```yaml
apiVersion: v1
kind: Secret
metadata:
 name: gift-secret
 labels:
   app: gift-service
   project: gift-service
   tier: backend
# From source_5, source_11

data:
 JWT_SECRET: WW4ya2ppYmRkRkFXdG5QSjJBRmxMOFdYbW9oSk1DdmlnUWdnYYV5cGE1RT0=
 KAKAO_CLIENT_ID: YWMyMGQ4NGY5N2I2NzlmMGJiZDlhODA2NWJhODIzZjM=
 KAKAO_CLIENT_SECRET: RnczSmlMNVBrcGV2WFZaOU9pVk04YmZiZ0hVUEQ1RUE=
# From source_5, source_13
---
apiVersion: v1
kind: ConfigMap
metadata:
 name: gift-config
 labels:
   app: gift-service
   project: gift-service
   tier: backend
# From source_5, source_11

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
# From source_5
---
apiVersion: apps/v1
kind: Deployment
metadata:
 name: api-gateway
 labels:
   app: api-gateway
   project: gift-service
   tier: backend
# From source_6, source_11

spec:
 replicas: 1
 selector:
   matchLabels:
     app: api-gateway
 template:
```

```yaml
    metadata:
      labels:
        app: api-gateway
        project: gift-service
        tier: backend
    spec:
      containers:
      - name: api-gateway
        image: sk124590/api-gateway:2.1
        ports:
        - containerPort: 8080
        env:
        - name: SERVICE_USER_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_USER_URI
        - name: SERVICE_PRODUCT_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_PRODUCT_URI
        - name: SERVICE_WISH_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_WISH_URI
        - name: SERVICE_ORDER_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_ORDER_URI
        - name: JWT_SECRET
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: JWT_SECRET
        resources:
          requests:
            cpu: "100m"
            memory: "256Mi"
          limits:
            cpu: "500m"
            memory: "512Mi"
# From source_6
---
apiVersion: v1
kind: Service
metadata:
 name: api-gateway-svc
 labels:
   app: api-gateway
   project: gift-service
   tier: backend
```

```yaml
# From source_6, source_11

spec:
  type: ClusterIP
  ports:
  - port: 8080
    targetPort: 8080
  selector:
    app: api-gateway
# From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: user-service
  labels:
    app: user-service
    project: gift-service
    tier: backend
# From source_8, source_11

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
        env:
        - name: JWT_SECRET
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: JWT_SECRET
        - name: KAKAO_CLIENT_ID
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: KAKAO_CLIENT_ID
        - name: KAKAO_CLIENT_SECRET
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: KAKAO_CLIENT_SECRET
```

```yaml
          - name: KAKAO_REDIRECT_URI
            valueFrom:
              configMapKeyRef:
                name: gift-config
                key: KAKAO_REDIRECT_URI
          resources:
            requests:
              cpu: "150m"
              memory: "512Mi"
            limits:
              cpu: "1000m"
              memory: "1Gi"
# From source_8
---
apiVersion: v1
kind: Service
metadata:
 name: user-service-svc
 labels:
   app: user-service
   project: gift-service
   tier: backend
# From source_8, source_11

spec:
 type: ClusterIP
 ports:
 - port: 8084
   targetPort: 8084
 selector:
   app: user-service
# From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
 name: product-service
 labels:
   app: product-service
   project: gift-service
   tier: backend
# From source_10, source_11

spec:
 replicas: 1
 selector:
   matchLabels:
     app: product-service
 template:
   metadata:
     labels:
       app: product-service
       project: gift-service
       tier: backend
```

```yaml
    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"
# From source_10
---
apiVersion: v1
kind: Service
metadata:
 name: product-service-svc
 labels:
    app: product-service
    project: gift-service
    tier: backend
# From source_10, source_11

spec:
 type: ClusterIP
 ports:
 - port: 8081
   targetPort: 8081
 selector:
    app: product-service
# From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
 name: wish-service
 labels:
    app: wish-service
    project: gift-service
    tier: backend
# From source_4, source_11

spec:
 replicas: 1
 selector:
   matchLabels:
     app: wish-service
  template:
    metadata:
      labels:
        app: wish-service
        project: gift-service
```

```yaml
      tier: backend
    spec:
      containers:
      - name: wish-service
        image: sk124590/wish-service:2.1
        ports:
        - containerPort: 8082
        env:
        - name: SERVICE_PRODUCT_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_PRODUCT_URI
        resources:
          requests:
            cpu: "100m"
            memory: "512Mi"
          limits:
            cpu: "700m"
            memory: "1Gi"
# From source_4
---
apiVersion: v1
kind: Service
metadata:
 name: wish-service-svc
 labels:
    app: wish-service
    project: gift-service
    tier: backend
# From source_4, source_11

spec:
 type: ClusterIP
 ports:
 - port: 8082
    targetPort: 8082
 selector:
    app: wish-service
# From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
 name: order-service
 labels:
    app: order-service
    project: gift-service
    tier: backend
# From source_7, source_11

spec:
 replicas: 1
 selector:
```

```yaml
      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
        env:
        - name: SERVICE_PRODUCT_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_PRODUCT_URI
        - name: SERVICE_WISH_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_WISH_URI
        - name: SERVICE_USER_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: SERVICE_USER_URI
        - name: FRONT_DOMAIN
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: FRONT_DOMAIN
        - name: KAKAO_CLIENT_ID
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: KAKAO_CLIENT_ID
        - name: KAKAO_CLIENT_SECRET
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: KAKAO_CLIENT_SECRET
        resources:
          requests:
            cpu: "100m"
            memory: "512Mi"
          limits:
            cpu: "700m"
            memory: "1Gi"
# From source_7
---
```

```yaml
apiVersion: v1
kind: Service
metadata:
 name: order-service-svc
 labels:
   app: order-service
   project: gift-service
   tier: backend
# From source_7, source_11

spec:
 type: ClusterIP
 ports:
 - port: 8083
   targetPort: 8083
 selector:
   app: order-service
# From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
 name: gift-app
 labels:
   app: gift-app
   project: gift-service
   tier: frontend
# From source_10, source_11

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
       env:
       - name: API_GATEWAY_URI
         valueFrom:
           configMapKeyRef:
             name: gift-config
             key: API_GATEWAY_URI
       - name: KAKAO_REDIRECT_URI
         valueFrom:
```

```yaml
              configMapKeyRef:
                name: gift-config
                key: KAKAO_REDIRECT_URI
          - name: JWT_SECRET
            valueFrom:
              secretKeyRef:
                name: gift-secret
                key: JWT_SECRET
          - name: KAKAO_CLIENT_ID
            valueFrom:
              secretKeyRef:
                name: gift-secret
                key: KAKAO_CLIENT_ID
          - name: KAKAO_CLIENT_SECRET
            valueFrom:
              secretKeyRef:
                name: gift-secret
                key: KAKAO_CLIENT_SECRET
        resources:
          requests:
            cpu: "100m"
            memory: "512Mi"
          limits:
            cpu: "500m"
            memory: "1Gi"
# From source_10
---
apiVersion: v1
kind: Service
metadata:
 name: gift-app-svc
 labels:
    app: gift-app
    project: gift-service
    tier: frontend
# From source_10, source_11

spec:
 type: LoadBalancer
 ports:
 - port: 8080
   targetPort: 8085
 selector:
    app: gift-app
# From source_12
```

YAML 위치와 내용	참조된 Source	비고
gift-secret.Secret.metadata.labels (block) = {app: gift-service, project: gift-service, tier: backend}	source_5, source_11	Secret 메타데이터 라벨 전체

YAML 위치와 내용	참조된 Source	비고
gift-secret.Secret.data (block) = `JWT_SECRET, KAKAO_CLIENT_ID, KAKAO_CLIENT_SECRET`	source_5, source_13	민감정보를 Secret에 저장(정책 포함)
gift-config.ConfigMap.metadata.labels (block) = `{app: gift-service, project: gift-service, tier: backend}`	source_5, source_11	ConfigMap 메타데이터 라벨 전체
gift-config.ConfigMap.data (block) = `API_GATEWAY_URI, SERVICE_USER_URI, SERVICE_PRODUCT_URI, SERVICE_WISH_URI, SERVICE_ORDER_URI, KAKAO_REDIRECT_URI, FRONT_DOMAIN`	source_5	공통 서비스 엔드포인트 및 프론트/리다이렉트 설정
api-gateway.Deployment.metadata.labels (block) = `{app: api-gateway, project: gift-service, tier: backend}`	source_6, source_11	배포 리소스/라벨 근거
api-gateway.Deployment.spec.template.spec.containers[0].image = `sk124590/api-gateway:2.1`	source_6	이미지/버전
api-gateway.Deployment.spec.template.spec.containers[0].ports[0].containerPort = `8080`	source_6	컨테이너 포트
api-gateway.Deployment.spec.template.spec.containers[0].env (block) = `SERVICE_USER_URI, SERVICE_PRODUCT_URI, SERVICE_WISH_URI, SERVICE_ORDER_URI, JWT_SECRET`	source_6	ConfigMap/Secret 주입 항목
api-gateway.Deployment.spec.template.spec.containers[0].resources (block) = `requests{100m,256Mi}, limits{500m,512Mi}`	source_6	리소스 요청/제한
api-gateway-svc.Service.metadata.labels (block) = `{app: api-gateway, project: gift-service, tier: backend}`	source_6, source_11	Service 메타데이터 라벨
api-gateway-svc.Service.spec.selector = `app=api-gateway` 등 Service 사양	source_12	서비스 선택자/포트 기준
user-service.Deployment.metadata.labels (block) = `{app: user-service, project: gift-service, tier: backend}`	source_8, source_11	배포 리소스/라벨 근거
user-service.Deployment.spec.replicas = `1`	source_8	복제 수
user-service.Deployment.spec.template.spec.containers[0].image = `sk124590/user-service:2.1`	source_8	이미지/버전
user-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = `8084`	source_8	컨테이너 포트
user-service.Deployment.spec.template.spec.containers[0].env (block) = `JWT_SECRET, KAKAO_CLIENT_ID, KAKAO_CLIENT_SECRET, KAKAO_REDIRECT_URI`	source_8	Secret/ConfigMap 주입 항목

YAML 위치와 내용	참조된 Source	비고
user-service.Deployment.spec.template.spec.containers[0].resources (block) = `requests{150m,512Mi}, limits{1000m,1Gi}`	source_8	리소스 요청/제한
user-service-svc.Service.metadata.labels (block) = `{app: user-service, project: gift-service, tier: backend}`	source_8, source_11	Service 메타데이터 라벨
user-service-svc.Service.spec.selector = `app=user-service` 등 Service 사양	source_12	서비스 선택자/포트 기준
product-service.Deployment.metadata.labels (block) = `{app: product-service, project: gift-service, tier: backend}`	source_10, source_11	배포 리소스/라벨 근거
product-service.Deployment.spec.replicas = `1`	source_10, source_11	복제 수
product-service.Deployment.spec.template.spec.containers[0].image = `sk124590/product-service:2.1`	source_10	이미지/버전
product-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = `8081`	source_10	컨테이너 포트
product-service.Deployment.spec.template.spec.containers[0].resources (block) = `requests{150m,512Mi}, limits{1000m,1Gi}`	source_10	리소스 요청/제한
product-service-svc.Service.metadata.labels (block) = `{app: product-service, project: gift-service, tier: backend}`	source_10, source_11	Service 메타데이터 라벨
product-service-svc.Service.spec.selector = `app=product-service` 등 Service 사양	source_12	서비스 선택자/포트 기준
wish-service.Deployment.metadata.labels (block) = `{app: wish-service, project: gift-service, tier: backend}`	source_4, source_11	배포 리소스/라벨 근거
wish-service.Deployment.spec.replicas = `1`	source_4	복제 수
wish-service.Deployment.spec.template.spec.containers[0].image = `sk124590/wish-service:2.1`	source_4	이미지/버전
wish-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = `8082`	source_4	컨테이너 포트
wish-service.Deployment.spec.template.spec.containers[0].env (block) = `SERVICE_PRODUCT_URI`	source_4	ConfigMap 주입 항목
wish-service.Deployment.spec.template.spec.containers[0].resources (block) = `requests{100m,512Mi}, limits{700m,1Gi}`	source_4	리소스 요청/제한
wish-service-svc.Service.metadata.labels (block) = `{app: wish-service, project: gift-service, tier: backend}`	source_4, source_11	Service 메타데이터 라벨

YAML 위치와 내용	참조된 Source	비고
wish-service-svc.Service.spec.selector = `app=wish-service` 등 Service 사양	source_12	서비스 선택자/포트 기준
order-service.Deployment.metadata.labels (block) = `{app: order-service, project: gift-service, tier: backend}`	source_7, source_11	배포 리소스/라벨 근거
order-service.Deployment.spec.replicas = `1`	source_7	복제 수
order-service.Deployment.spec.template.spec.containers[0].image = `sk124590/order-service:2.1`	source_7	이미지/버전
order-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = `8083`	source_7	컨테이너 포트
order-service.Deployment.spec.template.spec.containers[0].env (block) = `SERVICE_PRODUCT_URI, SERVICE_WISH_URI, SERVICE_USER_URI, FRONT_DOMAIN, KAKAO_CLIENT_ID, KAKAO_CLIENT_SECRET`	source_7	ConfigMap/Secret 주입 항목
order-service.Deployment.spec.template.spec.containers[0].resources (block) = `requests{100m,512Mi}, limits{700m,1Gi}`	source_7	리소스 요청/제한
order-service-svc.Service.metadata.labels (block) = `{app: order-service, project: gift-service, tier: backend}`	source_7, source_11	Service 메타데이터 라벨
order-service-svc.Service.spec.selector = `app=order-service` 등 Service 사양	source_12	서비스 선택자/포트 기준
gift-app.Deployment.metadata.labels (block) = `{app: gift-app, project: gift-service, tier: frontend}`	source_10, source_11	배포 리소스/라벨 근거
gift-app.Deployment.spec.replicas = `1`	source_10	복제 수
gift-app.Deployment.spec.template.spec.containers[0].image = `sk124590/gift-app:2.1`	source_10	이미지/버전
gift-app.Deployment.spec.template.spec.containers[0].ports[0].containerPort = `8085`	source_10	컨테이너 포트
gift-app.Deployment.spec.template.spec.containers[0].env (block) = `API_GATEWAY_URI, KAKAO_REDIRECT_URI, JWT_SECRET, KAKAO_CLIENT_ID, KAKAO_CLIENT_SECRET`	source_10	ConfigMap/Secret 주입 항목
gift-app.Deployment.spec.template.spec.containers[0].resources (block) = `requests{100m,512Mi}, limits{500m,1Gi}`	source_10	리소스 요청/제한
gift-app-svc.Service.metadata.labels (block) = `{app: gift-app, project: gift-service, tier: frontend}`	source_10, source_11	Service 메타데이터 라벨
gift-app-svc.Service.spec (block) = `type: LoadBalancer, ports: 8080→8085, selector: app=gift-app`	source_12	L4 노출/포워딩 및 선택자 기준