Welcome to Day 1 of the Bash Scripting Challenge! Today, we will cover the basics of bash scripting to get you started. Let's dive in:

#!/bin/bash

###Task 1: Comments

In bash scripts, comments are used to add explanatory notes or disable certain lines of code. Your task is to create a bash script with comments explaining what the script does.###

#!/bin/bash

# This is a single-line comment

# that spans multiple lines

# but each line is treated as a separate comment.

echo "This is the main part of the script."

# You can also "comment out" blocks of code

# by adding # at the beginning of each line.

# This effectively disables the code.

# echo "This line will not be executed."

echo "Script execution completed."

: <<'COMMENT'

This is a multiline comment

that spans several lines

and explains the purpose of the script

COMMENT

echo "Hello, world!"

Task 2: Echo

The echo command is used to display messages on the terminal. Your task is to create a bash script that uses echo to print a message of your choice.

echo”This is Rubeena khan ”

Task 3: Variables

Variables in bash are used to store data and can be referenced by their name. Your task is to create a bash script that declares variables and assigns values to them.

#!/bin/bash

# Declare variables and assign values

name="John"

age=30

city="New York"

# Print the values of the variables

echo "Name: $name"

echo "Age: $age"

echo "City: $city"

Task 4: Using Variables

Now that you have declared variables, let's use them to perform a simple task. Create a bash script that takes two variables (numbers) as input and prints their sum using those variables.

#!/bin/bash

# Prompt the user to enter the first number

read -p "Enter the first number: " num1

# Prompt the user to enter the second number

read -p "Enter the second number: " num2

# Perform the addition operation

sum=$((num1 + num2))

# Print the result

echo "The sum of $num1 and $num2 is: $sum"

Task 5: Using Built-in Variables

Bash provides several built-in variables that hold useful information. Your task is to create a bash script that utilizes at least three different built-in variables to display relevant information.

#!/bin/bash

# Get the current user's username

username=$(whoami)

# Get the current date

current\_date=$(date +"%Y-%m-%d")

# Get the current working directory

current\_directory=$(pwd)

# Display relevant information

echo "Hello, $username!"

echo "Today is $current\_date."

echo "You are currently in the directory: $current\_directory"

Task 6: Wildcards

Wildcards are special characters used to perform pattern matching when working with files. Your task is to create a bash script that utilizes wildcards to list all the files with a specific extension in a directory

.#!/bin/bash

# Prompt the user to enter the directory path

read -p "Enter the directory path: " directory\_path

# Prompt the user to enter the file extension

read -p "Enter the file extension (e.g., txt, jpg, etc.): " file\_extension

# List all files with the specified extension in the given directory

echo "Files with the .$file\_extension extension in $directory\_path:"

ls "$directory\_path"/\*."$file\_extension"