

```

apiVersion: v1
kind: Secret
metadata:
  name: gift-secret # From source_5
  labels:
    app: gift-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
data:
  JWT_SECRET: ww4ya2ppYmRkRkFXdG5QSjJBmRmxMOFdYbW9oSk1DdmInUWdnYUV5cGE1RT0= # From source_5
  KAKAO_CLIENT_ID: YwMyMGQ4NGY5N2I2NzlmMGJiZDlhODA2NWJhODIzZjM= # From source_5
  KAKAO_CLIENT_SECRET: RnczSmIMNVBrGV2WFZaOU9pVvK04YmZiZ0hVUEQ1RUE= # From source_5
---
apiVersion: v1
kind: ConfigMap
metadata:
  name: gift-config # From source_5
  labels:
    app: gift-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
data:
  API_GATEWAY_URI: http://api-gateway-svc:8080 # From source_5
  SERVICE_USER_URI: http://user-service-svc:8084 # From source_5
  SERVICE_PRODUCT_URI: http://product-service-svc:8081 # From source_5
  SERVICE_WISH_URI: http://wish-service-svc:8082 # From source_5
  SERVICE_ORDER_URI: http://order-service-svc:8083 # From source_5
  KAKAO_REDIRECT_URI: http://localhost:8080/members/login/oauth2/code/kakao # From source_5
  FRONT_DOMAIN: http://localhost:8080 # From source_5
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: api-gateway # From source_1
  labels:
    app: api-gateway # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  replicas: 1 # From source_6
  selector:
    matchLabels:
      app: api-gateway # From source_11
  template:
    metadata:
      labels:
        app: api-gateway # From source_11
        project: gift-service # From source_11
        tier: backend # From source_11
    spec:
      containers:
        - name: api-gateway
          image: sk124590/api-gateway:2.1 # From source_6
          ports:
            - containerPort: 8080 # From source_6
          env:
            - name: SERVICE_USER_URI
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: SERVICE_USER_URI # From source_6
            - name: SERVICE_PRODUCT_URI
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: SERVICE_PRODUCT_URI # From source_6
            - name: SERVICE_WISH_URI
              valueFrom:
                configMapKeyRef:

```

```

        name: gift-config
        key: SERVICE_WISH_URI # From source_6
- name: SERVICE_ORDER_URI
  valueFrom:
    configMapKeyRef:
      name: gift-config
      key: SERVICE_ORDER_URI # From source_6
- name: JWT_SECRET
  valueFrom:
    secretKeyRef:
      name: gift-secret
      key: JWT_SECRET # From source_6
resources:
  requests:
    cpu: "100m" # From source_6
    memory: "256Mi" # From source_6
  limits:
    cpu: "500m" # From source_6
    memory: "512Mi" # From source_6
---
apiVersion: v1
kind: Service
metadata:
  name: api-gateway-svc # From source_5
  labels:
    app: api-gateway # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  type: ClusterIP # From source_12
  selector:
    app: api-gateway # From source_11
  ports:
- port: 8080 # From source_12
  targetPort: 8080 # From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: user-service # From source_1
  labels:
    app: user-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  replicas: 1 # From source_8
  selector:
    matchLabels:
      app: user-service # From source_11
  template:
    metadata:
      labels:
        app: user-service # From source_11
        project: gift-service # From source_11
        tier: backend # From source_11
    spec:
      containers:
- name: user-service
  image: sk124590/user-service:2.1 # From source_8
  ports:
- containerPort: 8084 # From source_8
  env:
- name: JWT_SECRET
  valueFrom:
    secretKeyRef:
      name: gift-secret
      key: JWT_SECRET # From source_8
- name: KAKAO_CLIENT_ID
  valueFrom:
    secretKeyRef:
      name: gift-secret

```

```

        key: KAKAO_CLIENT_ID # From source_8
- name: KAKAO_CLIENT_SECRET
  valueFrom:
    secretKeyRef:
      name: gift-secret
      key: KAKAO_CLIENT_SECRET # From source_8
- name: KAKAO_REDIRECT_URI
  valueFrom:
    configMapKeyRef:
      name: gift-config
      key: KAKAO_REDIRECT_URI # From source_8
resources:
  requests:
    cpu: "150m" # From source_8
    memory: "512Mi" # From source_8
  limits:
    cpu: "1000m" # From source_8
    memory: "1Gi" # From source_8
---
apiVersion: v1
kind: Service
metadata:
  name: user-service-svc # From source_12
  labels:
    app: user-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  type: ClusterIP # From source_12
  selector:
    app: user-service # From source_11
  ports:
- port: 8084 # From source_12
  targetPort: 8084 # From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: product-service # From source_1
  labels:
    app: product-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  replicas: 1 # From source_10
  selector:
    matchLabels:
      app: product-service # From source_11
  template:
    metadata:
      labels:
        app: product-service # From source_11
        project: gift-service # From source_11
        tier: backend # From source_11
    spec:
      containers:
- name: product-service
  image: sk124590/product-service:2.1 # From source_10
  ports:
- containerPort: 8081 # From source_10
  resources:
    requests:
      cpu: "150m" # From source_10
      memory: "512Mi" # From source_10
    limits:
      cpu: "1000m" # From source_10
      memory: "1Gi" # From source_10
---
apiVersion: v1
kind: Service
metadata:

```

```

name: product-service-svc # From source_12
labels:
  app: product-service # From source_11
  project: gift-service # From source_11
  tier: backend # From source_11
spec:
  type: ClusterIP # From source_12
  selector:
    app: product-service # From source_11
  ports:
    - port: 8081 # From source_12
      targetPort: 8081 # From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wish-service # From source_1
  labels:
    app: wish-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  replicas: 1 # From source_4
  selector:
    matchLabels:
      app: wish-service # From source_11
  template:
    metadata:
      labels:
        app: wish-service # From source_11
        project: gift-service # From source_11
        tier: backend # From source_11
    spec:
      containers:
        - name: wish-service
          image: sk124590/wish-service:2.1 # From source_4
          ports:
            - containerPort: 8082 # From source_4
          env:
            - name: SERVICE_PRODUCT_URI
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: SERVICE_PRODUCT_URI # From source_4
      resources:
        requests:
          cpu: "100m" # From source_4
          memory: "512Mi" # From source_4
        limits:
          cpu: "700m" # From source_4
          memory: "1Gi" # From source_4
---
apiVersion: v1
kind: Service
metadata:
  name: wish-service-svc # From source_12
  labels:
    app: wish-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  type: ClusterIP # From source_12
  selector:
    app: wish-service # From source_11
  ports:
    - port: 8082 # From source_12
      targetPort: 8082 # From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:

```

```

name: order-service # From source_1
labels:
  app: order-service # From source_11
  project: gift-service # From source_11
  tier: backend # From source_11
spec:
  replicas: 1 # From source_7
  selector:
    matchLabels:
      app: order-service # From source_11
  template:
    metadata:
      labels:
        app: order-service # From source_11
        project: gift-service # From source_11
        tier: backend # From source_11
    spec:
      containers:
        - name: order-service
          image: sk124590/order-service:2.1 # From source_7
          ports:
            - containerPort: 8083 # From source_7
          env:
            - name: SERVICE_PRODUCT_URI
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: SERVICE_PRODUCT_URI # From source_7
            - name: SERVICE_WISH_URI
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: SERVICE_WISH_URI # From source_7
            - name: SERVICE_USER_URI
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: SERVICE_USER_URI # From source_7
            - name: FRONT_DOMAIN
              valueFrom:
                configMapKeyRef:
                  name: gift-config
                  key: FRONT_DOMAIN # From source_7
            - name: KAKAO_CLIENT_ID
              valueFrom:
                secretKeyRef:
                  name: gift-secret
                  key: KAKAO_CLIENT_ID # From source_7
            - name: KAKAO_CLIENT_SECRET
              valueFrom:
                secretKeyRef:
                  name: gift-secret
                  key: KAKAO_CLIENT_SECRET # From source_7
      resources:
        requests:
          cpu: "100m" # From source_7
          memory: "512Mi" # From source_7
        limits:
          cpu: "700m" # From source_7
          memory: "1Gi" # From source_7
---
apiVersion: v1
kind: Service
metadata:
  name: order-service-svc # From source_12
  labels:
    app: order-service # From source_11
    project: gift-service # From source_11
    tier: backend # From source_11
spec:
  type: ClusterIP # From source_12

```

```

selector:
  app: order-service # From source_11
ports:
- port: 8083 # From source_12
  targetPort: 8083 # From source_12
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: gift-app # From source_1
  labels:
    app: gift-app # From source_11
    project: gift-service # From source_11
    tier: frontend # From source_11
spec:
  replicas: 1 # From source_10
  selector:
    matchLabels:
      app: gift-app # From source_11
  template:
    metadata:
      labels:
        app: gift-app # From source_11
        project: gift-service # From source_11
        tier: frontend # From source_11
    spec:
      containers:
      - name: gift-app
        image: sk124590/gift-app:2.1 # From source_10
        ports:
        - containerPort: 8085 # From source_10
        env:
        - name: API_GATEWAY_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: API_GATEWAY_URI # From source_10
        - name: KAKAO_REDIRECT_URI
          valueFrom:
            configMapKeyRef:
              name: gift-config
              key: KAKAO_REDIRECT_URI # From source_10
        - name: JWT_SECRET
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: JWT_SECRET # From source_10
        - name: KAKAO_CLIENT_ID
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: KAKAO_CLIENT_ID # From source_10
        - name: KAKAO_CLIENT_SECRET
          valueFrom:
            secretKeyRef:
              name: gift-secret
              key: KAKAO_CLIENT_SECRET # From source_10
      resources:
        requests:
          cpu: "100m" # From source_10
          memory: "512Mi" # From source_10
        limits:
          cpu: "500m" # From source_10
          memory: "1Gi" # From source_10
---
apiVersion: v1
kind: Service
metadata:
  name: gift-app-svc # From source_12
  labels:
    app: gift-app # From source_11

```

```

project: gift-service # From source_11
tier: frontend # From source_11
spec:
  type: LoadBalancer # From source_12
  selector:
    app: gift-app # From source_11
  ports:
    - port: 8080 # From source_12
      targetPort: 8085 # From source_12

```

YAML 위치와 내용	참조된 Source	비고
gift-secret.Secret.metadata.name = gift-secret	source_5	Secret 이름
gift-secret.Secret.metadata.labels.project = gift-service	source_11	공통 라벨(project)
gift-secret.Secret.metadata.labels.tier = backend	source_11	공통 라벨(tier)
gift-secret.Secret.data.JWT_SECRET = WW4ya2ppYmRkRkFXdG5QSjJBRmxMOFdYbW9oSk1DdmlnUWdnYUV5cGE1RT0=	source_5	JWT 시크릿(Base64)
gift-secret.Secret.data.KAKAO_CLIENT_ID = YWMYMGQ4NGY5N2I2NzlmMGJiZDIhODA2NWJhODIzZjM=	source_5	카카오 Client ID(Base64)
gift-secret.Secret.data.KAKAO_CLIENT_SECRET = RnczSmlMNVBrcGV2WFZaOU9pVk04YmZiZ0hVUEQ1RUE=	source_5	카카오 Client Secret(Base64)
gift-config.ConfigMap.metadata.name = gift-config	source_5	ConfigMap 이름
gift-config.ConfigMap.metadata.labels.project = gift-service	source_11	공통 라벨(project)
gift-config.ConfigMap.metadata.labels.tier = backend	source_11	공통 라벨(tier)
gift-config.ConfigMap.data.API_GATEWAY_URI = http://api-gateway-svc:8080	source_5	API 게이트웨이 내부 URI
gift-config.ConfigMap.data.SERVICE_USER_URI = http://user-service-svc:8084	source_5	사용자 서비스 URI
gift-config.ConfigMap.data.SERVICE_PRODUCT_URI = http://product-service-svc:8081	source_5	상품 서비스 URI
gift-config.ConfigMap.data.SERVICE_WISH_URI = http://wish-service-svc:8082	source_5	위시 서비스 URI
gift-config.ConfigMap.data.SERVICE_ORDER_URI = http://order-service-svc:8083	source_5	주문 서비스 URI
gift-config.ConfigMap.data.KAKAO_REDIRECT_URI = http://localhost:8080/members/login/oauth2/code/kakao	source_5	카카오 Redirect URI
gift-config.ConfigMap.data.FRONT_DOMAIN = http://localhost:8080	source_5	프런트 도메인
api-gateway.Deployment.metadata.name = api-gateway	source_6	Deployment 이름
api-gateway.Deployment.metadata.labels.app = api-gateway	source_11	공통 라벨(app)
api-gateway.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨(project)
api-gateway.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
api-gateway.Deployment.spec.replicas = 1	source_6	파드 복제 수
api-gateway.Deployment.spec.template.spec.containers[0].image = sk124590/api-gateway:2.1	source_6	컨테이너 이미지
api-gateway.Deployment.spec.template.spec.containers[0].ports[0].containerPort = 8080	source_6	컨테이너 포트
api-gateway-svc.Service.metadata.name = api-gateway-svc	source_3	Service 이름
api-gateway-svc.Service.metadata.labels.app = api-gateway	source_11	공통 라벨(app)
api-gateway-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨(project)
api-gateway-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
api-gateway-svc.Service.spec.ports[0].port = 8080	source_3	서비스 포트
user-service.Deployment.metadata.name = user-service	source_8	Deployment 이름

YAML 위치와 내용	참조된 Source	비고
user-service.Deployment.metadata.labels.app = user-service	source_11	공통 라벨(app)
user-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨(project)
user-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
user-service.Deployment.spec.replicas = 1	source_8	파드 복제 수
user-service.Deployment.spec.template.spec.containers[0].image = sk124590/user-service:2.1	source_8	컨테이너 이미지
user-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = 8084	source_8	컨테이너 포트
user-service-svc.Service.metadata.name = user-service-svc	source_12	Service 이름
user-service-svc.Service.metadata.labels.app = user-service	source_11	공통 라벨(app)
user-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨(project)
user-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
user-service-svc.Service.spec.ports[0].port = 8084	source_12	서비스 포트
product-service.Deployment.metadata.name = product-service	source_10	Deployment 이름
product-service.Deployment.metadata.labels.app = product-service	source_11	공통 라벨(app)
product-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨(project)
product-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
product-service.Deployment.spec.replicas = 1	source_11	파드 복제 수
product-service.Deployment.spec.template.spec.containers[0].image = sk124590/product-service:2.1	source_10	컨테이너 이미지
product-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = 8081	source_10	컨테이너 포트
product-service-svc.Service.metadata.name = product-service-svc	source_12	Service 이름
product-service-svc.Service.metadata.labels.app = product-service	source_11	공통 라벨(app)
product-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨(project)
product-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
product-service-svc.Service.spec.ports[0].port = 8081	source_12	서비스 포트
wish-service.Deployment.metadata.name = wish-service	source_4	Deployment 이름
wish-service.Deployment.metadata.labels.app = wish-service	source_11	공통 라벨(app)
wish-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨(project)
wish-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
wish-service.Deployment.spec.replicas = 1	source_4	파드 복제 수
wish-service.Deployment.spec.template.spec.containers[0].image = sk124590/wish-service:2.1	source_4	컨테이너 이미지
wish-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = 8082	source_4	컨테이너 포트
wish-service-svc.Service.metadata.name = wish-service-svc	source_12	Service 이름
wish-service-svc.Service.metadata.labels.app = wish-service	source_11	공통 라벨(app)
wish-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨(project)
wish-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
wish-service-svc.Service.spec.ports[0].port = 8082	source_12	서비스 포트
order-service.Deployment.metadata.name = order-service	source_7	Deployment 이름
order-service.Deployment.metadata.labels.app = order-service	source_11	공통 라벨(app)
order-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨(project)
order-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)

YAML 위치와 내용	참조된 Source	비고
order-service.Deployment.spec.replicas = 1	source_7	파드 복제 수
order-service.Deployment.spec.template.spec.containers[0].image = sk124590/order-service:2.1	source_7	컨테이너 이미지
order-service.Deployment.spec.template.spec.containers[0].ports[0].containerPort = 8083	source_7	컨테이너 포트
order-service-svc.Service.metadata.name = order-service-svc	source_12	Service 이름
order-service-svc.Service.metadata.labels.app = order-service	source_11	공통 라벨(app)
order-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨(project)
order-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
order-service-svc.Service.spec.ports[0].port = 8083	source_12	서비스 포트
gift-app.Deployment.metadata.name = gift-app	source_10	Deployment 이름
gift-app.Deployment.metadata.labels.app = gift-app	source_11	공통 라벨(app)
gift-app.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨(project)
gift-app.Deployment.metadata.labels.tier = frontend	source_11	공통 라벨(tier)
gift-app.Deployment.spec.replicas = 1	source_10	파드 복제 수
gift-app.Deployment.spec.template.spec.containers[0].image = sk124590/gift-app:2.1	source_10	컨테이너 이미지
gift-app.Deployment.spec.template.spec.containers[0].ports[0].containerPort = 8085	source_10	컨테이너 포트
gift-app-svc.Service.metadata.name = gift-app-svc	source_12	Service 이름
gift-app-svc.Service.metadata.labels.app = gift-app	source_11	공통 라벨(app)
gift-app-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨(project)
gift-app-svc.Service.metadata.labels.tier = frontend	source_11	공통 라벨(tier)
gift-app-svc.Service.spec.ports[0].port = 8080	source_12	외부 포트
gift-app-svc.Service.spec.type = LoadBalancer	source_12	서비스 타입

YAML 위치와 내용	참조된 Source	비고
gift-secret.Secret.metadata.name = gift-secret	source_5	Secret 이름
gift-secret.Secret.metadata.labels.app = gift-service	source_11	공통 라벨 (app)
gift-secret.Secret.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
gift-secret.Secret.metadata.labels.tier = backend	source_11	공통 라벨(tier)
gift-secret.Secret.data.JWT_SECRET = WW4ya2ppYmRkRkFXdG5QSjJBRmxMOFdYbW9oSk1DdmInUWdnYUV5cGE1RT0=	source_5	JWT 시크릿 (Base64)
gift-secret.Secret.data.KAKAO_CLIENT_ID = YWMYMGQ4NGY5N2I2NzlmMGJiZDIhODA2NWJhODIzZjM=	source_5	카카오 Client ID(Base64)
gift-secret.Secret.data.KAKAO_CLIENT_SECRET = RnczSmlMNVBrcGV2WFZaOU9pVk04YmZiZ0hVUEQ1RUE=	source_5	카카오 Client Secret(Base64)
gift-config.ConfigMap.metadata.name = gift-config	source_5	ConfigMap 이름
gift-config.ConfigMap.metadata.labels.app = gift-service	source_11	공통 라벨 (app)
gift-config.ConfigMap.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
gift-config.ConfigMap.metadata.labels.tier = backend	source_11	공통 라벨(tier)

YAML 위치와 내용	참조된 Source	비고
gift-config.ConfigMap.data.API_GATEWAY_URI = http://api-gateway-svc:8080	source_5	게이트웨이 내부 URI
gift-config.ConfigMap.data.SERVICE_USER_URI = http://user-service-svc:8084	source_5	사용자 서비스 URI
gift-config.ConfigMap.data.SERVICE_PRODUCT_URI = http://product-service-svc:8081	source_5	상품 서비스 URI
gift-config.ConfigMap.data.SERVICE_WISH_URI = http://wish-service-svc:8082	source_5	위시 서비스 URI
gift-config.ConfigMap.data.SERVICE_ORDER_URI = http://order-service-svc:8083	source_5	주문 서비스 URI
gift-config.ConfigMap.data.KAKAO_REDIRECT_URI = http://localhost:8080/members/login/oauth2/code/kakao	source_5	카카오 Redirect URI
gift-config.ConfigMap.data.FRONT_DOMAIN = http://localhost:8080	source_5	프런트 도메인
api-gateway.Deployment.metadata.name = api-gateway	source_1	Deployment 이름
api-gateway.Deployment.metadata.labels.app = api-gateway	source_11	공통 라벨 (app)
api-gateway.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
api-gateway.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
api-gateway.Deployment.spec.replicas = 1	source_6	파드 복제 수
api-gateway.Deployment.spec.selector.matchLabels.app = api-gateway	source_11	셀렉터(app)
api-gateway.Deployment.spec.template.metadata.labels.app = api-gateway	source_11	파드 라벨 (app)
api-gateway.Deployment.spec.template.metadata.labels.project = gift-service	source_11	파드 라벨 (project)
api-gateway.Deployment.spec.template.metadata.labels.tier = backend	source_11	파드 라벨(tier)
api-gateway.Deployment.spec.template.spec.containers[0].image = sk124590/api-gateway:2.1	source_6	컨테이너 이미지
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.SERVICE_USER_URI.valueFrom.configMapKeyRef.key = SERVICE_USER_URI	source_6	CM 키 참조
api-gateway.Deployment.spec.template.spec.containers[0].env.SERVICE_PRODUCT_URI.valueFrom.configMapKeyRef.key = SERVICE_PRODUCT_URI	source_6	CM 키 참조
api-gateway.Deployment.spec.template.spec.containers[0].env.SERVICE_WISH_URI.valueFrom.configMapKeyRef.key = SERVICE_WISH_URI	source_6	CM 키 참조
api-gateway.Deployment.spec.template.spec.containers[0].env.SERVICE_ORDER_URI.valueFrom.configMapKeyRef.key = SERVICE_ORDER_URI	source_6	CM 키 참조
api-gateway.Deployment.spec.template.spec.containers[0].env.JWT_SECRET.valueFrom.secretKeyRef.key = JWT_SECRET	source_6	Secret 키 참조
api-gateway.Deployment.spec.template.spec.containers[0].resources.requests.cpu = 100m	source_6	요청 CPU
api-gateway.Deployment.spec.template.spec.containers[0].resources.requests.memory = 256Mi	source_6	요청 메모리
api-gateway.Deployment.spec.template.spec.containers[0].resources.limits.cpu = 500m	source_6	제한 CPU

YAML 위치와 내용	참조된 Source	비고
api-gateway.Deployment.spec.template.spec.containers[0].resources.limits.memory = 512Mi	source_6	제한 메모리
api-gateway-svc.Service.metadata.name = api-gateway-svc	source_5	Service 이름
api-gateway-svc.Service.metadata.labels.app = api-gateway	source_11	공통 라벨 (app)
api-gateway-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
api-gateway-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
api-gateway-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
api-gateway-svc.Service.spec.selector.app = api-gateway	source_11	셀렉터(app)
api-gateway-svc.Service.spec.ports[0].port = 8080	source_12	서비스 포트
api-gateway-svc.Service.spec.ports[0].targetPort = 8080	source_12	타깃 포트
user-service.Deployment.metadata.name = user-service	source_1	Deployment 이름
user-service.Deployment.metadata.labels.app = user-service	source_11	공통 라벨 (app)
user-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
user-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
user-service.Deployment.spec.replicas = 1	source_8	파드 복제 수
user-service.Deployment.spec.selector.matchLabels.app = user-service	source_11	셀렉터(app)
user-service.Deployment.spec.template.metadata.labels.app = user-service	source_11	파드 라벨 (app)
user-service.Deployment.spec.template.metadata.labels.project = gift-service	source_11	파드 라벨 (project)
user-service.Deployment.spec.template.metadata.labels.tier = backend	source_11	파드 라벨(tier)
user-service.Deployment.spec.template.spec.containers[0].image = sk124590/user-service:2.1	source_8	컨테이너 이미지
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.JWT_SECRET.valueFrom.secretKeyRef.key = JWT_SECRET	source_8	Secret 키 참조
user-service.Deployment.spec.template.spec.containers[0].env.KAKAO_CLIENT_ID.valueFrom.secretKeyRef.key = KAKAO_CLIENT_ID	source_8	Secret 키 참조
user-service.Deployment.spec.template.spec.containers[0].env.KAKAO_CLIENT_SECRET.valueFrom.secretKeyRef.key = KAKAO_CLIENT_SECRET	source_8	Secret 키 참조
user-service.Deployment.spec.template.spec.containers[0].env.KAKAO_REDIRECT_URI.valueFrom.configMapKeyRef.key = KAKAO_REDIRECT_URI	source_8	CM 키 참조
user-service.Deployment.spec.template.spec.containers[0].resources.requests.cpu = 150m	source_8	요청 CPU
user-service.Deployment.spec.template.spec.containers[0].resources.requests.memory = 512Mi	source_8	요청 메모리
user-service.Deployment.spec.template.spec.containers[0].resources.limits.cpu = 1000m	source_8	제한 CPU
user-service.Deployment.spec.template.spec.containers[0].resources.limits.memory = 1Gi	source_8	제한 메모리
user-service-svc.Service.metadata.name = user-service-svc	source_12	Service 이름

YAML 위치와 내용	참조된 Source	비고
user-service-svc.Service.metadata.labels.app = user-service	source_11	공통 라벨 (app)
user-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
user-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
user-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
user-service-svc.Service.spec.selector.app = user-service	source_11	셀렉터(app)
user-service-svc.Service.spec.ports[0].port = 8084	source_12	서비스 포트
user-service-svc.Service.spec.ports[0].targetPort = 8084	source_12	타겟 포트
product-service.Deployment.metadata.name = product-service	source_1	Deployment 이름
product-service.Deployment.metadata.labels.app = product-service	source_11	공통 라벨 (app)
product-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
product-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
product-service.Deployment.spec.replicas = 1	source_10	파드 복제 수
product-service.Deployment.spec.selector.matchLabels.app = product-service	source_11	셀렉터(app)
product-service.Deployment.spec.template.metadata.labels.app = product-service	source_11	파드 라벨 (app)
product-service.Deployment.spec.template.metadata.labels.project = gift-service	source_11	파드 라벨 (project)
product-service.Deployment.spec.template.metadata.labels.tier = backend	source_11	파드 라벨(tier)
product-service.Deployment.spec.template.spec.containers[0].image = sk124590/product-service:2.1	source_10	컨테이너 이미지
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.requests.cpu = 150m	source_10	요청 CPU
product-service.Deployment.spec.template.spec.containers[0].resources.requests.memory = 512Mi	source_10	요청 메모리
product-service.Deployment.spec.template.spec.containers[0].resources.limits.cpu = 1000m	source_10	제한 CPU
product-service.Deployment.spec.template.spec.containers[0].resources.limits.memory = 1Gi	source_10	제한 메모리
product-service-svc.Service.metadata.name = product-service-svc	source_12	Service 이름
product-service-svc.Service.metadata.labels.app = product-service	source_11	공통 라벨 (app)
product-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
product-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
product-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
product-service-svc.Service.spec.selector.app = product-service	source_11	셀렉터(app)
product-service-svc.Service.spec.ports[0].port = 8081	source_12	서비스 포트
product-service-svc.Service.spec.ports[0].targetPort = 8081	source_12	타겟 포트
wish-service.Deployment.metadata.name = wish-service	source_1	Deployment 이름

YAML 위치와 내용	참조된 Source	비고
wish-service.Deployment.metadata.labels.app = wish-service	source_11	공통 라벨 (app)
wish-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
wish-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
wish-service.Deployment.spec.replicas = 1	source_4	파드 복제 수
wish-service.Deployment.spec.selector.matchLabels.app = wish-service	source_11	셀렉터(app)
wish-service.Deployment.spec.template.metadata.labels.app = wish-service	source_11	파드 라벨 (app)
wish-service.Deployment.spec.template.metadata.labels.project = gift-service	source_11	파드 라벨 (project)
wish-service.Deployment.spec.template.metadata.labels.tier = backend	source_11	파드 라벨(tier)
wish-service.Deployment.spec.template.spec.containers[0].image = sk124590/wish-service:2.1	source_4	컨테이너 이미지
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.SERVICE_PRODUCT_URI.valueFrom.configMapKeyRef.key = SERVICE_PRODUCT_URI	source_4	CM 키 참조
wish-service.Deployment.spec.template.spec.containers[0].resources.requests.cpu = 100m	source_4	요청 CPU
wish-service.Deployment.spec.template.spec.containers[0].resources.requests.memory = 512Mi	source_4	요청 메모리
wish-service.Deployment.spec.template.spec.containers[0].resources.limits.cpu = 700m	source_4	제한 CPU
wish-service.Deployment.spec.template.spec.containers[0].resources.limits.memory = 1Gi	source_4	제한 메모리
wish-service-svc.Service.metadata.name = wish-service-svc	source_12	Service 이름
wish-service-svc.Service.metadata.labels.app = wish-service	source_11	공통 라벨 (app)
wish-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
wish-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
wish-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
wish-service-svc.Service.spec.selector.app = wish-service	source_11	셀렉터(app)
wish-service-svc.Service.spec.ports[0].port = 8082	source_12	서비스 포트
wish-service-svc.Service.spec.ports[0].targetPort = 8082	source_12	타깃 포트
order-service.Deployment.metadata.name = order-service	source_1	Deployment 이름
order-service.Deployment.metadata.labels.app = order-service	source_11	공통 라벨 (app)
order-service.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
order-service.Deployment.metadata.labels.tier = backend	source_11	공통 라벨(tier)
order-service.Deployment.spec.replicas = 1	source_7	파드 복제 수
order-service.Deployment.spec.selector.matchLabels.app = order-service	source_11	셀렉터(app)
order-service.Deployment.spec.template.metadata.labels.app = order-service	source_11	파드 라벨 (app)

YAML 위치와 내용	참조된 Source	비고
order-service.Deployment.spec.template.metadata.labels.project = gift-service	source_11	파드 라벨 (project)
order-service.Deployment.spec.template.metadata.labels.tier = backend	source_11	파드 라벨(tier)
order-service.Deployment.spec.template.spec.containers[0].image = sk124590/order-service:2.1	source_7	컨테이너 이미지
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.SERVICE_PRODUCT_URI.valueFrom.configMapKeyRef.key = SERVICE_PRODUCT_URI	source_7	CM 키 참조
order-service.Deployment.spec.template.spec.containers[0].env.SERVICE_WISH_URI.valueFrom.configMapKeyRef.key = SERVICE_WISH_URI	source_7	CM 키 참조
order-service.Deployment.spec.template.spec.containers[0].env.SERVICE_USER_URI.valueFrom.configMapKeyRef.key = SERVICE_USER_URI	source_7	CM 키 참조
order-service.Deployment.spec.template.spec.containers[0].env.FRONT_DOMAIN.valueFrom.configMapKeyRef.key = FRONT_DOMAIN	source_7	CM 키 참조
order-service.Deployment.spec.template.spec.containers[0].env.KAKAO_CLIENT_ID.valueFrom.secretKeyRef.key = KAKAO_CLIENT_ID	source_7	Secret 키 참조
order-service.Deployment.spec.template.spec.containers[0].env.KAKAO_CLIENT_SECRET.valueFrom.secretKeyRef.key = KAKAO_CLIENT_SECRET	source_7	Secret 키 참조
order-service.Deployment.spec.template.spec.containers[0].resources.requests.cpu = 100m	source_7	요청 CPU
order-service.Deployment.spec.template.spec.containers[0].resources.requests.memory = 512Mi	source_7	요청 메모리
order-service.Deployment.spec.template.spec.containers[0].resources.limits.cpu = 700m	source_7	제한 CPU
order-service.Deployment.spec.template.spec.containers[0].resources.limits.memory = 1Gi	source_7	제한 메모리
order-service-svc.Service.metadata.name = order-service-svc	source_12	Service 이름
order-service-svc.Service.metadata.labels.app = order-service	source_11	공통 라벨 (app)
order-service-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
order-service-svc.Service.metadata.labels.tier = backend	source_11	공통 라벨(tier)
order-service-svc.Service.spec.type = ClusterIP	source_12	서비스 타입
order-service-svc.Service.spec.selector.app = order-service	source_11	셀렉터(app)
order-service-svc.Service.spec.ports[0].port = 8083	source_12	서비스 포트
order-service-svc.Service.spec.ports[0].targetPort = 8083	source_12	타겟 포트
gift-app.Deployment.metadata.name = gift-app	source_1	Deployment 이름
gift-app.Deployment.metadata.labels.app = gift-app	source_11	공통 라벨 (app)
gift-app.Deployment.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
gift-app.Deployment.metadata.labels.tier = frontend	source_11	공통 라벨(tier)
gift-app.Deployment.spec.replicas = 1	source_10	파드 복제 수
gift-app.Deployment.spec.selector.matchLabels.app = gift-app	source_11	셀렉터(app)

YAML 위치와 내용	참조된 Source	비고
gift-app.Deployment.spec.template.metadata.labels.app = gift-app	source_11	파드 라벨 (app)
gift-app.Deployment.spec.template.metadata.labels.project = gift-service	source_11	파드 라벨 (project)
gift-app.Deployment.spec.template.metadata.labels.tier = frontend	source_11	파드 라벨(tier)
gift-app.Deployment.spec.template.spec.containers[0].image = sk124590/gift-app:2.1	source_10	컨테이너 이미지
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.API_GATEWAY_URI.valueFrom.configMapKeyRef.key = API_GATEWAY_URI	source_10	CM 키 참조
gift-app.Deployment.spec.template.spec.containers[0].env.KAKAO_REDIRECT_URI.valueFrom.configMapKeyRef.key = KAKAO_REDIRECT_URI	source_10	CM 키 참조
gift-app.Deployment.spec.template.spec.containers[0].env.JWT_SECRET.valueFrom.secretKeyRef.key = JWT_SECRET	source_10	Secret 키 참조
gift-app.Deployment.spec.template.spec.containers[0].env.KAKAO_CLIENT_ID.valueFrom.secretKeyRef.key = KAKAO_CLIENT_ID	source_10	Secret 키 참조
gift-app.Deployment.spec.template.spec.containers[0].env.KAKAO_CLIENT_SECRET.valueFrom.secretKeyRef.key = KAKAO_CLIENT_SECRET	source_10	Secret 키 참조
gift-app.Deployment.spec.template.spec.containers[0].resources.requests.cpu = 100m	source_10	요청 CPU
gift-app.Deployment.spec.template.spec.containers[0].resources.requests.memory = 512Mi	source_10	요청 메모리
gift-app.Deployment.spec.template.spec.containers[0].resources.limits.cpu = 500m	source_10	제한 CPU
gift-app.Deployment.spec.template.spec.containers[0].resources.limits.memory = 1Gi	source_10	제한 메모리
gift-app-svc.Service.metadata.name = gift-app-svc	source_12	Service 이름
gift-app-svc.Service.metadata.labels.app = gift-app	source_11	공통 라벨 (app)
gift-app-svc.Service.metadata.labels.project = gift-service	source_11	공통 라벨 (project)
gift-app-svc.Service.metadata.labels.tier = frontend	source_11	공통 라벨(tier)
gift-app-svc.Service.spec.type = LoadBalancer	source_12	서비스 타입
gift-app-svc.Service.spec.selector.app = gift-app	source_11	셀렉터(app)
gift-app-svc.Service.spec.ports[0].port = 8080	source_12	외부 포트
gift-app-svc.Service.spec.ports[0].targetPort = 8085	source_12	타겟 포트