**Variable Scopes in Shell scripts**

Lets create two shell scripts >> script1.sh

#!/bin/bash

name="Learning Thoughts"

echo "This is from script 1 name= ${name}"

./script2.sh

Script2.sh

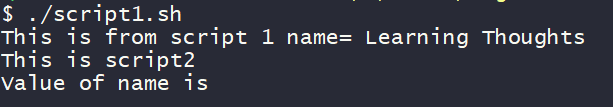
#!/bin/bash

echo "This is script2"

echo "Value of name is ${name}"

From script1 lets call script2 and try to see if the variable defined in script1 is available for use in script2

let’s execute script1.sh



The value of name is not available in script2. The default scope of variable is the same script file.

So let’s find out if there is any other way to pass the variable from script1 to external scripts,

add Arguments to script2

./script2.sh

# replace this with

./script2.sh $name

export the variable using export statement in script1. so lets change script1.sh

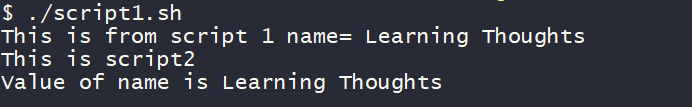
#!/bin/bash

name="Learning Thoughts"

echo "This is from script 1 name= ${name}"

export name

./script2.sh



**Debugging your scripts**

As the script grows and decision paths are included with conditional statements, we start using looping structures etc, we may need some level of debugging to analyse the scripts.

This can be done as bas provides two options for us

-v option

-x option

Create a bash script debugdemo.sh with the following content

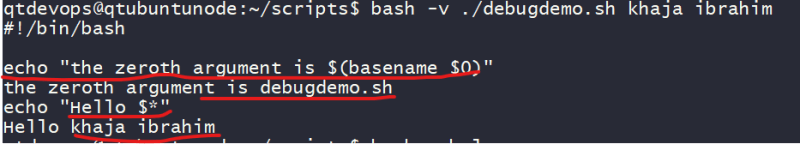
#!/bin/bash

echo "the zeroth argument is $(basename $0)"

echo "Hello $\*"

Now execute this script with the following command

bash -v ./debugdemo.sh khaja ibrahim

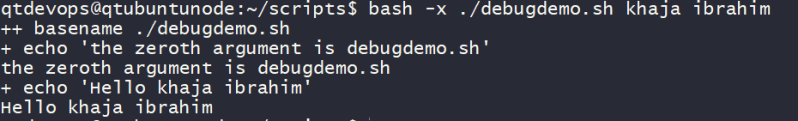


-v option displays the verbose output from bash

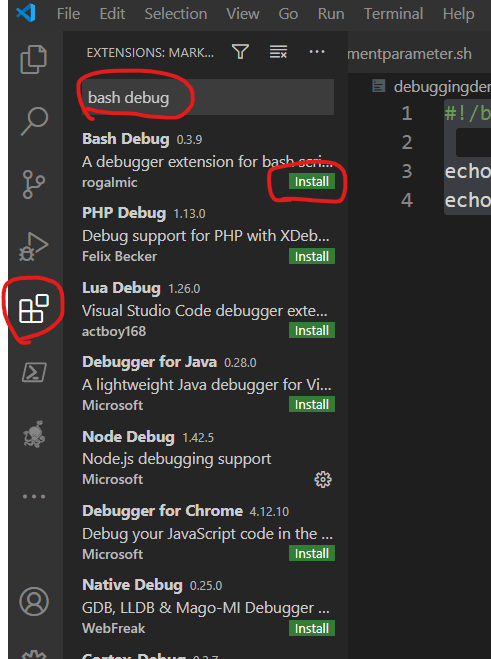
-x option, which displays the commands as they are executed and is most commonly used.

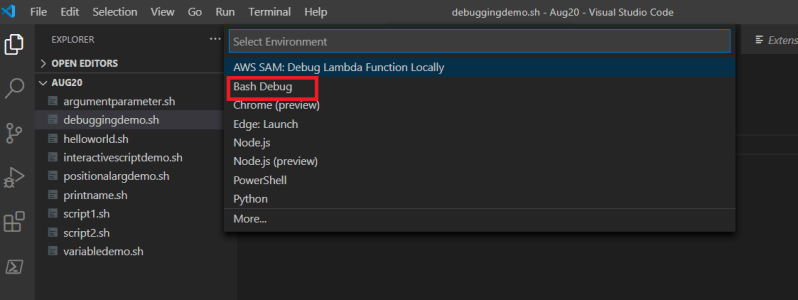
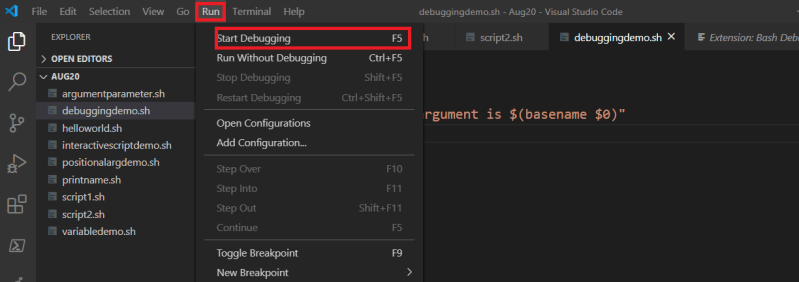
Now lets run the script

bash -x ./debugdemo.sh khaja ibrahim



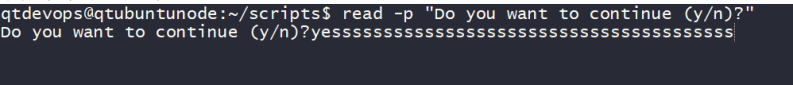
This way shows how the evaluated and is very helpful to know the decision branch that has been chosen by the script.

Install bash *debug plugin* in visual studio code and we can debug the shell script. 

Now we can debug the shell script 

This approach works with visual studio code installed on the linux desktops or mac.

**Enhancing interactive scripts**

Lets understand how to limit number of characters entered 

so limit number of characters to be entered, lets run the same command with one more option Preview

Let’s understand how to control visibility of entered text, If we request for sensitive content like pin/password etc, its not a good approach to show the text while user is typing, Shell gives an option of hiding text when the user is typing. This can be achieved by adding a -s option to read command 