

Vote-servcie

* Create a new service: name = vote-service
* port = '5000'
* targetPort = '80'
* nodePort= '31000'
* service endpoint exposes deployment 'vote-deployment'

vote-deployment

* Create a deployment: name = 'vote-deployment'
* image = 'kodekloud/examplevotingapp\_vote:before'
* status: 'Running'

redis-service

* New Service, name = 'redis'
* port: '6379'
* targetPort: '6379'
* type: 'ClusterIP'
* service endpoint exposes deployment 'redis-deployment'

redis-deployment

* Create new deployment, name: 'redis-deployment'
* image: 'redis:alpine'
* Volume Type: 'EmptyDir'
* Volume Name: 'redis-data'
* mountPath: '/data'
* status: 'Running'

worker-deployment

* Create new deployment. name: 'worker'
* image: 'kodekloud/examplevotingapp\_worker'
* status: 'Running'

db

* Create new service: 'db'
* port: '5432'
* targetPort: '5432'
* type: 'ClusterIP'

db-deployment

* Create new deployment. name: 'db-deployment'
* image: 'postgres:9.4' and add the env: 'POSTGRES\_HOST\_AUTH\_METHOD=trust'
* Volume Type: 'EmptyDir'
* Volume Name: 'db-data'
* mountPath: '/var/lib/postgresql/data'
* status: 'Running'

result-deployment

* Create new deployment, name: 'result-deployment'
* image: 'kodekloud/examplevotingapp\_result:before'
* status: 'Running'

result-service

* port: '5001'
* targetPort: '80'
* NodePort: '31001'