

```
apiVersion: v1
kind: Secret
metadata:
  name: gift-secret
  labels:
    app: gift-service
    project: gift-service
    tier: backend
# From source_5
stringData:
  JWT_SECRET: WW4ya2ppYmRkRkFXdG5QSjJBRmxMOFdYbW9oSk1Ddm1nUWdnYUV5cGE1RT0=
  KAKAO_CLIENT_ID: YWMyMGQ4NGY5N2I2NzlmMGJiZDlhODA2NWJhODIzZjM=
  KAKAO_CLIENT_SECRET: RnczSm1MNVBrcGV2WFZaOU9pVk04YmZiZ0hVUEQ1RUE=
---
apiVersion: v1
kind: ConfigMap
metadata:
  name: gift-config
  labels:
    app: gift-service
    project: gift-service
    tier: backend
# From source_5
data:
  API_GATEWAY_URI: "http://api-gateway-svc:8080"
  SERVICE_USER_URI: "http://user-service-svc:8084"
  SERVICE_PRODUCT_URI: "http://product-service-svc:8081"
  SERVICE_WISH_URI: "http://wish-service-svc:8082"
  SERVICE_ORDER_URI: "http://order-service-svc:8083"
  KAKAO_REDIRECT_URI: "http://localhost:8080/members/login/oauth2/code/kakao"
  FRONT_DOMAIN: "http://localhost:8080"
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: api-gateway
  labels:
    app: api-gateway
    project: gift-service
    tier: backend
# From source_6
spec:
  replicas: 1
  selector:
    matchLabels:
      app: api-gateway
  template:
    metadata:
      labels:
        app: api-gateway
        project: gift-service
        tier: backend
```

```
spec:
  containers:
    - name: api-gateway
      image: sk124590/api-gateway:2.1
      ports:
        - containerPort: 8080
      resources:
        requests:
          cpu: "100m"
          memory: "256Mi"
        limits:
          cpu: "500m"
          memory: "512Mi"
      envFrom:
        - configMapRef:
            name: gift-config
        - secretRef:
            name: gift-secret
  ---
apiVersion: v1
kind: Service
metadata:
  name: api-gateway-svc
  labels:
    app: api-gateway
    project: gift-service
    tier: backend
# From source_12
spec:
  type: ClusterIP
  ports:
    - port: 8080
      targetPort: 8080
  selector:
    app: api-gateway
  ---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: user-service
  labels:
    app: user-service
    project: gift-service
    tier: backend
# From source_8
spec:
  replicas: 1
  selector:
    matchLabels:
      app: user-service
  template:
    metadata:
      labels:
        app: user-service
```

```
    project: gift-service
    tier: backend
  spec:
    containers:
      - name: user-service
        image: sk124590/user-service:2.1
        ports:
          - containerPort: 8084
        resources:
          requests:
            cpu: "150m"
            memory: "512Mi"
          limits:
            cpu: "1000m"
            memory: "1Gi"
        envFrom:
          - configMapRef:
              name: gift-config
          - secretRef:
              name: gift-secret
    ---
  apiVersion: v1
  kind: Service
  metadata:
    name: user-service-svc
    labels:
      app: user-service
      project: gift-service
      tier: backend
  # From source_12
  spec:
    type: ClusterIP
    ports:
      - port: 8084
        targetPort: 8084
    selector:
      app: user-service
    ---
  apiVersion: apps/v1
  kind: Deployment
  metadata:
    name: product-service
    labels:
      app: product-service
      project: gift-service
      tier: backend
  # From source_10
  spec:
    replicas: 1
    selector:
      matchLabels:
        app: product-service
    template:
      metadata:
```

```
    labels:
      app: product-service
      project: gift-service
      tier: backend
  spec:
    containers:
      - name: product-service
        image: sk124590/product-service:2.1
        ports:
          - containerPort: 8081
        resources:
          requests:
            cpu: "150m"
            memory: "512Mi"
          limits:
            cpu: "1000m"
            memory: "1Gi"
---
apiVersion: v1
kind: Service
metadata:
  name: product-service-svc
  labels:
    app: product-service
    project: gift-service
    tier: backend
# From source_12
spec:
  type: ClusterIP
  ports:
    - port: 8081
      targetPort: 8081
  selector:
    app: product-service
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wish-service
  labels:
    app: wish-service
    project: gift-service
    tier: backend
# From source_4
spec:
  replicas: 1
  selector:
    matchLabels:
      app: wish-service
  template:
    metadata:
      labels:
        app: wish-service
        project: gift-service
```

```
    tier: backend
  spec:
    containers:
      - name: wish-service
        image: sk124590/wish-service:2.1
        ports:
          - containerPort: 8082
        resources:
          requests:
            cpu: "100m"
            memory: "512Mi"
          limits:
            cpu: "700m"
            memory: "1Gi"
        envFrom:
          - configMapRef:
              name: gift-config
    ---
  apiVersion: v1
  kind: Service
  metadata:
    name: wish-service-svc
    labels:
      app: wish-service
      project: gift-service
      tier: backend
  # From source_12
  spec:
    type: ClusterIP
    ports:
      - port: 8082
        targetPort: 8082
    selector:
      app: wish-service
    ---
  apiVersion: apps/v1
  kind: Deployment
  metadata:
    name: order-service
    labels:
      app: order-service
      project: gift-service
      tier: backend
  # From source_7
  spec:
    replicas: 1
    selector:
      matchLabels:
        app: order-service
    template:
      metadata:
        labels:
          app: order-service
          project: gift-service
```

```
    tier: backend
  spec:
    containers:
      - name: order-service
        image: sk124590/order-service:2.1
        ports:
          - containerPort: 8083
        resources:
          requests:
            cpu: "100m"
            memory: "512Mi"
          limits:
            cpu: "700m"
            memory: "1Gi"
        envFrom:
          - configMapRef:
              name: gift-config
          - secretRef:
              name: gift-secret
    ---
  apiVersion: v1
  kind: Service
  metadata:
    name: order-service-svc
    labels:
      app: order-service
      project: gift-service
      tier: backend
  # From source_12
  spec:
    type: ClusterIP
    ports:
      - port: 8083
        targetPort: 8083
    selector:
      app: order-service
    ---
  apiVersion: apps/v1
  kind: Deployment
  metadata:
    name: gift-app
    labels:
      app: gift-app
      project: gift-service
      tier: frontend
  # From source_10
  spec:
    replicas: 1
    selector:
      matchLabels:
        app: gift-app
    template:
      metadata:
        labels:
```

```

    app: gift-app
    project: gift-service
    tier: frontend
spec:
  containers:
    - name: gift-app
      image: sk124590/gift-app:2.1
      ports:
        - containerPort: 8085
      resources:
        requests:
          cpu: "100m"
          memory: "512Mi"
        limits:
          cpu: "500m"
          memory: "1Gi"
      envFrom:
        - configMapRef:
            name: gift-config
        - secretRef:
            name: gift-secret
  ---
apiVersion: v1
kind: Service
metadata:
  name: gift-app-svc
  labels:
    app: gift-app
    project: gift-service
    tier: frontend
# From source_12
spec:
  type: LoadBalancer
  ports:
    - port: 8080
      targetPort: 8085
  selector:
    app: gift-app
```

YAML 위치와 내용	참조된 Source	비고
gift-secret.Secret.metadata.name = gift-secret	source_5	Secret 리소스 이름
gift-config.ConfigMap.metadata.name = gift-config	source_5	ConfigMap 리소스 이름

YAML 위치와 내용	참조된 Source	비고
gift-config.ConfigMap.data = {API_GATEWAY_URI, SERVICE_USER_URI, SERVICE_PRODUCT_URI, SERVICE_WISH_URI, SERVICE_ORDER_URI, KAKAO_REDIRECT_URI, FRONT_DOMAIN}	source_5	서비스 엔드포인트/도메인 설정 값
api-gateway.Deployment.spec.replicas = 1	source_6	레플리카 수
api-gateway.Deployment.spec.template.spec.containers[api-gateway].image = sk124590/api-gateway:2.1	source_6	컨테이너 이미지 태그
api-gateway.Deployment.spec.template.spec.containers[api-gateway].ports[0].containerPort = 8080	source_6	컨테이너 포트
api-gateway.Deployment.spec.template.spec.containers[api-gateway].resources.requests = {cpu: "100m", memory: "256Mi"}	source_6	리소스 요청
api-gateway.Deployment.spec.template.spec.containers[api-gateway].resources.limits = {cpu: "500m", memory: "512Mi"}	source_6	리소스 제한
api-gateway-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
api-gateway-svc.Service.spec.ports[0].port = 8080	source_12	서비스 포트
user-service.Deployment.spec.replicas = 1	source_8	레플리카 수
user-service.Deployment.spec.template.spec.containers[user-service].image = sk124590/user-service:2.1	source_8	컨테이너 이미지 태그
user-service.Deployment.spec.template.spec.containers[user-service].ports[0].containerPort = 8084	source_8	컨테이너 포트
user-service.Deployment.spec.template.spec.containers[user-service].resources.requests = {cpu: "150m", memory: "512Mi"}	source_8	리소스 요청
user-service.Deployment.spec.template.spec.containers[user-service].resources.limits = {cpu: "1000m", memory: "1Gi"}	source_8	리소스 제한
user-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
user-service-svc.Service.spec.ports[0].port = 8084	source_12	서비스 포트
product-service.Deployment.spec.replicas = 1	source_10	레플리카 수

YAML 위치와 내용	참조된 Source	비고
product-service.Deployment.spec.template.spec.containers[product-service].image = sk124590/product-service:2.1	source_10	컨테이너 이미지 태그
product-service.Deployment.spec.template.spec.containers[product-service].ports[0].containerPort = 8081	source_10	컨테이너 포트
product-service.Deployment.spec.template.spec.containers[product-service].resources.requests = {cpu: "150m", memory: "512Mi"}	source_10	리소스 요청
product-service.Deployment.spec.template.spec.containers[product-service].resources.limits = {cpu: "1000m", memory: "1Gi"}	source_10	리소스 제한
product-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
product-service-svc.Service.spec.ports[0].port = 8081	source_12	서비스 포트
wish-service.Deployment.spec.replicas = 1	source_4	레플리카 수
wish-service.Deployment.spec.template.spec.containers[wish-service].image = sk124590/wish-service:2.1	source_4	컨테이너 이미지 태그
wish-service.Deployment.spec.template.spec.containers[wish-service].ports[0].containerPort = 8082	source_4	컨테이너 포트
wish-service.Deployment.spec.template.spec.containers[wish-service].resources.requests = {cpu: "100m", memory: "512Mi"}	source_4	리소스 요청
wish-service.Deployment.spec.template.spec.containers[wish-service].resources.limits = {cpu: "700m", memory: "1Gi"}	source_4	리소스 제한
wish-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
wish-service-svc.Service.spec.ports[0].port = 8082	source_12	서비스 포트
order-service.Deployment.spec.replicas = 1	source_7	레플리카 수
order-service.Deployment.spec.template.spec.containers[order-service].image = sk124590/order-service:2.1	source_7	컨테이너 이미지 태그
order-service.Deployment.spec.template.spec.containers[order-service].ports[0].containerPort = 8083	source_7	컨테이너 포트

YAML 위치와 내용	참조된 Source	비고
order-service.Deployment.spec.template.spec.containers[order-service].resources.requests = {cpu: "100m", memory: "512Mi"}	source_7	리소스 요청
order-service.Deployment.spec.template.spec.containers[order-service].resources.limits = {cpu: "700m", memory: "1Gi"}	source_7	리소스 제한
order-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
order-service-svc.Service.spec.ports[0].port = 8083	source_12	서비스 포트
gift-app.Deployment.spec.replicas = 1	source_10	레플리카 수
gift-app.Deployment.spec.template.spec.containers[gift-app].image = sk124590/gift-app:2.1	source_10	컨테이너 이미지 태그
gift-app.Deployment.spec.template.spec.containers[gift-app].ports[0].containerPort = 8085	source_10	컨테이너 포트
gift-app.Deployment.spec.template.spec.containers[gift-app].resources.requests = {cpu: "100m", memory: "512Mi"}	source_10	리소스 요청
gift-app.Deployment.spec.template.spec.containers[gift-app].resources.limits = {cpu: "500m", memory: "1Gi"}	source_10	리소스 제한
gift-app-svc.Service.spec.type = LoadBalancer	source_12	서비스 타입
gift-app-svc.Service.spec.ports[0].port = 8080	source_12	서비스 포트