

Q1.

```
sudo apt-get install nginx -y
```

```
cd //var//www//html//
```

```
vim index.html
```

```
<html>
```

```
<body>
```

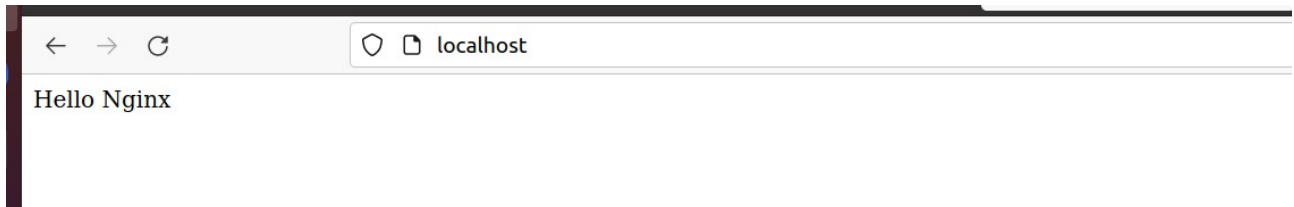
```
<p>Hello Nginx</p>
```

```
</body>
```

```
</html>
```

```
sudo systemctl start nginx
```

```
sudo systemctl enable nginx
```



Q2..

Security headers are directives used by web applications to configure security defenses in web browsers. Based on these directives, browsers can make it harder to exploit client-side vulnerabilities such as Cross-Site Scripting or Clickjacking. Headers can also be used to configure the browser to only allow valid TLS communication and enforce valid certificates, or even enforce using a specific server certificate.

They are directives to increase the protection and create more defense against vulnerabilities using browsers. For example, they modify the behavior of web browsers to avoid security vulnerabilities just to accept one kind of valid server certificate like TLS.

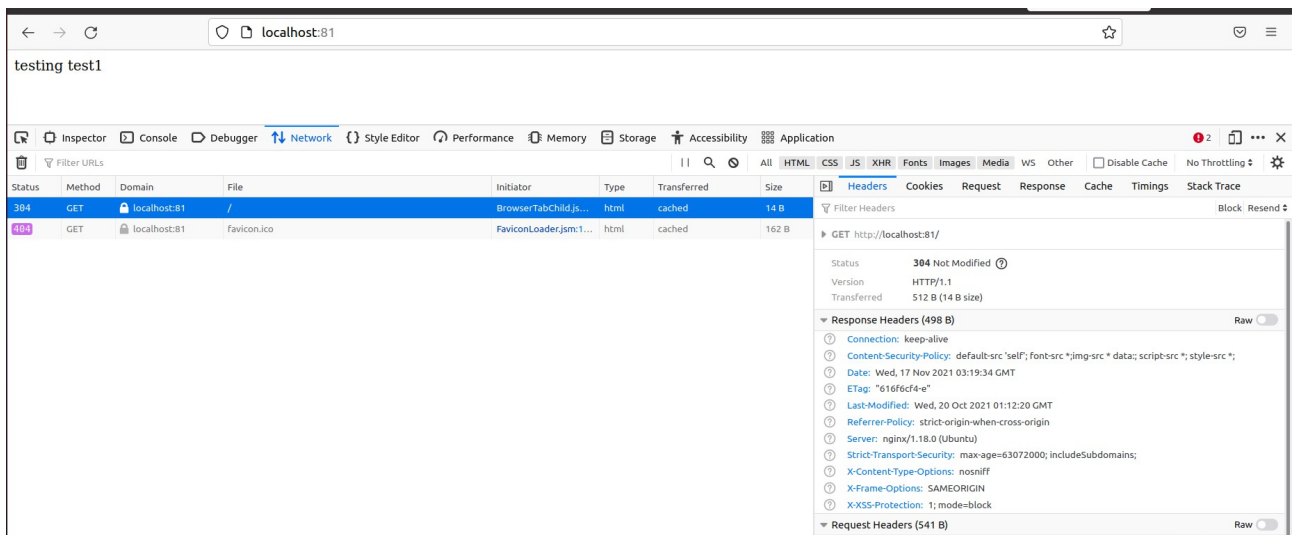
```
mkdir -p /var/www/test1.com/public_html
vim index.html
    testing test1
```

```
cd /etc/nginx/sites-available//
mv default default-bk
cd ../sites-enabled
rm default
```

```
cd /etc/nginx/conf.d//
touch test.conf
    server {
        listen 81;
        listen [::]:81;
        root /var/www/test1.com/public_html;
        index index.html;
        server_name test1.com www.test1.com;
        add_header Strict-Transport-Security "max-age=63072000; includeSubdomains;" always;
        add_header X-Frame-Options "SAMEORIGIN" always;
        add_header X-XSS-Protection "1; mode=block" always;
        add_header X-Content-Type-Options "nosniff" always;
        add_header Content-Security-Policy "default-src 'self'; font-src *;img-src * data;; script-src
        *; style-src *;" always;
        add_header Referrer-Policy "strict-origin-when-cross-origin" always;

        location /test1.com {
            try_files $uri $uri/ =404;
        }
    }
```

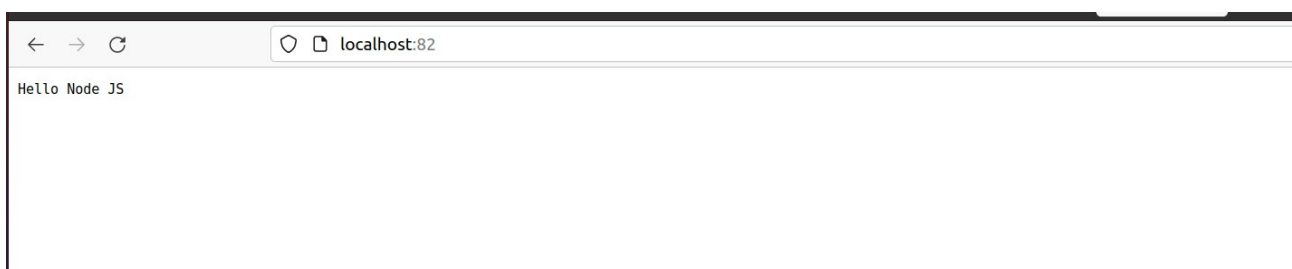
```
nginx -t
systemctl restart nginx
```



Q3.

```
cd //etc//nginx//conf.d//
touch test2.conf
vim test2.conf
server {
    listen 82;
    location / {
        proxy_pass http://localhost:6080;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
        proxy_http_version 1.1;
        proxy_set_header X-Real-IP $remote_addr;
    }
}
```

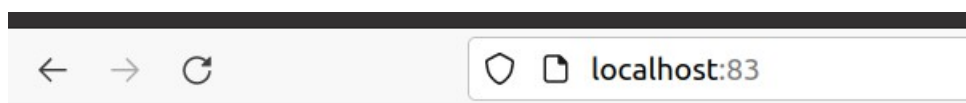
nginx -t
systemctl restart nginx



Q4.

```
cd //var/www//test1.com//public_html//
mkdir -p test2.com/html/
cd test2.com/html/
touch index.html
vim index.html
pwd
cd /etc/nginx/conf.d/
ls
cp test.conf test3.conf
vim test3.conf
server {
    listen 83;
    listen [::]:83;
    root /var/www/test1.com/public_html/test2.com/html;
    index index.html;
    location /{
        try_files $uri $uri/ =404;
    }
}
```

```
nginx -t
systemctl restart nginx
```



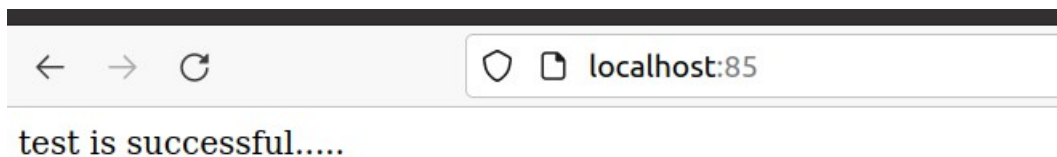
test is successful.....

Q5.

```
cd /etc/nginx/conf.d/  
touch rev-p.conf  
ls  
cat test2.conf  
cp test2.conf rev-p.conf  
vim rev-p.conf
```

```
server {  
    listen 85;  
    location / {  
        proxy_pass http://localhost:83;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection 'upgrade';  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
        proxy_http_version 1.1;  
        proxy_set_header X-Real-IP $remote_addr;  
    }  
}
```

```
nginx -t  
systemctl restart nginx
```



Q6.

```
sudo apt install php-fpm
```

```
cat test.conf
```

```
vim test.conf
```

```
server {
    listen 81;
    listen [::]:81;
    root /var/www/test1.com/public_html;
    index index.html index.php;
    server_name test1.com www.test1.com;
    add_header Strict-Transport-Security "max-age=63072000; includeSubdomains;" always;
    add_header X-Frame-Options "SAMEORIGIN" always;
    add_header X-XSS-Protection "1; mode=block" always;
    add_header X-Content-Type-Options "nosniff" always;
    add_header Content-Security-Policy "default-src 'self'; font-src *;img-src * data;; script-src
    *; style-src *;" always;
    add_header Referrer-Policy "strict-origin-when-cross-origin" always;
```

```
location /test1.com{
    try_files $uri $uri/ =404;
}
```

```
location ~ /\.php$ {
    include snippets/fastcgi-php.conf;
    fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
}
```

```
location ~ /\.ht {
    deny all;
}
```

```
}
```

```
cd /var/www/test1.com/public_html/
```

```
ls
```

```
vim info.php
```

```
<?php
phpinfo();
?>
```

systemctl restart nginx

localhost:81/info.php



PHP Version 7.4.3

System	Linux saroj-Inspiron-3576 5.11.0-40-generic #44~20.04.2-Ubuntu SMP Tue Oct 26 18:07:44 UTC 2021 x86_64
Build Date	Oct 25 2021 18:20:54
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/fpm
Loaded Configuration File	/etc/php/7.4/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/fpm/conf.d
Additional .ini files parsed	/etc/php/7.4/fpm/conf.d/10-opcache.ini, /etc/php/7.4/fpm/conf.d/10-pdo.ini, /etc/php/7.4/fpm/conf.d/20-calendar.ini, /etc/php/7.4/fpm/conf.d/20-ctype.ini, /etc/php/7.4/fpm/conf.d/20-exif.ini, /etc/php/7.4/fpm/conf.d/20-ffi.ini, /etc/php/7.4/fpm/conf.d/20-fileinfo.ini, /etc/php/7.4/fpm/conf.d/20-ftp.ini, /etc/php/7.4/fpm/conf.d/20-gettext.ini, /etc/php/7.4/fpm/conf.d/20-iconv.ini, /etc/php/7.4/fpm/conf.d/20-json.ini, /etc/php/7.4/fpm/conf.d/20-phar.ini, /etc/php/7.4/fpm/conf.d/20-posix.ini, /etc/php/7.4/fpm/conf.d/20-readline.ini, /etc/php/7.4/fpm/conf.d/20-shmop.ini, /etc/php/7.4/fpm/conf.d/20-sockets.ini, /etc/php/7.4/fpm/conf.d/20-sysvmsg.ini, /etc/php/7.4/fpm/conf.d/20-sysvsem.ini, /etc/php/7.4/fpm/conf.d/20-sysvshm.ini, /etc/php/7.4/fpm/conf.d/20-tokenizer.ini
PHP API	20190902
PHP Extension	20190902