[Nginx] 개발 서버 docker compose, nginx.conf 설정

🌟 **작업 디렉토리: /home/ubuntu/dev-server/webserver**

1. docker-compose.yml 파일 생성

```
version : '3.8'

services:
  nginx:
    container_name: nginx-dev
    image: nginx:latest
    restart: always
    volumes:
      - ./proxy/nginx.conf:/etc/nginx/nginx.conf
      - ./proxy/blue_green:/proxy
    environment:
      - TZ=Asia/Seoul
    networks:
      - chaekbang-dev-net
      - chaekbang-webserver-net

networks:
  chaekbang-dev-net:
    external: true
    name: "chaekbang-dev-net"
  chaekbang-webserver-net:
    external: true
    name: "chaekbang-webserver-net"
```

2. proxy/blue_green 디렉토리에 무중단 배포에 사용될 nginx.conf 파일 생성

- **nginx.blue.conf**

```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log warn;
pid     /var/run/nginx.pid;

events {
    worker_connections  1024;
}

http{

    include       /etc/nginx/mime.types;
    default_type  application/octet-stream;

    server {
        listen 80;
        listen [::]:80;

        client_max_body_size 20M;

        location /api {
                proxy_set_header        Host    $http_host;
                proxy_http_version      1.1;
                proxy_pass              http://dev-spring-app-blue:8081/api;
                proxy_set_header    X-Real-IP           $remote_addr;
        }

        location /ws {
                proxy_pass http://dev-spring-app-blue:8081/ws;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection "upgrade";
                proxy_set_header Host $host;
        }
    }

    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';
    access_log  /var/log/nginx/access.log  main;

    sendfile        on;
    keepalive_timeout  65;
    include /etc/nginx/conf.d/*.conf;

}
```

- **nginx.green.conf**

```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log warn;
pid     /var/run/nginx.pid;

events {
    worker_connections  1024;
}

http{

    include       /etc/nginx/mime.types;
    default_type  application/octet-stream;

    server {
        listen 80;
        listen [::]:80;

        client_max_body_size 20M;

        location /api {
                proxy_set_header        Host    $http_host;
                proxy_http_version      1.1;
                proxy_pass              http://dev-spring-app-green:8082/api;
                proxy_set_header    X-Real-IP           $remote_addr;

        }

        location /ws {
                proxy_pass http://dev-spring-app-green:8082/ws;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection "upgrade";
                proxy_set_header Host $host;
        }
    }

    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';
    access_log  /var/log/nginx/access.log  main;

    sendfile        on;
    keepalive_timeout  65;
    include /etc/nginx/conf.d/*.conf;

}
```

3. docker compose 실행

```
docker compose -p dev-webserver up -d
```