# Module 6 - Kubernetes Deployments (Blue-Green)

**Goal**: Learn how perform a blue-green deployment using Kubernetes

## Steps

1. Ensure that the local kubernetes cluster is running: kubectl get nodes
2. Create or login to an account on <https://hub.docker.com>
3. Execute docker login, you may be prompted to enter your Docker Hub username and password
4. Build the docker image: docker build -t YOUR\_DOCKER\_HUB\_USERNAME/my-flask-app:latest .
5. Push the image to Docker Hub: docker push YOUR\_DOCKER\_HUB\_USERNAME/my-flask-app:latest
6. Examine the following files. What purpose does each file serve?

* app.py
* Dockerfile
* blue-deployment.yml
* green-deployment.yml
* service.yml

1. Open blue-deployment.yml and green-deployment.yml and fill in your Docker Hub username. Save the files.
2. Deploy the blue version of the application:

kubectl apply -f blue-deployment.yaml  
kubectl apply -f service.yaml

1. Verify the Service routes to blue: kubectl get pods -l app=flask-app,version=blue
2. Navigate to localhost:80 and check that the blue application is running
3. Deploy the green application: kubectl apply -f green-deployment.yaml
4. Wait until the deployment of green is available: kubectl get pods -l app=flask-app,version=green
5. Once green is running, switch traffic with the following command:

kubectl patch service flask-service -p '{"spec":{"selector": {"app": "flask-app", "version": "green"}}}'

1. Navigate to localhost:80 and check that the green application is running
2. To switch back to blue, run:

kubectl patch service flask-service -p '{"spec":{"selector": {"app": "flask-app", "version": "blue"}}}'