

# Lab 6: Using Variables

Estimated time for completion: **10 minutes**

## Requirements

The following tasks must be completed before beginning this lab:

- Getting Started with NGINX, (the Getting Started Guide in LearnF5)
- Log into Hosted Environment, your lab initialization instructions are located in the LearnF5 course

## Scenario

In this exercise, create a server block with three locations to test variable configurations and behavior.

## Objectives

At the end of this lab you will be able to:

- Create a configuration file with variables
- Determine effective behavior of variables within different scopes

## Lab Contents

Exercise 1: Create a `variable_test.conf` configuration file and test it.



### IMPORTANT

You can copy and paste the commands and text from the examples to your terminal or editor, (just make sure you don't copy and paste the \$ prompt!)

## Exercise 1: Create a `variable_test.conf` configuration file and test it.

1. Rename your `mywebserver.conf` file to `mywebserver.bak` to make it active:

```
$ cd /etc/nginx/conf.d
$ sudo mv mywebserver.{conf,bak}
```

2. Create a new configuration file as follows:

```
$ sudo vim variable_test.conf
```

3. Add the following configuration:

```
server {
    listen 80;
    root /home/ubuntu/public_html;

    location / {
        return 200 "custom variable is $custom_variable \n";
    }

    location /test1 {
        set $custom_variable 42;
        return 200 "custom variable is $custom_variable \n";
    }

    location /test2 {
        set $custom_variable "null";
        return 200 "custom variable is $custom_variable \n";
    }
}
```

4. Save the file and reload NGINX. (`esc` and `:wq`).

```
$ sudo nginx -s reload
```

5. Test your configuration in the terminal with the following commands:

```
$ curl http://localhost/test1
$ curl http://localhost/test2
```

```
$ curl http://localhost/test
$ curl http://localhost
```

The last 2 curl commands match the catch all `location /` block which does not set a value for the variable `$ custom_variable`.

```
NGINX$ curl http://localhost/test1 custom variable is 42
NGINX$ curl http://localhost/test2 custom variable is null
NGINX$ curl http://localhost/test custom variable is
NGINX$ curl http://localhost/ custom variable is
NGINX$
```

6. Edit the `variable_test_conf` file and set a value of zero for the variable `$ custom_variable` in the server location:

```
$ sudo vim variable_test.conf
```

```
server {
    listen 80;
    root /home/ubuntu/public_html;
    set $custom_variable 0;
    ...
}
```

7. Save the file and reload NGINX. (`esc` and `:wq`).

```
$ sudo nginx -s reload
```

8. Test the same endpoints What changed? What did not? Why?

```
$ curl http://localhost/test1
$ curl http://localhost/test2
$ curl http://localhost/test
$ curl http://localhost
```

```
NGINX$ curl http://localhost/test1 custom variable is 42
NGINX$ curl http://localhost/test2 custom variable is null
NGINX$ curl http://localhost/test custom variable is 0
NGINX$ curl http://localhost/ custom variable is 0
```

The last 2 curl commands STILL match the catch all `location` / block which now inherits the value 0 for the variable `$ custom_variable`.

## Expected Results

In these exercises, you were able to create a server block with three locations to test variable configurations and behavior.

