Demo Project: Deploy Microservices with Helmfile

This project demonstrates how to Deploy Microservices with Helm and Helmfile.

Deploy Microservices with Helm
Deploy Microservices with Helmfile

Deploy Microservices with Helm

1. **Create** `install.sh`

```
helm install -f values/redis-values.yaml rediscart charts/redis

helm install -f values/email-service-values.yaml emailservice charts/microservice
helm install -f values/cart-service-values.yaml cartservice charts/microservice
helm install -f values/currency-service-values.yaml currencyservice charts/microservice
helm install -f values/payment-service-values.yaml paymentservice charts/microservice
helm install -f values/recommendation-service-values.yaml recommendationservice charts/microservice
helm install -f values/productcatalog-service-values.yaml productcatalogservice charts/microservice
helm install -f values/shipping-service-values.yaml shippingservice charts/microservice
helm install -f values/ad-service-values.yaml adservice charts/microservice
helm install -f values/checkout-service-values.yaml checkoutservice charts/microservice
helm install -f values/frontend-values.yaml frontendservice charts/microservice
```

2. **Make Script Executable**: `chmod u+x` `install.sh`

3. **Execute the script:** `./install.sh`

4. **Verify pods**: `kubectl get pod`

```
kubectl get pod

NAME                                         READY   STATUS    RESTARTS   AGE
adservice-54b74b55b7-dgd6l                   1/1     Running   0          29s
adservice-54b74b55b7-xbgjk                   1/1     Running   0          29s
cartservice-dd4f7764f-bk6qg                  1/1     Running   0          43s
cartservice-dd4f7764f-vgwnf                  1/1     Running   0          43s
checkoutservice-767cc65db4-bjczw             1/1     Running   0          26s
checkoutservice-767cc65db4-qs4wg             1/1     Running   0          26s
currencyservice-7dcb6cb45f-dtm5v             1/1     Running   0          41s
currencyservice-7dcb6cb45f-qb8c7             1/1     Running   0          41s
emailservice-6d488b67f8-bpm45                1/1     Running   0          52m
emailservice-6d488b67f8-d2zsw                1/1     Running   0          52m
frontend-696858b68d-4b99d                    1/1     Running   0          23s
frontend-696858b68d-xgcxw                    1/1     Running   0          23s
paymentservice-795d5d4bd8-kshxb              1/1     Running   0          39s
paymentservice-795d5d4bd8-n8l4n              1/1     Running   0          39s
productcatalogservice-7cf54479b6-ncrpr       1/1     Running   0          34s
productcatalogservice-7cf54479b6-v6m7h       1/1     Running   0          34s
recommendationservice-76887dcc55-7rjxs       1/1     Running   0          36s
recommendationservice-76887dcc55-vspc9       1/1     Running   0          36s
redis-cart-7ff4c98f7-9n528                   1/1     Running   0          47s
redis-cart-7ff4c98f7-svhhb                   1/1     Running   0          47s
shippingservice-d84b79bb-ff22b               1/1     Running   0          32s
shippingservice-d84b79bb-jwt64               1/1     Running   0          32s
```

5. **Clean-Up:**

   - **Create** `uninstall.sh`

   - **Make Script Executable**: `chmod u+x un` `install.sh`

   - **Execute the script:** `./uninstall.sh`

   - **Confirm pods are terminated:** `kubectl get pod`

Deploy Microservices with Helmfile

1. **Install helmfile:** `brew install helmfile`

   - In case you don't have "brew" install with these steps:

     1. **Install Homebrew**:
        Run the following command to install Homebrew:

        ```
        /bin/bash -c "$(curl -fsSL
        https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh )"
        ```

2. **Add Homebrew to your PATH:**
   Follow the on-screen instructions after installation, or add this to your `.zshrc` file:

   ```
   echo 'eval "$(/home/linuxbrew/.linuxbrew/bin/brew shellenv)"' >> ~/.zshrc
   source ~/.zshrc
   ```

   - **Verify Installation**: `helmfile --version`

2. **Create** `helmfile.yaml`:

   - Define all services and Redis in `helmfile.yaml`.

3. **Deploy Using Helmfile**: `helmfile sync`

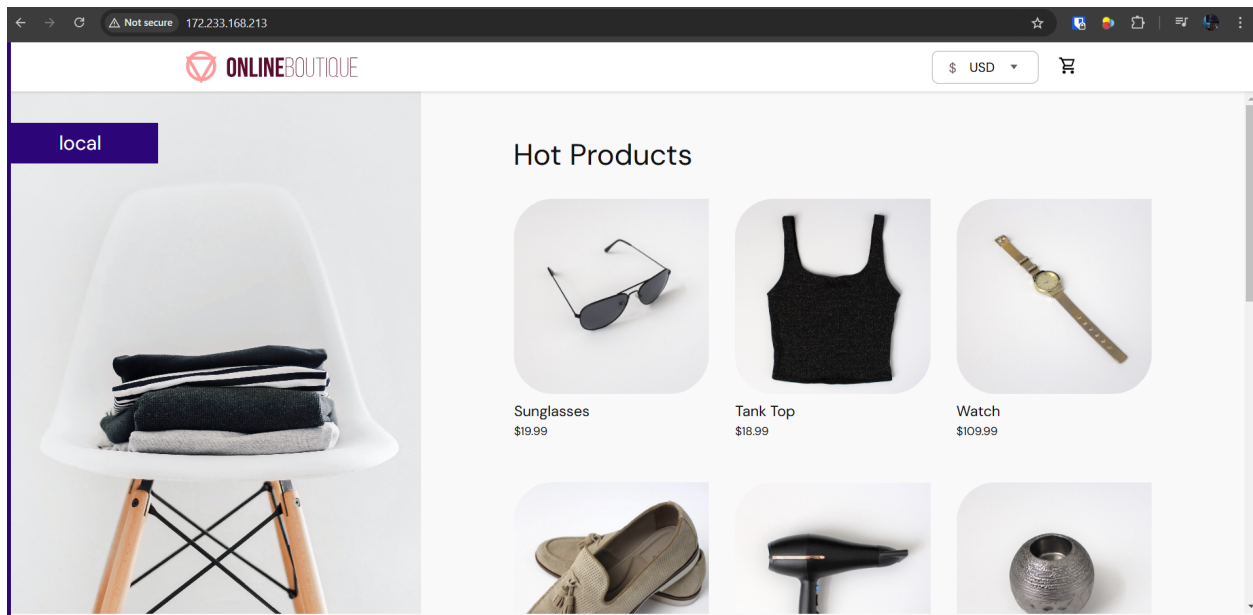4. **Verify running pods:** `kubectl get pod`

```
kubectl get pod
NAME                                    READY   STATUS    RESTARTS   AGE
adservice-54b74b55b7-ft6zg              1/1     Running   0          4m52s
adservice-54b74b55b7-xz8st              1/1     Running   0          4m53s
cartservice-dd4f7764f-wqpd7             1/1     Running   0          4m52s
cartservice-dd4f7764f-zj2l8             1/1     Running   0          4m51s
checkoutservice-767cc65db4-8w8cf        1/1     Running   0          4m54s
checkoutservice-767cc65db4-9bj4k        1/1     Running   0          4m54s
currencyservice-7dcb6cb45f-blmtd        1/1     Running   0          4m51s
currencyservice-7dcb6cb45f-n46qf        1/1     Running   0          4m51s
emailservice-6d488b67f8-5lt7z           1/1     Running   0          4m52s
emailservice-6d488b67f8-rhxrr           1/1     Running   0          4m53s
frontend-696858b68d-47q2w               1/1     Running   0          4m54s
frontend-696858b68d-nftjt               1/1     Running   0          4m54s
paymentservice-795d5d4bd8-7ggbx         1/1     Running   0          4m54s
paymentservice-795d5d4bd8-7m7zf         1/1     Running   0          4m53s
productcatalogservice-7cf54479b6-nlnc4  1/1     Running   0          4m52s
productcatalogservice-7cf54479b6-wxkr4  1/1     Running   0          4m53s
recommendationservice-76887dcc55-j8vvm  1/1     Running   0          4m54s
recommendationservice-76887dcc55-xx7ck  1/1     Running   0          4m54s
redis-cart-6dfd6dfd75-68pk7             1/1     Running   0          4m54s
shippingservice-d84b79bb-5ck2p          1/1     Running   0          4m54s
shippingservice-d84b79bb-kgdsv          1/1     Running   0          4m54s
```

5. **Validate the Application:**

   - Use the Linode NodeBalancer IP to access the application in the browser:

6. **Clean up:** `helmfile destroy`