

Example: Wordpress + MySQL

- [Example: Deploying WordPress and MySQL with Persistent Volumes | Kubernetes](#)

kustomization.yaml

```
secretGenerator:  
  - name: mysql-pass  
    literals:  
      - password=YOUR_PASSWORD  
  
resources:  
  - mysql-deployment.yaml  
  - wordpress-deployment.yaml
```

mysql-deployment.yaml

```
apiVersion: v1  
kind: Service  
metadata:  
  name: wordpress-mysql  
  labels:  
    app: wordpress  
spec:  
  ports:  
    - port: 3306  
  selector:  
    app: wordpress  
    tier: mysql  
  clusterIP: None  
---  
apiVersion: v1  
kind: PersistentVolumeClaim
```

```
metadata:
  name: mysql-pv-claim
  labels:
    app: wordpress
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 20Gi
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress-mysql
  labels:
    app: wordpress
spec:
  selector:
    matchLabels:
      app: wordpress
      tier: mysql
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: wordpress
        tier: mysql
    spec:
      containers:
        - image: mysql:8.0
          name: mysql
          env:
            - name: MYSQL_ROOT_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: mysql-pass
                  key: password
            - name: MYSQL_DATABASE
```

```
    value: wordpress
  - name: MYSQL_USER
    value: wordpress
  - name: MYSQL_PASSWORD
    valueFrom:
      secretKeyRef:
        name: mysql-pass
        key: password
  ports:
  - containerPort: 3306
    name: mysql
  volumeMounts:
  - name: mysql-persistent-storage
    mountPath: /var/lib/mysql
  volumes:
  - name: mysql-persistent-storage
    persistentVolumeClaim:
      claimName: mysql-pv-claim
```

wordpress-deployment.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: wordpress
  labels:
    app: wordpress
spec:
  ports:
  - port: 80
  selector:
    app: wordpress
    tier: frontend
  type: LoadBalancer
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: wp-pv-claim
  labels:
```

```
  app: wordpress
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 20Gi
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress
  labels:
    app: wordpress
spec:
  selector:
    matchLabels:
      app: wordpress
      tier: frontend
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: wordpress
        tier: frontend
    spec:
      containers:
        - image: wordpress:6.2.1-apache
          name: wordpress
          env:
            - name: WORDPRESS_DB_HOST
              value: wordpress-mysql
            - name: WORDPRESS_DB_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: mysql-pass
                  key: password
            - name: WORDPRESS_DB_USER
              value: wordpress
```

```
ports:
  - containerPort: 80
    name: wordpress
volumeMounts:
  - name: wordpress-persistent-storage
    mountPath: /var/www/html
volumes:
  - name: wordpress-persistent-storage
    persistentVolumeClaim:
      claimName: wp-pv-claim
```

Apply and verify

```
kubectl apply -k ./
kubectl get secrets
kubectl get pvc
```

Revision #5

Created 2026-01-30 10:39:52 CST by A-Lang (Admin)

Updated 2026-01-30 15:21:48 CST by A-Lang (Admin)