

ACTIVIDADES: SCRIPTS

1. ¿Qué es un script y cuál es su función principal?
2. ¿Qué significa IES?
3. Diferencias entre PowerShell y PowerShell IES
4. ¿Cuál puede ser la causa de no poder ejecutar un script en PowerShell?
5. Comprobar los niveles de seguridad
6. Como se cambian los permisos. Ejemplo
Donde NivelDeSeguridad puede ser
 - **restricted**: no permite ejecutar script
 - **allsigned**: todos los scripts tienen que estar autenticados
 - **unrestricted**: permite ejecutar cualquier script
 - **remoteSigned**: Permite ejecutar nuestros scripts, pero no los remotos.
7. ¿Cómo creamos los scripts en PowerShell?
8. Variables:
 - a. Definición teórica y sintáctica
 - b. Acceso a la información
 - c. Asignación de información
 - d. Declaración con un tipo de dato
 - e. ¿Distingue mayúsculas?
 - f. ¿Qué es el scope de una variable?
 - g. Borrar el contenido de una variable y borrar variable
Clear-variable -name nombreVariable
Remove-variable -name nombreVariable
 - h. Tipos de variables con ejemplos
 - i. ¿En qué se parece y en que se diferencia de las constantes?
9. ¿Cuál es la función de los comentarios en un script? Que tipos de comentarios existen y con qué símbolo se representan. Pon un ejemplo de cada
10. Cmdlet para interactuar con el usuario, pedirle informacion y mostrarle informacion.
11. Realiza un script documentado(comentarios) para Crear dos variables:
 - Asignándole el valor 20, e indicando el tipo entero
 - Asignándole el tipo carácterMuestre el contenido de las variables
Muestra el tipo de datos de las variables
Muestra su valor

12. Realiza un script documentado(comentarios) para pedirle al usuario su nombre y su edad mostrándole “tu nombre es ... y tienes ... años. Siendo los ... el nombre y la edad del usuario.
13. Crear un script para saludar al usuario. Mostrara
 - a. El Saludo: se proporciona con argumentos \$args[0] y puede ser :
 - i. Buenos días
 - ii. Buenas tarde
 - iii. Buenas noches
 - b. El nombreUsuario: se proporciona con argumentos \$args[1]
14. Idem que el anterior pero con parámetros Param()
15. Modifica el anterior para que, si no se introduce saludo, se muestre **Buenos Días**
16. Idem que el anterior, pero pidiendo los datos al usuario en tiempo de ejecución
17. Crea un diccionario llamado ordenador que guarde algunas características de los ordenadores:
 - a. Marca
 - b. So
 - c. Procesador

Después mostrara las marcas, so y el procesador de los ordenadores que tenga. Usa variable.elemento (\$ordenador.marca)

18. Realiza un script documentado(comentarios) para mostrar el nivel de seguridad que tiene el sistema a fecha de hoy, es decir, tendría que mostrar por pantalla:

Hoy día “fecha” el nivel de seguridad del sistema es

NOTA: Recordad el uso de: **\$()**

19. Crea un script que muestre los números del 10 al 20
20. ¿Cuál es el operador de concatenación?
21. En un script crea las siguientes variables y concaténalas en una sola llamada S4.

\$ s1= "Hola"

\$ s2= “Esto es PowerShell ISE y; ”

\$ s3= "Está concatenando cadenas:"

Verifica que se ha realizado bien la concatenación S4

22. Pedir al usuario el lado de un cuadrado y calcula su perímetro. Almacena los datos en una estructura que te permita guardar la siguiente información:

Lado1 = 5

Lado2 = 5

Lado3 = 5

Lado4 = 5

Perímetro = Resultado

23. Modifica el script anterior para que muestre los datos de forma ordenada, para ello utiliza [PScustomObjet]
24. Idem que el anterior, pero calculando el área de un rectángulo
25. Script que pida dos números y diga cual es el mayor
26. Script que pida 3 números y diga cual es mayor
27. Script que pida un número y diga si es Negativo o positivo
28. Script para pedir usuario y contraseña. Si coincide con "Pepe" y "abcde" indique has entrado en el sistema sino mostrará usuario no reconocido.
29. Script que compruebe si estamos en el mes de mayo. Informando Estamos en el mes de mayo o No estamos en el mes de mayo.
30. Realizar un script que escoja un número aleatorio entre 1 y 20, pregunte al usuario y le diga si es más pequeño o más grande.
31. Realizar un script que escoja un número aleatorio entre 1 y 20, pregunte al usuario y le diga si es más pequeño o más grande, que continúe hasta que acierte. Haz que muestre el número de intentos hasta que lo acierta cuando finalice. Utiliza Get-Random(rango)
32. Realizar un script que pide un número al usuario y muestre su tabla de multiplicar tabla_multiplicar.ps1
33. Desarrolla un cmdlet de powershell que
 - Defina un array de 4 nombres
 - Añada 2 nombres mas
 - Muestre por pantalla
 - Formas cortas (usando variable como objeto)
 - Con un while, con un for y con un foreach.
34. - Crea un script con:
 - un array de 10 números inicializados a 0 (todos los números del array son 0).
 - Inicialízalos con números aleatorios entre 10 y -10. Utiliza el cmdlet Get-Random para conseguir esos números. Usa length y bucle
 - Contabiliza cuántos números positivos, negativos e iguales a 0 hay en ese array. Usa bucle e If
35. Script que pida 10 números y los muestre en orden descendente

36. Utilizando arrays (para guardar los caracteres que pueden usar la contraseña) y números aleatorios, desarrolla un script de powershell que genere contraseñas. Max longitud 20.
37. Crea un cmdlet de powershell que espere tres parámetros, un llamado \$primero que espere un [int], otro llamado \$segundo que espere un String que si no se introduce debe solicitarlo al usuario. Una vez hecho el programa debe hacer un bucle for que muestre el valor de \$segundo tantas veces como indique \$primero.

Recuerda los parámetros se definen

Param(

[tipo]Parametro1,

[tipo]Parametro2.

....

)

38. **Crear una calculadora muy básica con PowerShell la cual solo tendrá cuatro posibles operaciones: suma, resta, multiplicación y división. Usa una función llamada "operar" a la que se pasaremos dos números y una operación. Usa switch para realizar operación dependiendo de la operación pasada**
39. Crea un cmdlet de PowerShell que defina un array relacional con usuarios. Los campos de cada elemento deben ser nombre_cuenta, nombre y password. El programa debe mantener ese array con un menú en el que se pueda:

-añadir

-eliminar

-buscar por nombre

-mostrar todos

Al crear uno nuevo deben solicitarse datos y el campo password debe ocultarse al escribir.

40. Realizar un script que reciba como único parámetro un nombre de fichero de texto, y que escriba en la salida estándar, para cada línea de dicho fichero, la lista de palabras de la misma, todas en minúsculas, separadas por espacios, ordenadas alfabéticamente, sin repeticiones, y sin ningún tipo de carácter especial (puntuación, etc.).

El guión debe comprobar la corrección de la llamada (esto es, que recibe sólo un parámetro, y que éste es un fichero existente).

Ejemplo:

```
$ cat quijote.txt
```

En un lugar de la Mancha, de cuyo nombre no quiero acordarme,
no ha mucho tiempo que vivía un hidalgo de los de lanza en astillero,
adarga antigua, rocín flaco y galgo corredor.

Una olla de algo más vaca que carnero, salpicón las más noches,
duelos y quebrantos los sábados, lentejas los viernes,
algún palomino de añadidura los domingos,
consumían las tres partes de su hacienda.

```
$ .\09_ordenar.sh quijote.txt
```

acordarme cuyo de de en la lugar mancha no nombre quiero un
astillero de de en ha hidalgo lanza los mucho no que tiempo un vivía
adarga antigua corredor flaco galgo rocín y
algo carnero de las más más noches olla que salpicón una vaca
duelos lentejas los los quebrantos sábados viernes y
algún añadidura de domingos los palomino
consumían de hacienda las partes su tres

41. Tenemos un fichero csv llamado ficheroCSV2.csv con la siguiente información:

Juan, Palomo, jpalomo@gangrena.com, 6593456

Muerte, Dolor, averno@infernial.com, 1890456

Luis, Barcenás, luiselcabron@populares.es, 1456239

Satan, Txiki, satanko@dragoibola.eh, 5673926

Imaginemos que queremos toda la información de cada registro y además el email repetido:

42. Realizar un script que muestre 5 números aleatorios del 1 al 10 seguidos de el mismo número de asteriscos,

1 *

3 **

1 *

6 *****

2 **

OPCIONALES

1. Script que pida un número y diga si es par o impar.
2. Crea una aplicación que permita adivinar un número. La aplicación genera un número aleatorio del 1 al 100. A continuación, va pidiendo números y va respondiendo si el número a adivinar es mayor o menor que el introducido, además de los intentos que te quedan (tienes 10 intentos para acertarlo). El programa termina cuando se acierta el número (además te dice en cuantos intentos lo has acertado), si se llega al límite de intentos te muestra el número que había generado.
3. Algoritmo que pida números hasta que se introduzca un cero. Debe imprimir la suma y la media de todos los números introducidos.
4. Realizar un algoritmo que pida números (se pedirá por teclado la cantidad de números a introducir). El programa debe informar de cuantos números introducidos son mayores que 0, menores que 0 e iguales a 0.
5. Crear una función que calcule la temperatura media de un día a partir de la temperatura máxima y mínima. Crear un programa principal, que, utilizando la función anterior, vaya pidiendo la temperatura máxima y mínima de cada día y vaya mostrando la media. El programa pedirá el número de días que se van a introducir.
6. Crea un script que me permita elegir un tipo de operación:
 - a. Área
 - b. Perímetro

Quando elija una de las opciones tendrá que elegir si quiere realizar la operación sobre cuadrados, círculos o rectángulos

7. Se quiere realizar un programa que lea por teclado las 5 notas obtenidas por un alumno (comprendidas entre 0 y 10). A continuación debe mostrar todas las notas, la nota media, la nota más alta que ha sacado y la menor.(array)
8. Idem pero guardando asignatura nota.
9. Queremos guardar la temperatura mínima y máxima de 5 días. Realiza un programa que de la siguiente información:
 - La temperatura media de cada día

- Los días con menos temperatura
- Se lee una temperatura por teclado y se muestran los días cuya temperatura máxima coincide con ella. si no existe ningún día se muestra un mensaje de información.