del sistema que poden iniciar sessió i permeti accedir al sistema amb l'usuari seleccionat. Si l'usuari seleccionat no existeix té que mostrar el missatge de "L'usuari no existeix"
Script:	

```
GNU nano 7.2                                exercici13.sh
# Daniel Roblas

# Ara fare un echo per printar una advertencia al usuari del que es printara
# per pantalla seguit de una comanda que mostrin tots els usuaris que poden
# iniciar sessio.
echo ""
echo "-----"
echo "Ara es mostrara per pantalla els usuaris que tenen aquest a aquest sistema:"
echo "-----"
echo ""

cat /etc/passwd | grep /bin/bash | cut -d ":" -f 1

echo ""
echo "-----"
echo "Dels usuaris proporcionats, digues amb quin vols iniciar sessió:"
echo "-----"
echo ""

# Aquí es la comprovació de si el usuari existeix, fent que si amb la comanda
# grep troba l'usuari, sigui 0 i sino sigui 1, i a partir de aquí ja es pot fer
# una estructura de control depenent de si el grep troba o no l'usuari indicat.
read user
grep $user /etc/passwd
n1=$?

if [ $n1 -ne 0 ]; then
echo "Error: L'usuari escrit, no existeix"
else
echo ""
echo "Iniciant sessio amb l'usuari seleccionat..."
echo ""
su $user
fi
```

Comrpovació:


```
isard@ubuntu-24:~$ nano exercici13.sh
isard@ubuntu-24:~$ ./exercici13.sh

-----
Ara es mostrara per pantalla els usuaris que tenen aquest a aquest sistema:
-----

root
isard
Daniel

-----
Dels usuaris proporcionats, digues amb quin vols iniciar sessió:
-----

Pepe
Error: L'usuari escrit, no existeix
isard@ubuntu-24:~$ ./exercici13.sh

-----
Ara es mostrara per pantalla els usuaris que tenen aquest a aquest sistema:
-----

root
isard
Daniel

-----
Dels usuaris proporcionats, digues amb quin vols iniciar sessió:
-----

Daniel
Daniel:x:1001:1001::/home/Daniel:/bin/bash

Iniciant sessio amb l'usuari seleccionat...

Contrasenya:
Daniel@ubuntu-24:/home/isard$ pwd
/home/isard
Daniel@ubuntu-24:/home/isard$ whoami
Daniel
Daniel@ubuntu-24:/home/isard$ █
```

Lectura de fitxers CSV

Una tasca molt habitual és la de llegir fitxers en format CSV (separat per ,) on hi ha informació com per exemple usuaris o directoris / fitxers a crear.

14.	<p>A partir d'un fitxer CSV (es passarà el nom del fitxer com a argument a l'script) que contindrà el següent format:</p> <p><i>Nom d'usuari, shell, home, password, email</i> → ex ("jcasas,/bin/bash,/home/jordi,jordi.casas@itb.cat")</p> <p>Fer un script que llegeixi cada línia i faci les següents operacions:</p> <ul style="list-style-type: none"> • Comprova que l'usuari no existeixi al sistema. Si existeix indica "Usuari \$nom_usuari ja existent al sistema" • Si no existeix crea l'usuari al sistema amb les dades indicades al fitxer. Un cop creat es traurà el missatge de "Usuari \$nom_usuari donat d'alta al sistema" <p>L'script ha de tenir el control dels paràmetres correctes i verificar que el fitxer existeix. Així mateix tot missatge que es genera de l'script, a part de sortir per pantalla, s'ha de registrar en un fitxer logs.txt</p>
-----	--

Script:

```

GNU nano 7.2                                exercici14.sh
#!/bin/bash

# Aquest script llegeix cada línia de un fitxer CSV i realitza unes operacions: Mirar si l'usuari
# existeix en el sistema i si existeix indica que sí, i que si no existeix, que l'escripti crei
# l'usuari amb les dades del fitxer csv.
# Daniel Roblas

# Utilitzo la comanda exec per copiar la sortida i entrada en un fitxer logs.txt
exec >>(tee -a logs.txt)

# Aquesta part del script mira si el paràmetre posat existeix, i si no, printa error. També, si
# el usuari no passa cap paràmetre, també dona error.
test -f $1
fitxer=$?
if [ $# -ne 1 ]; then
echo ""
echo "Error: Per dur a terme aquest script, tens que posar com a paràmetre el fitxer CSV i NOMÉS un para>
echo ""
exit 1
elif [ $fitxer -ne 0 ]; then
echo ""
echo "El fitxer CSV no existeix o no s'ha trobat"
echo ""
exit 1
fi

# Ara utilitzarem el while per a que llegeixi cada línia i identifiqui amb el IFS cada columna del arxiu
# seguidament comprova si existeix l'usuari de cada fila i si no existeix, el crea amb les mateixes dades
# del fitxer CSV amb el useradd.
while IFS=',' read usuari shell home password mail; do
grep $usuari /etc/passwd | grep /bin/bash >> /dev/null
usr=$?
if [ $usr -eq 0 ]; then
echo ""
echo "El usuari '$usuari' ja existeix en el sistema i es pot iniciar sessió"
echo ""
elif [ $usr -ne 0 ]; then
echo ""
echo "El usuari '$usuari' no existeix en el sistema. Seguidament ara es creara..."
echo "(Si dona error, executa l'escript amb 'sudo') "
echo ""
useradd -m -s /bin/bash -d "$home" -p "$password" "$usuari"
fi

```

Comrpovació:

Error: Per dur a terme aquest script, tens que posar com a parametre el fitxer CSV i NOMÉS un parametre

^C

```
isard@ubuntu-24:~$ ./exercici14.sh qwerty
```

El fitxer CSV no existeix o no s'ha trobat

```
isard@ubuntu-24:~$ ./exercici14.sh CSV14.csv
```

El usuari 'jcasas' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

useradd: Permission denied.
useradd: no es pot blocar /etc/passwd, intenteu-ho més tard

El usuari 'mperez' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

useradd: Permission denied.
useradd: no es pot blocar /etc/passwd, intenteu-ho més tard

El usuari 'aluque' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

useradd: Permission denied.
useradd: no es pot blocar /etc/passwd, intenteu-ho més tard

```
isard@ubuntu-24:~$ sudo ./exercici14.sh CSV14.csv
```

El usuari 'jcasas' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

useradd: warning: the home directory /home/jcasas already exists.
useradd: Not copying any file from skel directory into it.

El usuari 'mperez' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

useradd: warning: the home directory /home/mperez already exists.
useradd: Not copying any file from skel directory into it.

El usuari 'aluque' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

useradd: warning: the home directory /home/aluque already exists.
useradd: Not copying any file from skel directory into it.

```
isard@ubuntu-24:~$ grep /bin/bash /etc/passwd | cut -d ":" -f 1
```

root

isard

Daniel

jcasas

mperez

aluque

```
isard@ubuntu-24:~$
```

```
isard@ubuntu-24:~$ sudo ./exercici14.sh CSV14.csv

El usuari 'jcasas' ya existeix en el sistema i es pot iniciar sessio

El usuari 'mperez' ya existeix en el sistema i es pot iniciar sessio

El usuari 'aluque' ya existeix en el sistema i es pot iniciar sessio

isard@ubuntu-24:~$ grep /bin/bash /etc/passwd | cut -d ":" -f 1
root
isard
Daniel
jcasas
mperez
aluque
isard@ubuntu-24:~$
```

Archiu logs.txt:

```
Obre ▾  [+]
```

```
logs.txt
~/
```

```
|
Error: Per dur a terme aquest script, tens que posar com a parametre el fitxer CSV i NOMÉS un parametre

Error: Per dur a terme aquest script, tens que posar com a parametre el fitxer CSV i NOMÉS un parametre

Error: Per dur a terme aquest script, tens que posar com a parametre el fitxer CSV i NOMÉS un parametre

Error: Per dur a terme aquest script, tens que posar com a parametre el fitxer CSV i NOMÉS un parametre

El fitxer CSV no existeix o no s'ha trobat

El usuari 'jcasas' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

El usuari 'mperez' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

El usuari 'alugue' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

El usuari 'jcasas' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

El usuari 'mperez' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

El usuari 'alugue' no existeix en el sistema. Seguidement ara es creara...
(Si dona error, executa l'ecript amb 'sudo')

El usuari 'jcasas' ya existeix en el sistema i es pot iniciar sessio

El usuari 'mperez' ya existeix en el sistema i es pot iniciar sessio
```

Script de còpia de seguretat

15.	<p>Realitzar un script final que farà una còpia de seguretat d'una carpeta que es passi per paràmetre. El segon paràmetre serà el directori on cal guardar el fitxer de la còpia. A tenir en compte:</p> <ul style="list-style-type: none"> • La còpia cal fer-la amb la comanda tar i que comprimeixi el contingut. • El nom del fitxer ha de ser backup_nom_directori_YYYYMMDD.tar.gz • L'script cal que controlï primer que no existeixi ja un fitxer prèviament amb el mateix nom. Si és així, l'script donarà error i parará, retornant codi 2 • Si no existeix el directori a fer-ne la còpia, l'script donarà error i parará, retornant codi 1
-----	---

Scripts:

```

GNU nano 7.2                                exercici15.sh
#!/bin/bash

# Aquest script fara una copia de seguretat d'una carpeta que es pasi com a parametre. El segon
# parametre sera el directori on cal guradar el fitxer de la copia.
# Daniel Roblas

# Ara mirarem que el usuari posa be els parametres, sino retornara error
if [ $# -ne 2 ]; then
echo ""
echo "Error: Tens que posar NOMÉS dos parametres"
echo ""
exit 1
fi
test -f $1
txt=$?
test -d $2
dir=$?
if [ $txt -ne 0 ]; then
echo ""
echo "Error: El fitxer " $1 " no s'ha trobat o no existeix"
echo ""
exit 1
elif [ $dir -ne 0 ]; then
echo ""
echo "Error: El directori " $2 " no s'ha trobat o no existeix "
echo ""
exit 1
fi

# Ara es creen les variables que seran neccesaries per a formar el nom
nom=$1
directori=$2
data=$(date +%Y%m%d)
n_back="backup_$(basename "$nom")_$data.tar.gz"

# Ara mirarem que no existeixi un fitxer amb el mateix nom en el directori que el usuari el vol copiar
if [ -f "$directori/$n_back" ]; then
echo ""
echo "Error: El archiu a copiar, ya existeix en el directori en el qual es vol fer el backup"
echo ""
exit 2
fi

# Ara farem la copia del fitxer al directori indicant la variable n_back que sera el nom del backup
tar -czvf "$directori/$n_back" -C "$(dirname "$nom")" "$(basename "$nom")"

```

Comprovació:

```
isard@ubuntu-24:~$ nano exercici15.sh
isard@ubuntu-24:~$ ./exercici15.sh

Error: Tens que posar NOMÉS dos parametres

isard@ubuntu-24:~$ ./exercici15.sh qweqwe qweqwe

Error: El fitxer qweqwe no s'ha trobat o no existeix

isard@ubuntu-24:~$ ./exercici15.sh qweqwe qweqwe qweqwe

Error: Tens que posar NOMÉS dos parametres

isard@ubuntu-24:~$ ./exercici15.sh exercici5.sh Escriptori
exercici5.sh
isard@ubuntu-24:~$ ls
1          du          exercici12.sh.save  exercici17.sh      exercici6.sh      holaaa      Pepe          snap
Baixades   Escriptori  exercici13.sh      exercici1.sh      exercici7.sh      holaaaa    Plantilles    Vídeos
B_pepe     exercici10.sh exercici14.sh      exercici3.sh      exercici8.sh      Imatges    Públic
CSV14.csv  exercici11.sh exercici15.sh      exercici4.sh      exercici9.sh      logs.txt   script1234.sh
Documents  exercici12.sh exercici16.sh      exercici5.sh      hola              Música     scripts
isard@ubuntu-24:~$ ls Escriptori/
backup__20260115.tar.gz  backup_exercici5.sh_20260115.tar.gz  backup_nom_20260115.tar.gz  logs.txt
isard@ubuntu-24:~$
```

Operadors

16.	<p>Escriu les ordres necessàries per cercar el nom d'un usuari anomenat "messi" al fitxer /etc/passwd i donar missatges diferents segons si es troba o no al fitxer d'usuaris..</p>
	<p>Scripts:</p>

```
GNU nano 7.2                                exercici16.sh *
#!/bin/bash
# Aquest script cerca si l'usuari "messi" existeix dins del fitxer /etc/passwd
# i mostra un missatge depenent de si hi es o no.
# Daniel Roblas

# Fem servir grep per buscar la linia exacta que comenci per "messi:"
grep "^messi" /etc/passwd > /dev/null 2>&1

if [ $? -eq 0 ]; then
echo ""
echo "L'usuari 'messi' s'ha trobat correctament en el fitxer d'usuaris"
echo ""
else
echo ""
echo "L'usuari 'messi' no existeix en aquest sistema"
echo ""
fi
```

Comprovació:

```
isard@ubuntu-24:~$ nano exercici16.sh
isard@ubuntu-24:~$ ./exercici16.sh

L'usuari 'messi' no existeix en aquest sistema

isard@ubuntu-24:~$ sudo useradd messi
isard@ubuntu-24:~$ ./exercici16.sh

L'usuari 'messi' s'ha trobat correctament en el fitxer d'usuaris

isard@ubuntu-24:~$
```

17.

Escriu les ordres necessàries per assegurar-te que el superusuari root és qui està executant el guió de shell i donar un missatge depenent de si ho és o no.

Scripts:


```
isard@ubuntu-24: ~  
GNU nano 7.2 exercici17.sh  
#!/bin/bash  
# Aquest script serveix per assegurar-se que qui l'executa  
# es l'usuari root.  
# Daniel Roblas  
  
if [ "$EUID" -ne 0 ]; then  
echo ""  
echo "Error: Aquest guio s'ha d'executar amb privilegis de root."  
echo ""  
exit 1  
else  
echo ""  
echo "Confirmacio: Estas executant el script com a superusuari root"  
echo ""  
fi
```

Comprovació:

```
isard@ubuntu-24: ~  
isard@ubuntu-24:~$ nano exercici17.sh  
isard@ubuntu-24:~$ ./exercici17.sh  
  
Error: Aquest guio s'ha d'executar amb privilegis de root.  
  
isard@ubuntu-24:~$ sudo ./exercici17.sh  
  
Confirmacio: Estas executant el script com a superusuari root  
isard@ubuntu-24:~$
```

AWS CLI

Es poden automatitzar les activitats amb els elements d'AWS utilitzant el CLI que ofereix AWS.

Recordar que el CLI cal configurar-lo amb les credencials correctes posant-les al fitxer

18.	<p>Definiu un script que faci un bolcat de totes les EC2 que estan operatives en un entorn d'AWS i en generi un CSV, indicant la següent informació:</p> <ul style="list-style-type: none"> • InstanceID • Estat de la instància (running, stopped, ...) • Type (t2.small, t2.micro, ...) • Nom de la màquina
-----	---

Scripts:

```

GNU nano 2.5.3                                     File: exercici18.sh

#!/bin/bash
# Daniel Roblas

# Primer defineixo el nom del fitxer on guardarem les dades
FITXER_SORTIDA="instancies_ec2.csv"

# Escric la capçalera del CSV
# El signe ">" crea el fitxer nou i esborra si hi havia alguna cosa d'abans
echo "InstanceID,Estat,Tipus,Nom" > $FITXER_SORTIDA

# Ara faig la consulta a AWS
# He fet servir '--query' per triar exactament
# L'opció '--output text' fa que sigui més fàcil de llegir per a l'script
aws ec2 describe-instances --query 'Reservations[*].Instances[*].[
    InstanceId,
    State.Name,
    InstanceType,
    Tags[?Key==`Name`].Value | [0]
]' --output text | while read -r ID ESTAT TIPUS NOM; do

    # De vegades les màquines no tenen nom i AWS torna un valor buit o "None"
    # Si passa això, hi posarem "SenseNom" per no deixar la columna buida
    if [ "$NOM" == "None" ] || [ -z "$NOM" ]; then
        NOM="SenseNom"
    fi

    # Finalment, guardem la línia al fitxer CSV separada per comes
    echo "$ID,$ESTAT,$TIPUS,$NOM" >> $FITXER_SORTIDA
done

echo "El procés ha acabat i s'ha creat el fitxer $FITXER_SORTIDA"

```

Comprovació:

```
eee_W_5489385@runweb204934:~$ ./exercici18.sh
bash: ./exercici18.sh: Permission denied
eee_W_5489385@runweb204934:~$ sudo ./exercici18.sh
[sudo] password for eee_W_5489385:
Sorry, try again.
[sudo] password for eee_W_5489385:
sudo: 1 incorrect password attempt
eee_W_5489385@runweb204934:~$ source exercici18.sh
El procés ha acabat i s'ha creat el fitxer instancies_ec2.csv
eee_W_5489385@runweb204934:~$ cat instancies_ec2.csv
InstanceID,Estat,Tipus,Nom
i-0c6e7a96482bd1d0b,running,t3.micro,AP2-3
i-06c672faecfae9778,running,t3.micro,AP2-0
i-037af8ac7bebb44a8,running,t3.micro,AP2-1
i-04f64e428457a4606,running,t3.micro,AP2-2
eee_W_5489385@runweb204934:~$ █
```

19.	<p>Defiueix un script que a partir d'un CSV que conté -com a mínim- aquesta informació, creï les màquines a AWS de forma automàtica:</p> <ul style="list-style-type: none"> • Tipus d'instància • Tipus d'AMI • Security group • Nom de la instància (etiqueta)
<p>Scripts:</p>	

```
GNU nano 2.5.3                               File: exercici19.sh

#!/bin/bash
#Daniel Roblas

# Defineixo quin és el fitxer que conté les dades
FITXER_DADES="instancies_a_crear.csv"

# Comprovo si el fitxer existeix abans de començar
if [ ! -f "$FITXER_DADES" ]; then
    echo "Error: No trobo el fitxer $FITXER_DADES"
    exit 1
fi

# Faig servir 'tail' per saltar-me la primera línia (la de les capçaleres)
# Després llegeixo el fitxer línia per línia
tail -n +2 "$FITXER_DADES" | while IFS=',' read -r TIPUS AMI SG NOM
do
    echo "Intentant crear la instància: $NOM..."

    # Aquest és el comandament per crear la màquina
    # He après que 'TagSpecifications' serveix per posar-li el nom des del principi
    aws ec2 run-instances \
        --image-id "$AMI" \
        --instance-type "$TIPUS" \
        --security-group-ids "$SG" \
        --tag-specifications "ResourceType=instance,Tags=[{Key=Name,Value=$NOM}]" \
        --output table

    if [ $? -eq 0 ]; then
        echo "La màquina $NOM s'ha enviat a crear correctament."
    else
        echo "Hi ha hagut un error en crear la màquina $NOM."
    fi

    echo "-----"
done

echo "Procés de lectura finalitzat."
```

Fitxer CSV:

```
GNU nano 2.5.3                               File: instancies_a_crear.csv

Tipus,AMI,SecurityGroup,Nom
t3.micro,ami-068c0051b15cdb816,sg-0617bf20b4f32bc54,AP2-3
t3.micro,ami-068c0051b15cdb816,sg-0617bf20b4f32bc54,AP2-0
t3.micro,ami-068c0051b15cdb816,sg-0617bf20b4f32bc54,AP2-1
t3.micro,ami-068c0051b15cdb816,sg-0617bf20b4f32bc54,AP2-2
```

Comprovació:

```
eee_W_5489385@runweb204934:~$ source exercici19.sh
Intentant crear la instància: Proves-1...
```

```
An error occurred (InvalidAMIID.NotFound) when calling the RunInstances operation: The image id '[ami-0c55b159cbfafe1f0]' does not exist
Hi ha hagut un error en crear la màquina Proves-1.
```

```
-----
Intentant crear la instància: Proves-2...
```

```
An error occurred (InvalidAMIID.NotFound) when calling the RunInstances operation: The image id '[ami-0c55b159cbfafe1f0]' does not exist
Hi ha hagut un error en crear la màquina Proves-2.
```

```
-----
Procés de lectura finalitzat.
```

```
eee_W_5489385@runweb204934:~$
```

```
eee_W_5489385@runweb204934:~$ source exercici19.sh
: AP2-3...
```

```
-----
|                                     RunInstances                                     |
+-----+-----+-----+-----+
| OwnerId          | 211125540631 |
| ReservationId    | r-0f6dd3f0414a449c1 |
+-----+-----+-----+-----+
|                                     Instances                                     |
+-----+-----+-----+-----+
| AmiLaunchIndex   | 0             |
| Architecture     | x86_64        |
| ClientToken      | 6936d6f0-b00e-470d-b845-88b47c459b18 |
| EbsOptimized     | False         |
| EnaSupport       | True          |
| Hypervisor       | xen           |
| ImageId          | ami-068c0051b15cdb016 |
| InstanceId       | i-0fa822db1f3e29672 |
| InstanceType     | t3.micro       |
| LaunchTime       | 2026-01-15T07:46:00+00:00 |
| PrivateDnsName   | ip-172-31-20-221.ec2.internal |
| PrivateIpAddress | 172.31.20.221 |
| PublicDnsName    |                |
| RootDeviceName   | /dev/xvda     |
| RootDeviceType   | ebs           |
| SourceDestCheck  | True          |
| StateTransitionReason |                |
| SubnetId         | subnet-010b25c3a8a4b3d9b |
| VirtualizationType | hvm          |
| VpcId            | vpc-0cf067967a5ed56f7 |
+-----+-----+-----+-----+
|                                     CapacityReservationSpecification                                     |
+-----+-----+-----+-----+
| CapacityReservationPreference | open |
+-----+-----+-----+-----+
|                                     CpuOptions                                     |
+-----+-----+-----+-----+
| CoreCount        | 1             |
| ThreadsPerCore   | 2             |
+-----+-----+-----+-----+
|                                     EnclaveOptions                                     |
+-----+-----+-----+-----+
| Enabled          | False         |
+-----+-----+-----+-----+
|                                     MetadataOptions                                     |
+-----+-----+-----+-----+
| HttpEndpoint     | enabled       |
| HttpPutResponseHopLimit | 2           |
| HttpTokens       | required      |
| State            | pending       |
+-----+-----+-----+-----+
|
```