



---

*NGINX: Lab - Rewrite and Return*

## NGINX : Web-Server & Load Balancer

---

- **Return & Rewrite:** Both for one of two purposes-
  - First, if a URL has changed.
  - To control the request within Nginx.  
For example, a request can be forwarded to an application if the content will be generated dynamically.
- **Return Directive** - Return is the simpler directive to use compared to NGINX rewrite.
- Return must be enclosed within a **server** or **location** block which defines which URLs should be rewritten.

```
server {  
    listen 80;  
    server_name www.old-website.com;  
    return 301 $scheme://www.new-website.com$request_uri;  
}
```

# NGINX : Web-Server & Load Balancer

---

- **Rewrite Directive** - This directive needs to be in a **location** or **server** block in order to rewrite the URL.
- Rewrite directive can be used to perform more granular tasks as with it you can perform more complicated URL distinctions such as:
  - Capture elements in the original URL
  - Change or add elements in the path
- Syntax :  
`rewrite regex URL [flag];`
- Rewrite directive does not send a redirect to the client in all cases. If the rewritten URL matches with another following directive, Nginx will rewrite the URL again.
- A **rewrite** directive will only return an HTTP **301** or **302** status code. If another status code is required, a **return** directive is needed after the rewrite directive.

*Will see you in Next Lecture...*

---

*Thank you!*

A close-up photograph of a hand holding a black marker, completing the cursive word 'Thank you!' on a white surface. The hand is positioned on the right side of the frame, with the index and thumb fingers visible, holding the marker. The marker's tip is just finishing the exclamation point. The background is a plain, light-colored surface.

*See you in next lecture ...*