Bash Concatenate String

In this topic, we have explained how to add or concatenate strings in Bash Shell Scripting.

In bash scripting, we can add or join two or more strings together, which is known as string concatenation. It is one of the common requirement for any programming language. A special character or built-in function is applied to perform string concatenation. However, Bash does not contain any built-in function to combine string data or variables. The easiest method to perform string concatenation in bash is to write variables side by side.

For example, assume that we have two strings (i.e., "welcome" & "to javatpoint"), and we join both the strings together and a new string ("welcome to javatpoint") is created. This concept is referred to as String Concatenation.

Command

The example command for concatenating the strings can be defined as:

```
1. str3="$str1$str2"
```

Note: Observe the above command; there should not be any space before or after the assignment (=) operator. 'str' is used to indicate strings.

This command will concatenate the values of str1 and str2 and store it in a third variable str3.

Following are some examples demonstrating the different ways of string concatenation:

Example 1: Write Variables Side by Side

This is the basic example of String Concatenation, and we do not need any extra operator or function in this method.

Bash Script

```
#!/bin/bash
#Script to Concatenate Strings
#Declaring the first String
str1="We welcome you"
#Declaring the Second String
str2=" on Javatpoint."
#Combining first and second string
str3="$str1$str2"
#Printing a new string by combining both
echo $str3
```

Output

We welcome you on Javatpoint.

Example 2: Using Double Quotes

Another easy method is to use variables inside the string, which is defined with double-quotes. The string variable can be applied in any position of the string data.

Bash Script

```
#!/bin/bash
#Script to Concatenate Strings
#Declaring String Variable
str="We welcome you"
#Add the variable within the string
echo "$str on Javatpoint."
```

Output

We welcome you on Javatpoint.

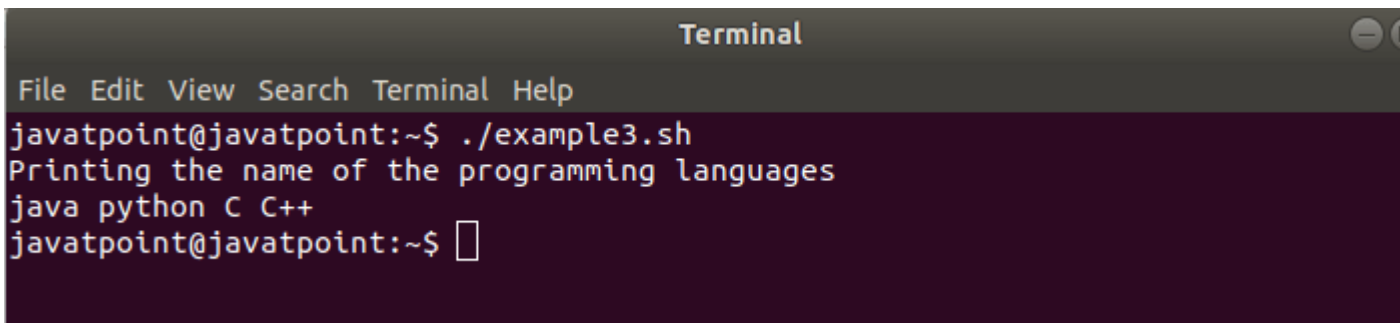
Example 3: Using Append Operator with Loop

Most of the popular programming languages provide support for append operator (+=) which is the combination of the plus and equal sign. It will add new strings to the end of the string variable.

Bash Script

```
#!/bin/bash
echo "Printing the name of the programming languages"
#Initializing the variable before combining
lang=""
#for loop for reading the list
for value in 'java''python''C''C++';
do
lang+="$value "  #Combining the list values using append operator
done
#Printing the combined values
echo "$lang"
```

Output



Example 4: Using the Printf Function

In bash, **printf** is a function which is used to print and concatenate the strings.

Bash Script

```
#!/bin/bash
str="Welcome"
printf -v new_str "$str to Javatpoint."
echo $new_str
```

Output

Welcome to Javatpoint.

Example 5: Using Literal Strings

String concatenation can also be performed with a literal string by using curly braces{}. They should be used in such a way that the variable does not mix up with the literal string.

Bash Script

```
#!/bin/bash
str="Welcome to"
newstr="${str} Javatpoint."
echo "$newstr"
```

Output

Welcome to Javatpoint.

Example 6: Using Underscore

Using underscore for concatenating the string in bash shell is one of the common tasks. It is mostly used for assigning a name to the files.

Bash Script

```
#!/bin/bash
str1="Hello"
str2="World!"
echo "${str1}_${str2}"
```

Output

Hello_World!

Example 7: Using any Character

Bash Script

```
#!/bin/bash
#String Concatenation by Character (,) with User Input
read -p "Enter First Name: " name
read -p "Enter State: " state
read -p "Enter Age: " age
combine="$name,$state,$age"
echo "Name, State, Age: $combine"
```

Output