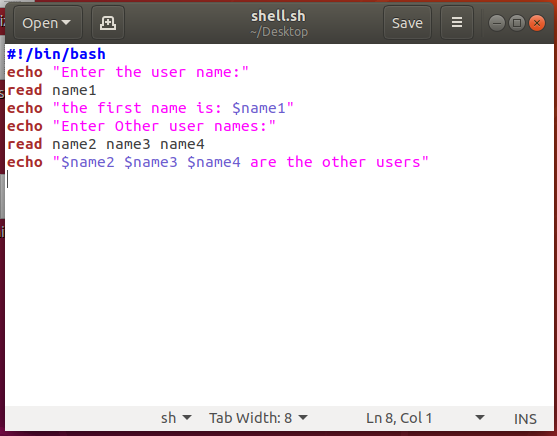
***LAB 05: Read user input in shell scripting***

**Theory:**

* **Read User Input**  
  To read the Bash user input, we use the built-in Bash command called **read**. It takes input from the user and assigns it to the variable. It reads only a single line from the Bash shell.
* Below is the syntax for its implementation.
* **Syntax**

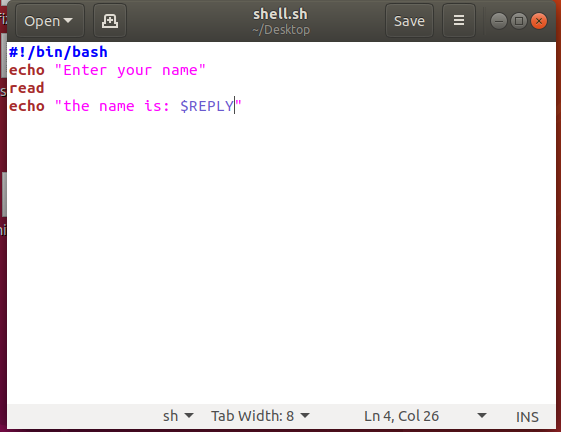
read <variable\_name>    
**Example 1:**

In this example, we read both the single and multiple variables from the Bash Script by using read command.



**What will happen if we don't pass any variable with the read command?**

If we don't pass any variable with the read command, then we can pass a built-in variable called REPLY (should be prefixed with the $ sign) while displaying the input.



**Example 2:**

In this example, we enter the input on the same PROMPT by using the -p command line option as follows:

* Syntax: read -p PROMPT <variable\_name>

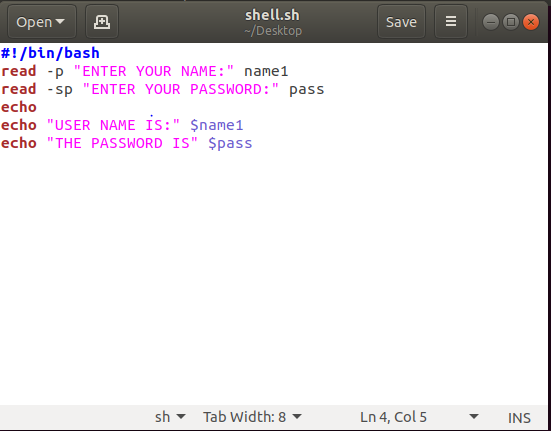
****

**Example 3:**

* This example is to keep the input on silent mode, such that whatever be a user input on the command line will be hidden to others.
* Where -s allows a user to keep the input on silent mode
* -p to input on newly command prompt.

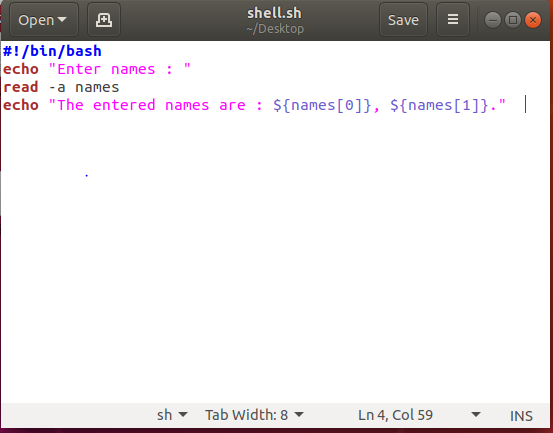
Syntax:

read -sp PROMPT <variable\_name>



**Example 4:**

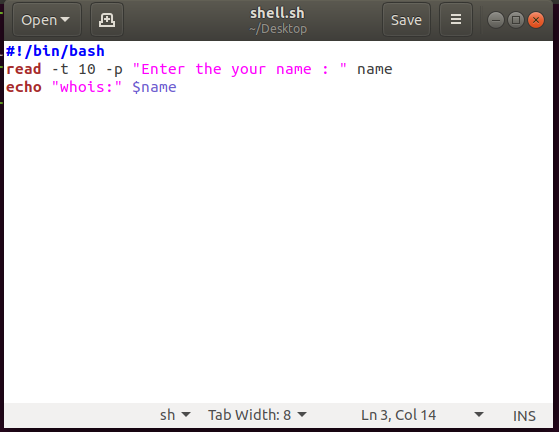
this example is to enter multiple inputs using an array. So use the -a command line option as follows:



**Example 5:**

Timeout Input

* You can time out read command using the -t option.
* It causes read to time out and return failure if a complete line of input is not read within TIMEOUT seconds.
* For example, if no input provided within 10 second, program will be aborted (domain2.sh):



**Lab task**

1. Write a script to display the user bio data by using read command.
2. Write a script to make user login using –sp and –t.
3. Difference between –s and –p in shell scripting.
4. Write a script to find the area of a triangle.
5. Write the functionality of REPLY variable.