

ejercicio 1_

ponemos echo es un comando que se utiliza para mostrar texto y decimos buenas noches y despues pedimos que ingrese un texto

despues ponemos el read que es para leer la entrada ya asignarla a una variable en este caso a la variable texto

por ultimo utilizamos el echo para mostrar el texto ingresado y con la variable texto con el valor que el usuario le dio el la linea anterior en este caso buenas noches

ejercicio 2

empezamos igual pero ahora con un echo que nos pide un nombre para un archivo

con el read lo leemos y lo almasenamos en la variable

definimos una variable llamada texto_predeterminado y le asignamos el valor de

archivo_predeterminado

la linea nombre_completo=\${nombre_usuario}\${texto_predeterminado}.txt crea un nombre combinando el valor de la variable texto_predeterminado y nombre_usuario con la extencion txt y luego asigna este nomrbre completo a la variable nombre_completo

usamos el touch para crear un archivo vacio con el nombre almacenado de la variable nombre_completo el ultimo echo muestra que el archivo se creo

ejercicio 3

el -p permite mostra el mensaje ingrese una palabra antes que el usuario ingrese su dato

date -d \$fecha1 +%j devuelve el número del día del año para la fecha proporcionada

el lt verifica si la diferencia_dias es menor que 0

ejercicio 4

el -p permite mostra el mensaje ingrese una palabra antes que el usuario ingrese su dato

la opcion -n evita que echo añada un salto de linea al final de la salida lo cual nos ayuda a obtener una cuenta precisa de los caracteres

wc es un word count agregandole el -c cuenta los caracteres en la entrada

no encuentre la solucion y termine buscando en internet un hasta que encuentre el wc y lo termien usando

ejercicio 5

con el es_primo definimos la funcion verificas

si el número es menor que 2. Si esta condición es verdadera, se imprime "No es primo" y se sale de la función con return se incia un bucle for

la condicion $i*i \leq \text{numero}$ asegura que solo se verifiquen divisores hasta la raiz cuadrada

ejercicio 6

se utiliza if para verificar si el numero es par el operador% calcula el residuo de dividir numero entre 2 si el residuo es 0 el numero es par

si este if no es verdadero se ejucta el else que imprime el numero es impar

ejercicio 7

Utiliza el comando echo para pasar la frase a tr, que transforma los caracteres en mayúsculas ([:upper:]) a minúsculas ([:lower:]). El resultado se guarda en la variable frase_minusculas

tr se usa para eliminar o cambiar caracteres

ejercicio 8

si la condicion if no es verdadera el bloque elif verifica si num1 es menor que num2 usando el operador -lt (menos que)

ejercicio 9

en la division primero verificamos con -ne que num2 no sea igual a 0 calculamos estableciendo la escala a 2 decimales

bc es una herramienta en la línea de comandos que realiza cálculos matemáticos con una precisión que puedes ajustar según tus necesidades

ejercicio 10

es igual que el 9 pero con menos operaciones

ejercicio 11

todo lo echo esta explicado en los ejercicios anteriores

ejercicio 12

El primer if lo usamos para verificar si la nota es menor que 5 con -lt si es v asignamos el insuficiente si la primera no es verdadera el bloque siguiente con el elif verifica si la nota es mayo o igual a 5 y menor que 7 el -ge significa (mayor o igual que) asi con los demas hasta que da la nota

ejercicio 13

definimos la credenciales correctas para la autenticacion

con el read -p se solicita el nombre que se almacena en (usuario)y la contraseña (que se almacena en contrasena ya que con contraseña da error)

se agrega un ech para imprimir una nueva linea despues de la entrada de la contraseña

hacemos lo mismo que en el 7 para transformar las mayusc en minusc

en el if verifica si el nombre ingresado coincide con el nomvbre de usuario correcto y lo mismo con la contraseña