

Introduction to shell scripting

How to create a shell file?

- Create a file with a `.sh` extension
- Add shebang at the beginning of the file

```
#!/bin/bash
```

- Make the file executable with

```
chmod +x <file-name>.sh
```

- Run the shell

```
./<file-name>.sh
```

How to print in a shell script?

```
#!/bin/bash  
echo "Hello World!!!"
```

How to pass input?

```
#!/bin/bash  
echo "Pass an input"  
read user_input  
echo $user_input
```

How to declare a variable?

```
#!/bin/bash  
variable="Hello World!!!"
```

```
echo $variable
```

How to read environment files in a shell script?

```
ENV_FILE=".env"
```

Function

```
#!/bin/bash
func(){
    echo "Hello World!!!"
}

func
```

Parameterized Function

```
#!/bin/bash
add_numbers(){
    local num1="$1"
    local num2="$2"
    local sum=$((num1+num2))
    echo "The sum of $num1 and $num2 is: $sum"
}

add_numbers 10 20
```

if-else

```
#!/bin/bash

echo "Enter a number:"
read number

if [ "$number" -eq 100 ]; then
```

```
    echo "The number is 100."
else
    echo "The number is not 100."
fi
```

Nested if-else

```
#!/bin/bash
echo "Please enter a number:"
read user_input
if [ "$user_input" -gt 10 ]; then
    echo "The entered number is greater than 10."
else
    if [ "$user_input" -lt 10 ]; then
        echo "The entered number is less than 10."
    else
        echo "The entered number is equal to 10."
    fi
fi
```

Case

```
#!/bin/bash
read -p "Enter a fruit name: " fruit
case "$fruit" in
    "apple")
        echo "You selected an apple."
        ;;
    "banana")
        echo "You selected a banana."
        ;;
    *) # Code to execute if $variable doesn't match
        echo "Unknown fruit."
        ;;
esac # Signal end of case statement
```

For Loop

```
#!/bin/bash
# Define an array of fruits
fruits=("apple" "banana" "cherry" "date" "fig")
# Iterate through the array using a for loop
for fruit in "${fruits[@]}; do
    echo "I like $fruit."
done
```

While Loop

```
#!/bin/bash
counter=1
max=5
# Use a while loop to count from 1 to 5
while [ "$counter" -le "$max" ]; do
    echo "Count: $counter"
    counter=$((counter + 1))
done
```

Until Loop

```
#!/bin/bash
count=1
until [ $count -gt 5 ]
do
    echo "Count: $count"
    count=$((count + 1))
done
```