Bash String

In this topic, we have demonstrated about bash string and its operators.

Like other programming languages, Bash String is a data type such as as an integer or floating-point unit. It is used to represent text rather than numbers. It is a combination of a set of characters that may also contain numbers.

For example, the word "javatpoint" and the phrase "Welcome to javatpoint" are the strings. Even "01234" could be considered as a string, if specified correctly. Programmers are required to enclose strings in quotation marks for the data to be considered as a string and not a number, variable name or array, etc.

Bash consists of multiple ways to perform string operations and manipulate them.

Following are some operators in Shell Script used to perform string operations:

Equal Operator

An equal operator (=) is used to check whether two strings are equal.

Syntax

```
Operand1 = Operand2
```

Example

```
#!/bin/bash
#Script to check whether two strings are equal.

str1="WelcometoJavatpoint."
str2="javatpoint"
```

```
if [ $str1 = $str2 ];
then
echo "Both the strings are equal."
else
echo "Strings are not equal."
fi
```

Output

Strings are not equal.

Not Equal Operator

Not equal operator (!=) is used to define that strings are not equal.

Syntax

1. Operand1 != Operand2

Example

```
#!/bin/bash
#Script to check whether two strings are equal.

str1="WelcometoJavatpoint."
str2="javatpoint"

if [[ $str1 != $str2 ]];
then
echo "Strings are not equal."
else
echo "Strings are equal."
fi
```

Output

Strings are not equal.

Less than Operator

The 'less than operator (\<)' is a conditional operator which is used to check if string1 is less than string2.

Syntax

1. Operand1 \< Operand2

Example

```
#!/bin/sh

str1="WelcometoJavatpoint"
str2="Javatpoint"
if [ $str1 \< $str2 ];
then
   echo "$str1 is less then $str2"
else
   echo "$str1 is not less then $str2"
fi
```

Output

WelcometoJavatpoint is not less then Javatpoint

Greater than Operator

The 'greater than operator (\>)` is used to check if string1 is greater than string2.

Syntax

1. Operand1 \> Operand2

Example

```
#!/bin/sh

str1="WelcometoJavatpoint"
str2="Javatpoint"
if [ $str1 \> $str2 ];
```

```
then
   echo "$str1 is greater then $str2"
else
   echo "$str1 is less then $str2"
fi
```

Output

WelcometoJavatpoint is greater then Javatpoint

To check if the string length is greater than Zero:

This operator is used to check if the string is zero or greater than zero.

Syntax

1. [-n Operand]

Example

```
#!/bin/sh

str="WelcometoJavatpoint"

if [ -n $str ];
then
   echo "String is not empty"
else
   echo "String is empty"
fi
```

Output

String is not empty

To check if the string length is equal to Zero

This operator is used to check if the string is empty or equal to zero.

Syntax

1. [-z Operand]

Example

```
#!/bin/sh

str=""

if [ -z $str ];
then
   echo "String is empty."
else
   echo "String is non-empty."
fi
```

Output

String is empty