

Steps	Process	Commands
1	Get the Application code	<i>git clone https://github.com/WASdev/sample.servlet.git</i>
2	Build the code	<i>cd sample.servlet mvn clean install</i>
3	Create a Dockerfile	Create a file called "Dockerfile" with the following content FROM websphere-liberty:webProfile7 COPY sample.servlet/target/sample.servlet.war /config/dropins/servlet.war
4	Build the docker image	<i>sudo docker build -t simpleapp .</i>
5	Test the application by running the docker image Check the application running	<i>sudo docker run -d -p 80:9080 -p 443:9443 simpleapp</i> Access http://localhost/servlet/ in your browser
6	Store your docker image in a repository (public/private) from where you can access it In this example we will use the public docker registry (You need to have a login at https://hub.docker.com/)	<i>docker login</i> Enter your login credentials tag the docker image – <i>docker tag simpleapp [your docker id]/simpleapp</i> <i>docker push [your docker id]/simpleapp</i>

Steps	Process	Commands
7	Create the deployment configuration file needed to deploy the application to ICP	sample Yaml file is shown in the next slide. Copy it and save it in a file called <i>simpleapp.yaml</i>
8	Login and set context	Go to https://icp_ip:8443 Click on “configure client” Copy the Configurations Paste those configurations at command line
9	Deploy the application to ICP	kubectl apply -f simpleapp.yaml
10	Wait for the application to be deployed Check the log to see if the application has started	<i>Kubectl get pods</i> <i>kubectl logs <POD NAME></i>
11	Access the application	

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: simpleapp-service
spec:
  replicas: 1
  template:
    metadata:
      name: simpleapp-service
      labels:
        run: simpleapp-service
    spec:
      containers:
        - name: simpleapp-service
          image: "[your docker id]/simpleapp"
          imagePullPolicy: IfNotPresent
---
apiVersion: v1
kind: Service
metadata:
  name: simpleapp-service
  labels:
    run: simpleapp-service
spec:
  type: NodePort
  selector:
    run: simpleapp-service
  ports:
    - protocol: TCP
      port: 9080
```