



# expect command in Linux with Examples

**expect** command or scripting '  '  user inputs. It automates the task by providing inputs.

[Hire with us!](#)

// We can install expect command using following if not installed

// On Ubuntu

**\$sudo apt install expect**

// On Redhat based systems

**\$ yum install expect**

First we write a script that will be needing inputs from users and then we will write another script using expect to automate the task.

```
#!/bin/bash  
  
echo "Enter your name"  
  
read $REPLY  
  
echo "Enter your age"  
  
read $REPLY
```



```
#!/usr/bin/perl
echo "Enter your salary"

read $REPLY
```

Ok, so let's write down the script to answer questions of the above script.

```
#!/usr/bin/expect -f

set timeout -1

spawn ./que.sh

expect "Enter your name\r"

send -- "I am Nikhil\r"

expect "Enter your age\r"

send -- "24\r"

expect "Enter your salary\r"

send -- "100k\r"

expect eof
```

The first line defines the expect command path which is `#!/usr/bin/expect`.

On the second line of code, we disable the timeout. Then start our script using spawn command.

We can use spawn to run any program we want or any other interactive script. The spawn command is used to start a script or a program like the shell, FTP, Telnet, SSH, SCP, and so on.

The remaining lines are the Expect script that interacts with our shell script.

The last line is the end of file which means the end of the interaction.

First we have to make the file executable and then run it. You can do this by typing the following:

```
$ chmod +x ./ans.sh
$ ./ans.sh
```

This will produce an output like this :

**Output:**



```
demon@AJ7:~$ ./ans.sh
spawn ./que.sh
Enter your name
I am Nikhil
Enter your age
24
Enter your salary
100k
demon@AJ7:~$ █
```

## autoexpect

To build an expect script automatically, you can use the autoexpect command. autoexpect works like expect, but it builds the automation script for you. The script you want to automate is passed to autoexpect as a parameter and you answer the questions and your answers are saved in a file.

```
$ autoexpect ./que.sh
```

```
demon@AJ7:~$ autoexpect ./que.sh
autoexpect started, file is script.exp
Enter your name
Nikhil
Enter your age
24
Enter your salary
40000
autoexpect done, file is script.exp
demon@AJ7:~$ █
```

A file is generated called script.exp contains the same code as we did above with some additions. if you run the file script.exp you will see the same answers that you provided.

```
demon@AJ7:~$ ./script.exp
spawn ./que.sh
Enter your name
Nikhil
Enter your age
24
Enter your salary
40000
demon@AJ7:~$ █
```

---

### Recommended Posts:

- [until command in Linux with Examples](#)
- [w command in Linux with Examples](#)
- [more command in Linux with Examples](#)
- [top command in Linux with Examples](#)
- [atd command in Linux with examples](#)
- [col command in Linux with Examples](#)
- [cp command in Linux with examples](#)
- [bc command in Linux with examples](#)
- [ed command in Linux with examples](#)
- [df Command in Linux with examples](#)

