

UNITAT 4. PRÀCTICA 1.

SISTEMA OPERATIU

WINDOWS

Alumne: David Peñalver Navarro

Professor: Manuel Enguñados

Assignatura: Sistemes Informàtics

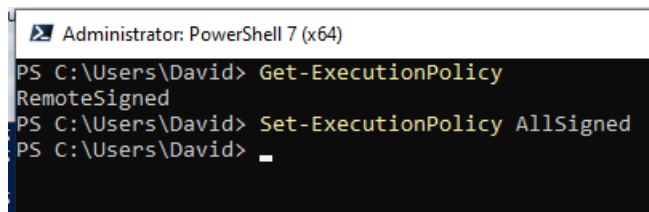
Índex

Pràctica 1.....	3
1- Activitat 1.....	3
2- Activitat 2.....	5
3- Activitat 3.....	6
4- Activitat 4.....	7
5- Activitat 5.....	8
6- Activitat 6.....	9
7- Activitat 7.....	11
8- Activitat 8.....	12
9- Activitat 9.....	13
10- Activitat 10.....	14

Pràctica 1

1- Activitat 1

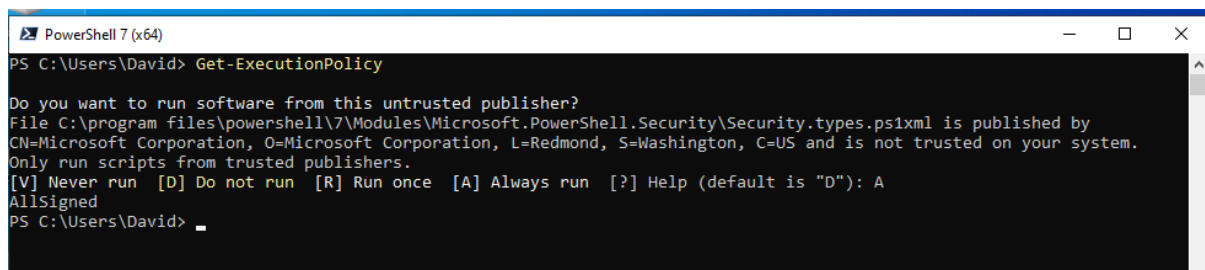
Primer de tot, obrim PowerShell com administrador per a ficar-nos permís de tots els scripts:



```
Administrator: PowerShell 7 (x64)
PS C:\Users\David> Get-ExecutionPolicy
RemoteSigned
PS C:\Users\David> Set-ExecutionPolicy AllSigned
PS C:\Users\David> _
```

Figura 1. Ficant-nos permís per a tots els scripts.

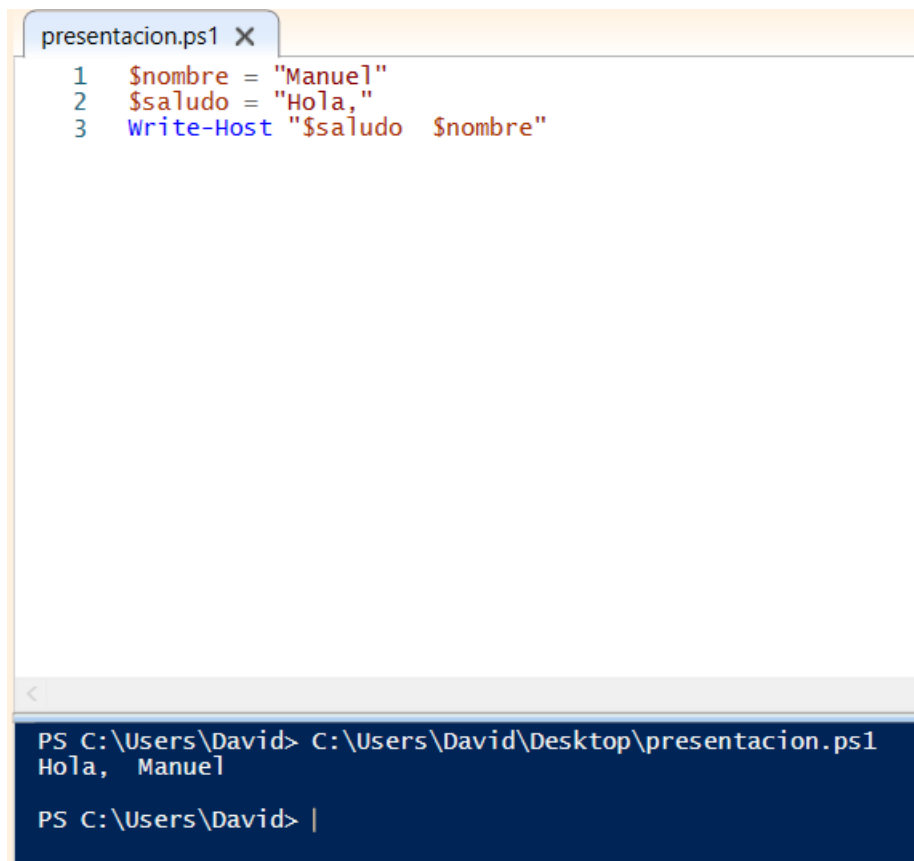
Ara ja tenim els permissos sense obrir com administrador:



```
PowerShell 7 (x64)
PS C:\Users\David> Get-ExecutionPolicy
Do you want to run software from this untrusted publisher?
File C:\program files\powershell\7\Modules\Microsoft.PowerShell.Security\Security.types.ps1xml is published by
CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US and is not trusted on your system.
Only run scripts from trusted publishers.
[V] Never run [D] Do not run [R] Run once [A] Always run [?] Help (default is "D"): A
AllSigned
PS C:\Users\David> _
```

Figura 2. Comprovant els permissos.

Ara ja podem començar el primer exercici. Escribim el següent codi i executem:



The image shows a Windows PowerShell window with a tab titled 'presentacion.ps1'. The script contains three lines of code: 1. `$nombre = "Manuel"`, 2. `$saludo = "Hola,"`, and 3. `Write-Host "$saludo $nombre"`. The terminal output shows the command being executed: `PS C:\Users\David> C:\Users\David\Desktop\presentacion.ps1`, followed by the output: `Hola, Manuel`. The prompt then returns to `PS C:\Users\David> |`.

```
presentacion.ps1 X
1 $nombre = "Manuel"
2 $saludo = "Hola,"
3 Write-Host "$saludo $nombre"

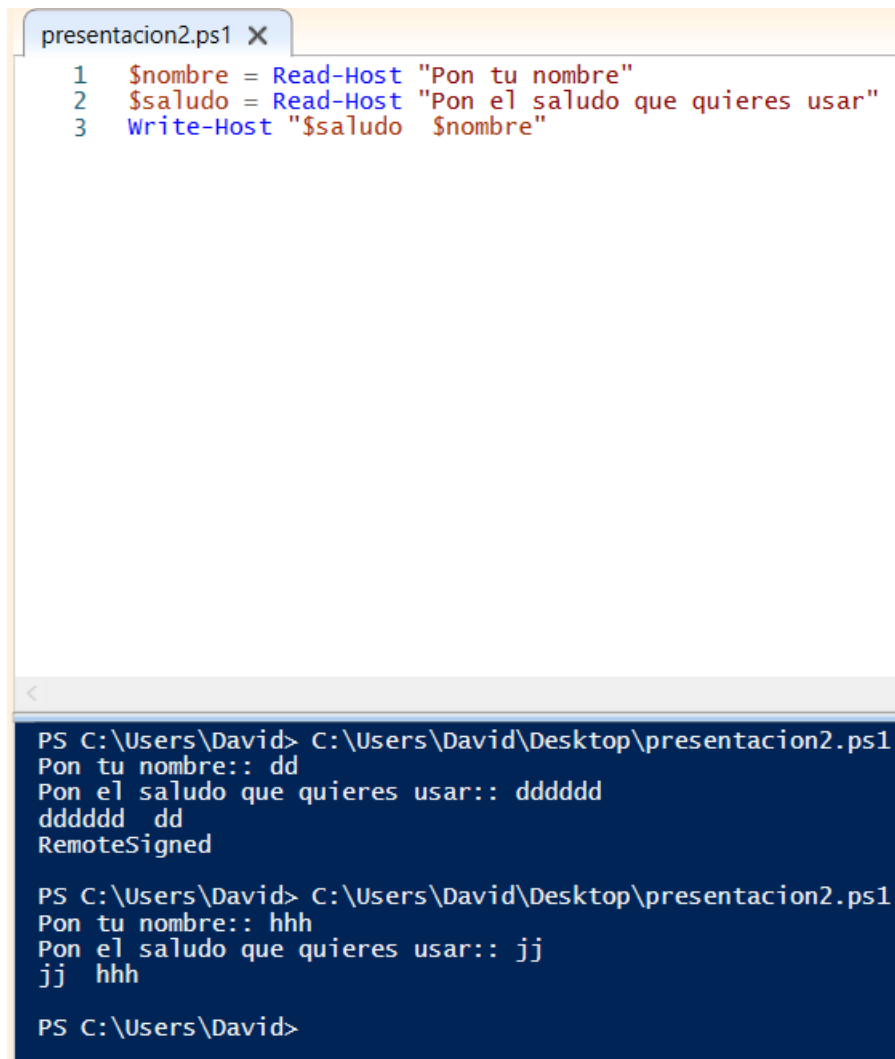
PS C:\Users\David> C:\Users\David\Desktop\presentacion.ps1
Hola, Manuel

PS C:\Users\David> |
```

Figura 3. Exercici 1.

2- Activitat 2

Fiquem el codi de la part en blanc i a l'executar eixirà la part blava:



The image shows a PowerShell script editor window titled 'presentacion2.ps1'. The script contains three lines of code: 1. `$nombre = Read-Host "Pon tu nombre"`, 2. `$saludo = Read-Host "Pon el saludo que quieres usar"`, and 3. `Write-Host "$saludo $nombre"`. Below the editor, a terminal window shows the execution of the script. The first run shows the user entering 'dd' for the name and 'dddddd' for the greeting, resulting in the output 'dddddd dd'. The second run shows the user entering 'hhh' for the name and 'jj' for the greeting, resulting in the output 'jj hhh'.

```
presentacion2.ps1 X
1 $nombre = Read-Host "Pon tu nombre"
2 $saludo = Read-Host "Pon el saludo que quieres usar"
3 Write-Host "$saludo $nombre"

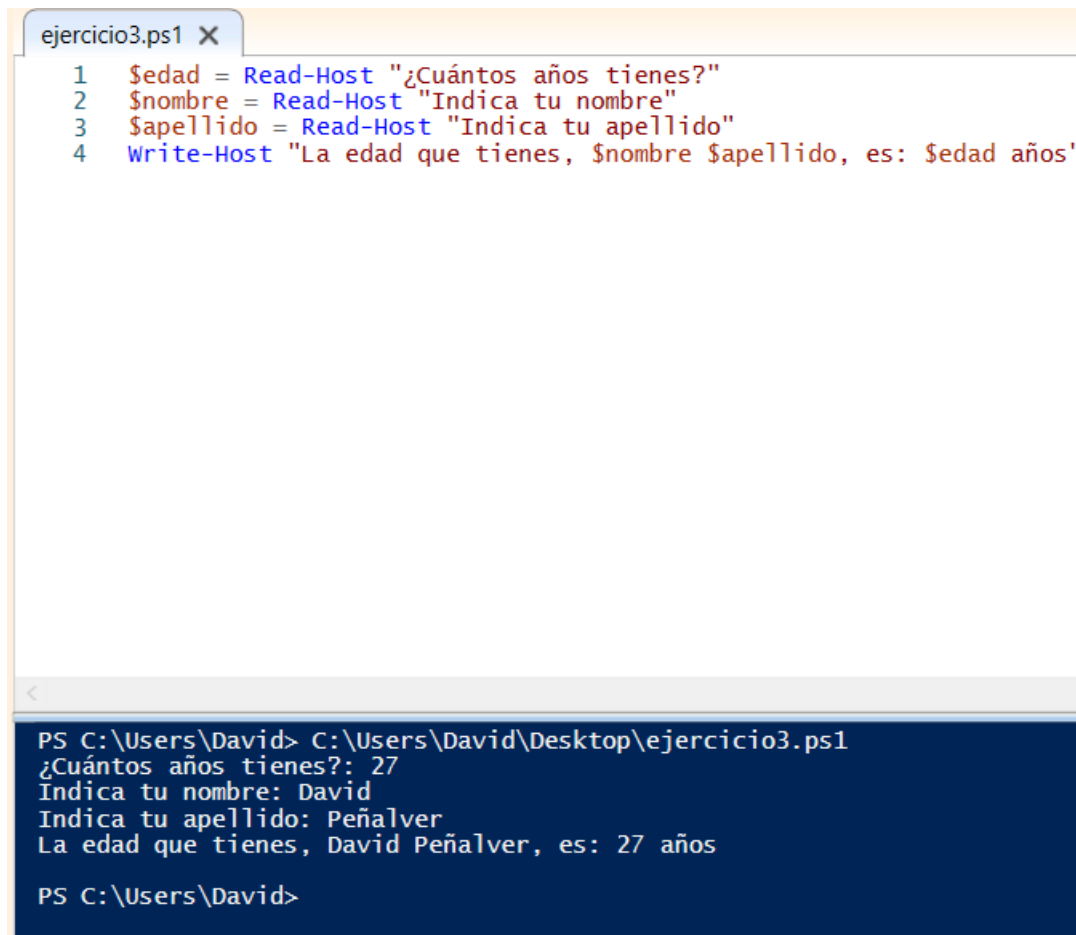
PS C:\Users\David> C:\Users\David\Desktop\presentacion2.ps1
Pon tu nombre:: dd
Pon el saludo que quieres usar:: ddddddd
dddddd dd
RemoteSigned

PS C:\Users\David> C:\Users\David\Desktop\presentacion2.ps1
Pon tu nombre:: hhh
Pon el saludo que quieres usar:: jj
jj hhh

PS C:\Users\David>
```

Figura 4. Exercici 2.

3- Activitat 3



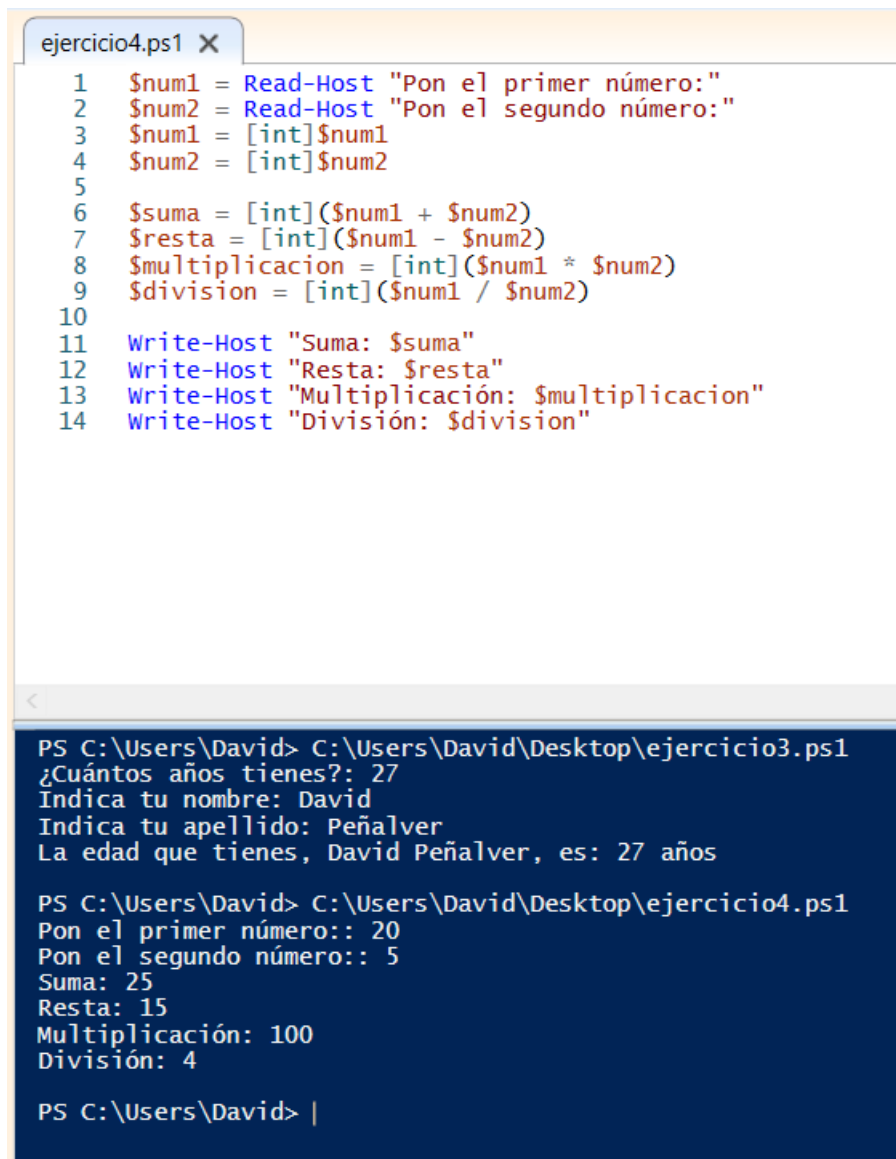
The image shows a PowerShell script editor window titled 'ejercicio3.ps1'. The script contains four lines of code: 1. `$edad = Read-Host "¿Cuántos años tienes?"`, 2. `$nombre = Read-Host "Indica tu nombre"`, 3. `$apellido = Read-Host "Indica tu apellido"`, and 4. `Write-Host "La edad que tienes, $nombre $apellido, es: $edad años"`. Below the editor, a terminal window shows the execution of the script. The prompt is `PS C:\Users\David> C:\Users\David\Desktop\ejercicio3.ps1`. The script prompts for age, name, and surname, and then outputs the result: `¿Cuántos años tienes?: 27`, `Indica tu nombre: David`, `Indica tu apellido: Peñalver`, and `La edad que tienes, David Peñalver, es: 27 años`. The terminal prompt returns to `PS C:\Users\David>`.

```
ejercicio3.ps1 X
1 $edad = Read-Host "¿Cuántos años tienes?"
2 $nombre = Read-Host "Indica tu nombre"
3 $apellido = Read-Host "Indica tu apellido"
4 Write-Host "La edad que tienes, $nombre $apellido, es: $edad años"

PS C:\Users\David> C:\Users\David\Desktop\ejercicio3.ps1
¿Cuántos años tienes?: 27
Indica tu nombre: David
Indica tu apellido: Peñalver
La edad que tienes, David Peñalver, es: 27 años
PS C:\Users\David>
```

Figura 5. Exercici 3.

4- Activitat 4



The image shows a PowerShell script named 'ejercicio4.ps1' and its execution output in a terminal window.

```
ejercicio4.ps1 X
1 $num1 = Read-Host "Pon el primer número:"
2 $num2 = Read-Host "Pon el segundo número:"
3 $num1 = [int]$num1
4 $num2 = [int]$num2
5
6 $suma = [int]($num1 + $num2)
7 $resta = [int]($num1 - $num2)
8 $multiplicacion = [int]($num1 * $num2)
9 $division = [int]($num1 / $num2)
10
11 Write-Host "Suma: $suma"
12 Write-Host "Resta: $resta"
13 Write-Host "Multiplicación: $multiplicacion"
14 Write-Host "División: $division"

PS C:\Users\David> C:\Users\David\Desktop\ejercicio3.ps1
¿Cuántos años tienes?: 27
Indica tu nombre: David
Indica tu apellido: Peñalver
La edad que tienes, David Peñalver, es: 27 años

PS C:\Users\David> C:\Users\David\Desktop\ejercicio4.ps1
Pon el primer número:: 20
Pon el segundo número:: 5
Suma: 25
Resta: 15
Multiplicación: 100
División: 4

PS C:\Users\David> |
```

Figura 6. Exercici 4.

5- Activitat 5

The image shows a Windows desktop environment with two windows open. The left window is a PowerShell command prompt (PS C:\Users\David>) running a script named 'ejercicio5.ps1'. The script prompts the user to enter the names of various soccer players and their positions. The right window is a text editor showing the source code of 'ejercicio5.ps1'. The code uses 'Write-Host' for prompts and 'Read-Host' to capture user input, storing it in variables like \$portero, \$sdef1, etc. Finally, it uses 'Write-Host' to display the collected data in a structured format.

```

PS C:\Users\David> C:\Users\David\Desktop\ejercicio5.ps1
Escribe el nombre del portero:
Palop
Escribe el nombre del primer central:
Ayala
Escribe el nombre del segundo central:
Ballesteros
Escribe el nombre del lateral derecho:
Miguel Brito
Escribe el nombre del lateral izquierdo:
Gayà
Escribe el nombre del pivote defensivo:
Iborra
Escribe el nombre del centrocampista:
Albelda
Escribe el nombre del interior derecho:
El Zhar
Escribe el nombre del interior izquierdo:
Vicente
Escribe el nombre del mediapunta:
Juan Sánchez
Escribe el nombre del delantero:
Caicedo
Portero: Palop
LD: Miguel Brito DEF: Ayala DEF: LI: Gayà
ID: El Zhar CC: Iborra CC: Albelda II: Vicente
MP: Juan Sánchez DC: Caicedo
PS C:\Users\David>

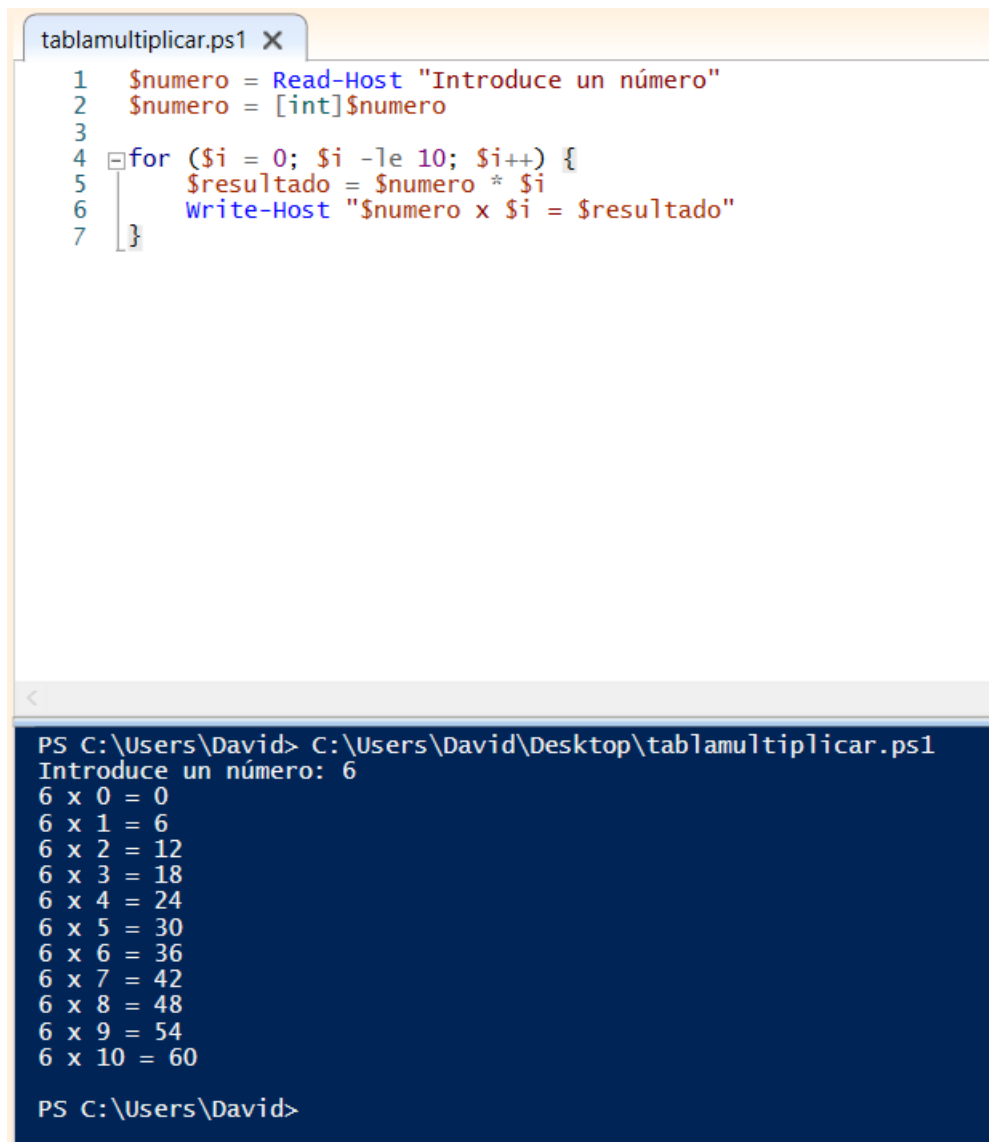
ejercicio5.ps1 X
1 Write-Host "Escribe el nombre del portero:"
2 $portero = Read-Host
3 Write-Host "Escribe el nombre del primer central:"
4 $sdef1 = Read-Host
5 Write-Host "Escribe el nombre del segundo central:"
6 $sdef2 = Read-Host
7 Write-Host "Escribe el nombre del lateral derecho:"
8 $sdef3 = Read-Host
9 Write-Host "Escribe el nombre del lateral izquierdo:"
10 $sdef4 = Read-Host
11 Write-Host "Escribe el nombre del pivote defensivo:"
12 $scent1 = Read-Host
13 Write-Host "Escribe el nombre del centrocampista:"
14 $scent2 = Read-Host
15 Write-Host "Escribe el nombre del interior derecho:"
16 $scent3 = Read-Host
17 Write-Host "Escribe el nombre del interior izquierdo:"
18 $scent4 = Read-Host
19 Write-Host "Escribe el nombre del mediapunta:"
20 $sdel1 = Read-Host
21 Write-Host "Escribe el nombre del delantero:"
22 $sdel2 = Read-Host
23
24
25 Write-Host "Portero: $portero"
26 Write-Host "LD: $sdef3 DEF: $sdef1 DEF: $def LI: $sdef4"
27 Write-Host "ID: $scent3 CC: $scent1 CC: $scent2 II: $scent4"
28 Write-Host "MP: $sdel1 DC: $sdel2"

```

Figura 7. Exercici 5.

6- Activitat 6

En este exercici, és clau fer un “cast” del número. En cas contrari, concatenarà les vegades per les que multipliquem el número.



The image shows a PowerShell script editor window titled 'tablamultiplicar.ps1' and a terminal window showing the script's execution. The script prompts the user to enter a number, casts it to an integer, and then uses a for loop to calculate and display the multiplication table for that number from 0 to 10.

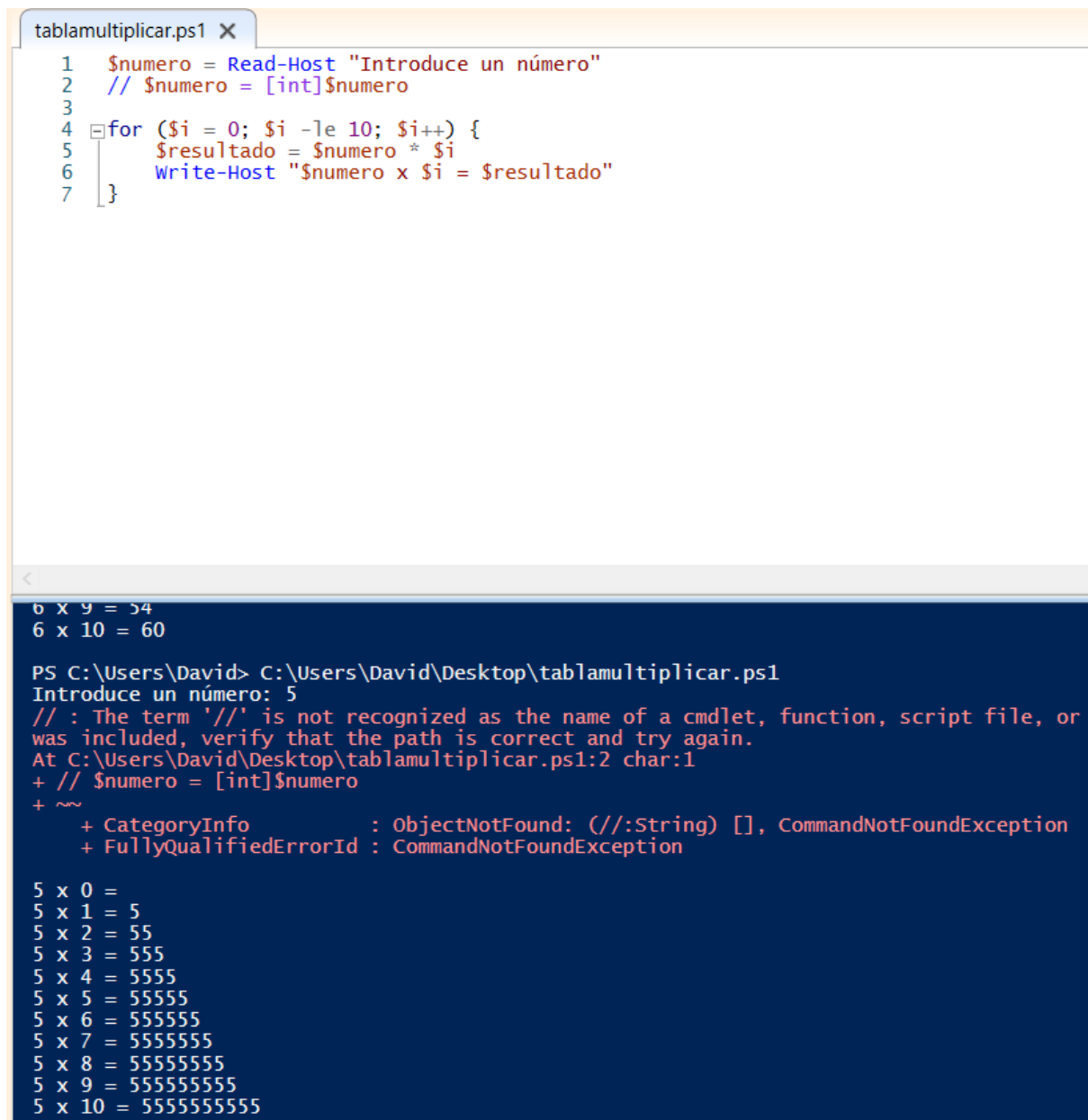
```
1 $numero = Read-Host "Introduce un número"
2 $numero = [int]$numero
3
4 for ($i = 0; $i -le 10; $i++) {
5     $resultado = $numero * $i
6     Write-Host "$numero x $i = $resultado"
7 }
```

Execution output:

```
PS C:\Users\David> C:\Users\David\Desktop\tablamultiplicar.ps1
Introduce un número: 6
6 x 0 = 0
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
PS C:\Users\David>
```

Figura 8. Exercici 6.

En cas de no castejar, donaria el següent resultat erroni:



```
1 $numero = Read-Host "Introduce un número"
2 // $numero = [int]$numero
3
4 for ($i = 0; $i -le 10; $i++) {
5     $resultado = $numero * $i
6     Write-Host "$numero x $i = $resultado"
7 }
```

6 x 9 = 54
6 x 10 = 60

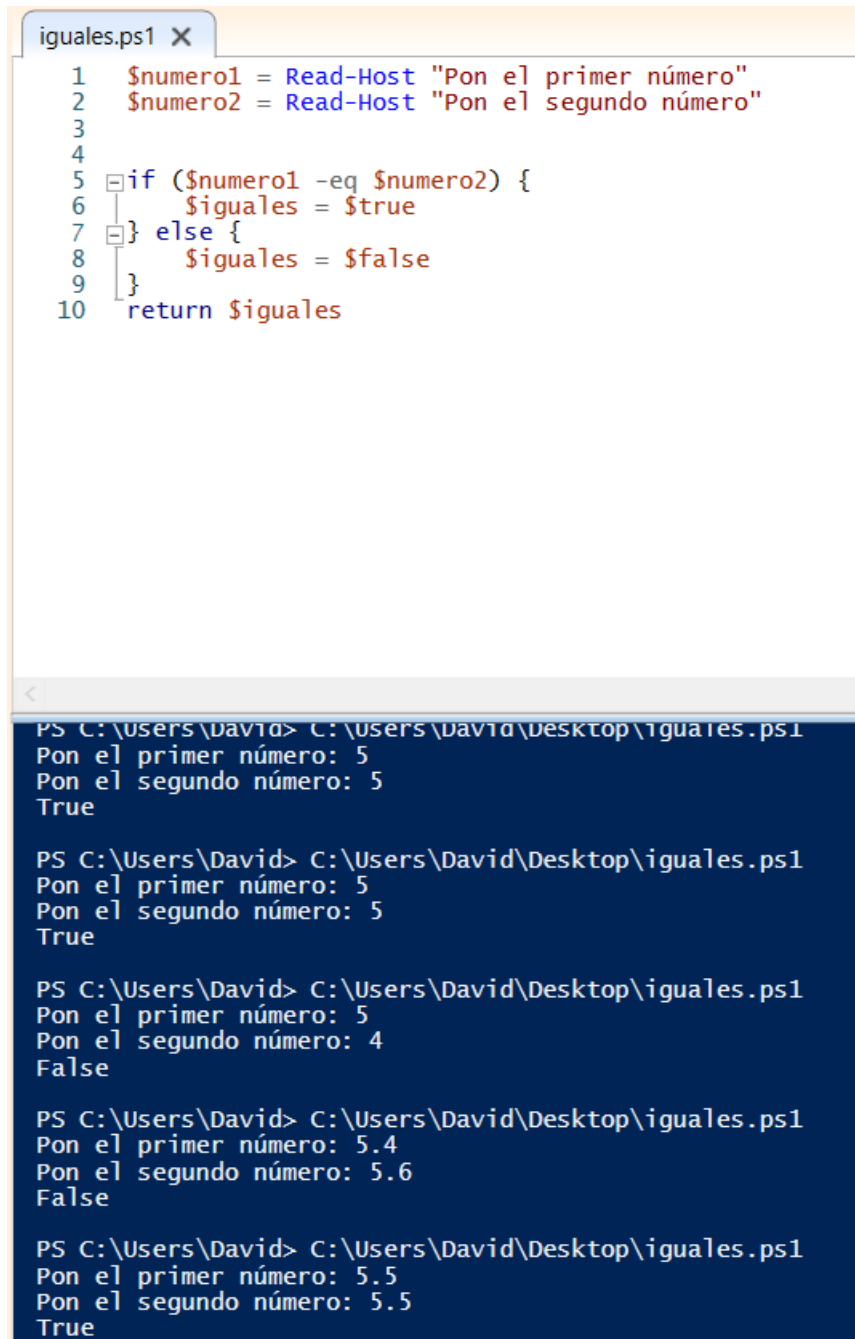
PS C:\Users\David> C:\Users\David\Desktop\tablamultiplicar.ps1
Introduce un número: 5
// : The term '//' is not recognized as the name of a cmdlet, function, script file, or
was included, verify that the path is correct and try again.
At C:\Users\David\Desktop\tablamultiplicar.ps1:2 char:1
+ // \$numero = [int]\$numero
+ ~~~
+ CategoryInfo : ObjectNotFound: (//:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

5 x 0 =
5 x 1 = 5
5 x 2 = 55
5 x 3 = 555
5 x 4 = 5555
5 x 5 = 55555
5 x 6 = 555555
5 x 7 = 5555555
5 x 8 = 55555555
5 x 9 = 555555555
5 x 10 = 5555555555

Figura 9. Resultat erroni per no castejar.

7- Activitat 7

Al contrari que en l'exercici anterior, no és necessari fer un "cast" dels números rebuts. Podem fer-ho de totes formes, no variarà el resultat. De fet, en cas de ficar números amb decimals que tinguin la mateixa part entera donaria un "true".



```
iguales.ps1 X
1 $numero1 = Read-Host "Pon el primer número"
2 $numero2 = Read-Host "Pon el segundo número"
3
4
5 if ($numero1 -eq $numero2) {
6     $iguales = $true
7 } else {
8     $iguales = $false
9 }
10 return $iguales

PS C:\Users\David> C:\Users\David\Desktop\iguales.ps1
Pon el primer número: 5
Pon el segundo número: 5
True

PS C:\Users\David> C:\Users\David\Desktop\iguales.ps1
Pon el primer número: 5
Pon el segundo número: 5
True

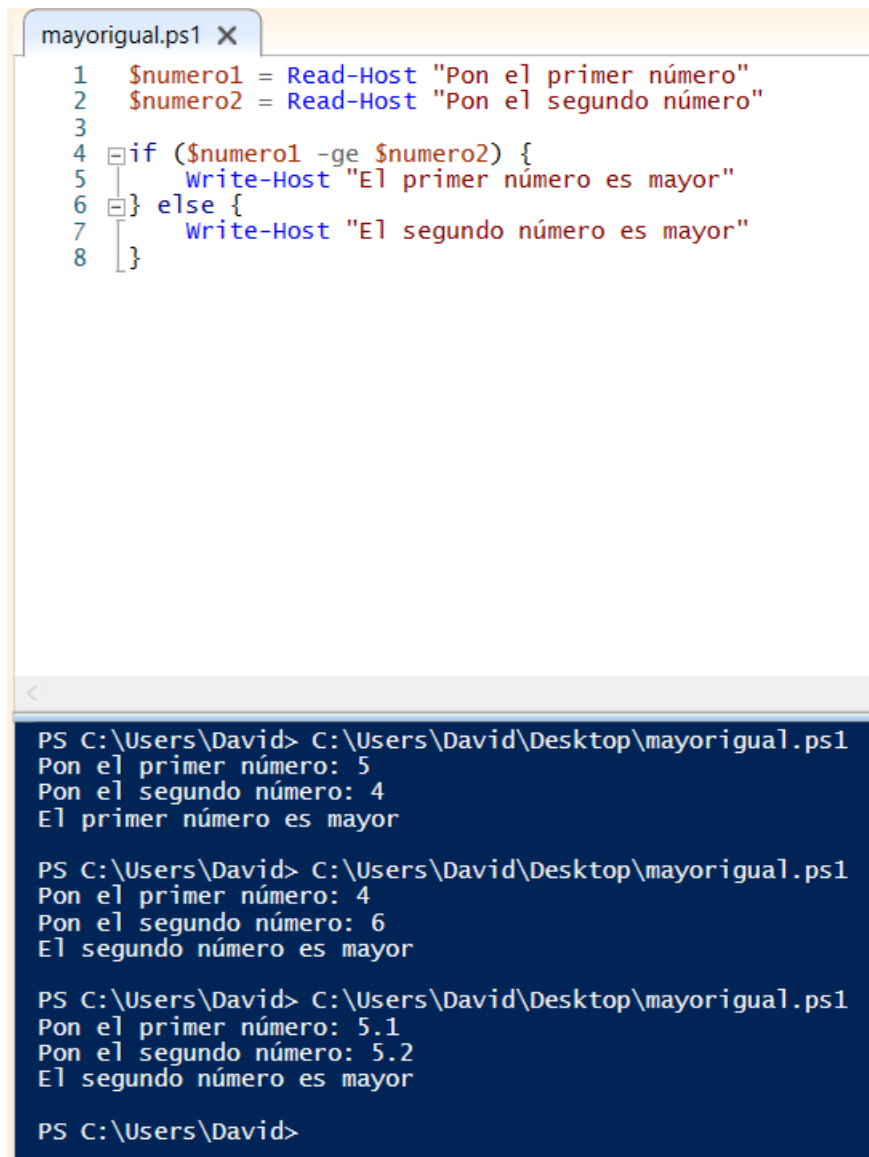
PS C:\Users\David> C:\Users\David\Desktop\iguales.ps1
Pon el primer número: 5
Pon el segundo número: 4
False

PS C:\Users\David> C:\Users\David\Desktop\iguales.ps1
Pon el primer número: 5.4
Pon el segundo número: 5.6
False

PS C:\Users\David> C:\Users\David\Desktop\iguales.ps1
Pon el primer número: 5.5
Pon el segundo número: 5.5
True
```

Figura 10. Exercici 7.

8- Activitat 8



The image shows a PowerShell script editor window titled 'mayorigual.ps1'. The script contains the following code:

```
1 $numero1 = Read-Host "Pon el primer número"
2 $numero2 = Read-Host "Pon el segundo número"
3
4 if ($numero1 -ge $numero2) {
5     Write-Host "El primer número es mayor"
6 } else {
7     Write-Host "El segundo número es mayor"
8 }
```

Below the editor, the execution output is shown in a dark blue console window. It displays three test cases where the script compares two numbers and prints the result.

```
PS C:\Users\David> C:\Users\David\Desktop\mayorigual.ps1
Pon el primer número: 5
Pon el segundo número: 4
El primer número es mayor

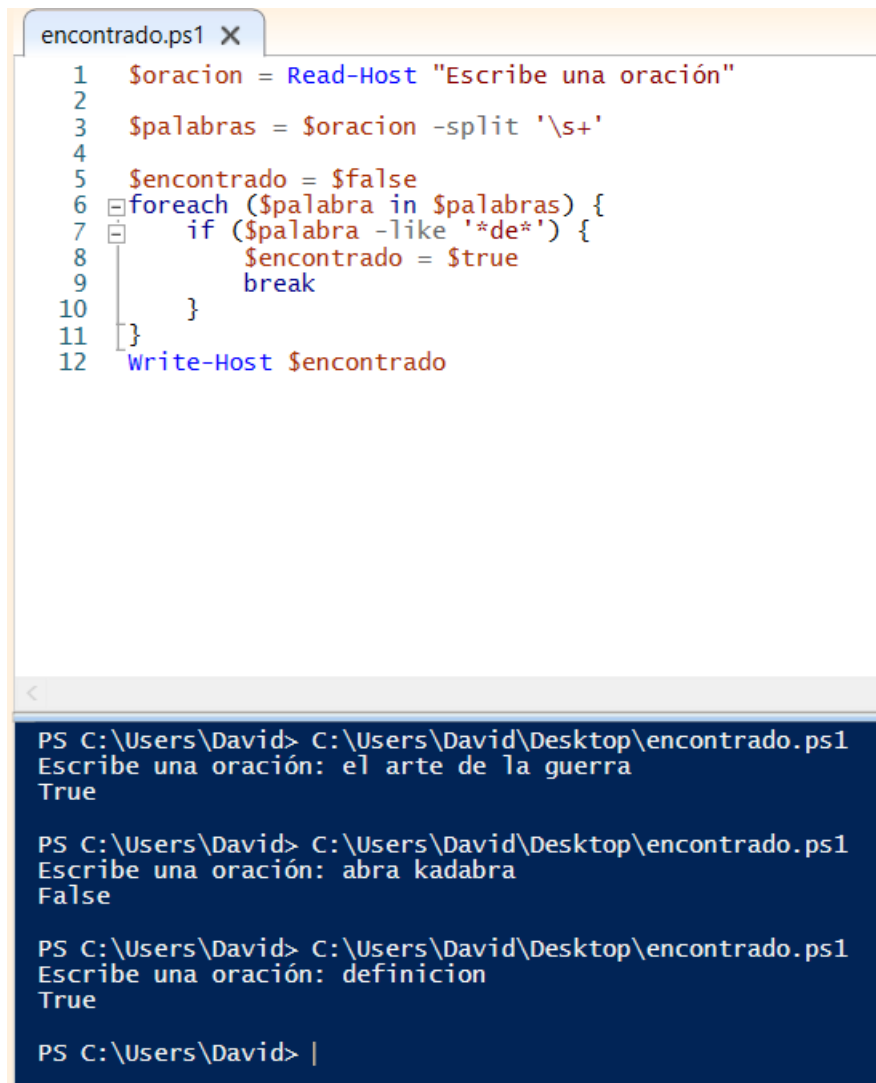
PS C:\Users\David> C:\Users\David\Desktop\mayorigual.ps1
Pon el primer número: 4
Pon el segundo número: 6
El segundo número es mayor

PS C:\Users\David> C:\Users\David\Desktop\mayorigual.ps1
Pon el primer número: 5.1
Pon el segundo número: 5.2
El segundo número es mayor

PS C:\Users\David>
```

Figura 11. Exercici 8.

9- Activitat 9



The image shows a PowerShell script named `encontrado.ps1` and its execution results. The script is designed to check if a sentence contains the word "de".

```
1 $oracion = Read-Host "Escribe una oración"
2
3 $palabras = $oracion -split '\s+'
4
5 $encontrado = $false
6 foreach ($palabra in $palabras) {
7     if ($palabra -like '*de*') {
8         $encontrado = $true
9         break
10    }
11 }
12 Write-Host $encontrado
```

The execution results show three examples:

```
PS C:\Users\David> C:\Users\David\Desktop\encontrado.ps1
Escribe una oración: el arte de la guerra
True

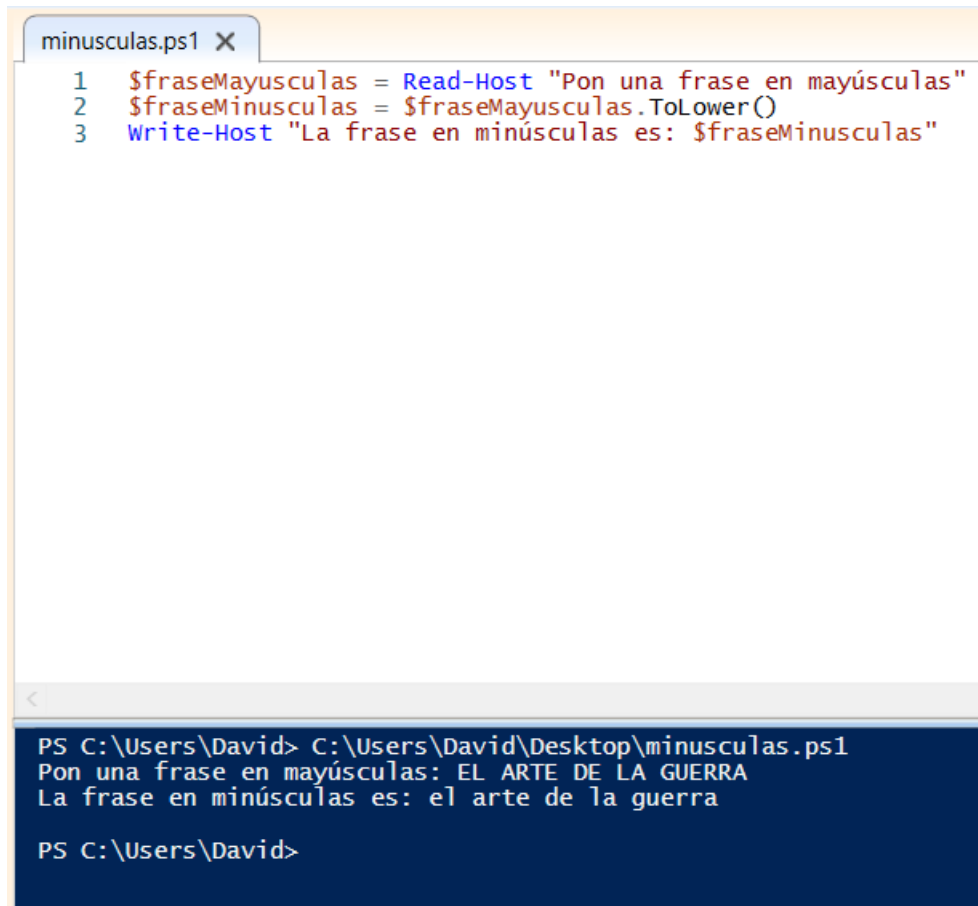
PS C:\Users\David> C:\Users\David\Desktop\encontrado.ps1
Escribe una oración: abra kadabra
False

PS C:\Users\David> C:\Users\David\Desktop\encontrado.ps1
Escribe una oración: definicion
True

PS C:\Users\David> |
```

Figura 12. Exercici 9.

10- Activitat 10



The image shows a PowerShell script named 'minusculas.ps1' in a text editor. The script contains three lines of code: 1. '\$fraseMayusculas = Read-Host "Pon una frase en mayúsculas"', 2. '\$fraseMinusculas = \$fraseMayusculas.ToLower()', and 3. 'Write-Host "La frase en minúsculas es: \$fraseMinusculas"'. Below the script, a terminal window shows the execution of the script. The prompt is 'PS C:\Users\David> C:\Users\David\Desktop\minusculas.ps1'. The first line of output is 'Pon una frase en mayúsculas: EL ARTE DE LA GUERRA'. The second line of output is 'La frase en minúsculas es: el arte de la guerra'. The prompt then returns to 'PS C:\Users\David>'.

```
minusculas.ps1 X
1 $fraseMayusculas = Read-Host "Pon una frase en mayúsculas"
2 $fraseMinusculas = $fraseMayusculas.ToLower()
3 Write-Host "La frase en minúsculas es: $fraseMinusculas"

PS C:\Users\David> C:\Users\David\Desktop\minusculas.ps1
Pon una frase en mayúsculas: EL ARTE DE LA GUERRA
La frase en minúsculas es: el arte de la guerra

PS C:\Users\David>
```

Figura 13. Exercici 10.