

## 5. Connect to and consume Azure services and third-party services

### └ 5.1 Implement API Management

#### └ 5.1.2 Create and document APIs

1. How do you import an API into Azure API Management?
2. What API definition formats are supported for import?
3. What are the common ways to create an API manually in APIM?
4. What is the purpose of API operations and how are they defined?
5. How do you add or modify request/response parameters in APIM?
6. How does versioning work in APIM for APIs?
7. How do you document APIs using OpenAPI (Swagger) in APIM?
8. What is the role of the developer portal in API documentation?
9. How do you secure developer portal access?
10. What tools support automated API documentation publishing to APIM?

---

#### 1. How do you import an API into Azure API Management?

Use the Azure Portal, CLI, or ARM/Bicep to import via OpenAPI, WSDL, GraphQL, or direct URL.

```
az apim api import \  
  --resource-group myrg \  
  --service-name myapim \  
  --path myapi \  
  --specification-format OpenApi \  
  --specification-path ./openapi.json
```

---

#### 2. What API definition formats are supported for import?

- OpenAPI 2.0/3.0 (JSON/YAML)
- WSDL (SOAP)
- GraphQL schemas
- Azure Functions or App Services (via App Insights tracing)

---

#### 3. What are the common ways to create an API manually in APIM?

- Define operations manually in the Azure Portal
- Use Azure CLI or ARM templates
- Add operations under a base path, define HTTP verbs, request parameters, and response types

---

#### 4. What is the purpose of API operations and how are they defined?

Operations define how the API is consumed (e.g., GET /users). Each includes:

- HTTP method
- URL template
- Request/response schema
- Optional policies (rate limits, auth)

---

#### 5. How do you add or modify request/response parameters in APIM?

Via Portal or CLI:

- Define path/query/header parameters for requests
- Define expected status codes and response body schemas
- Example: Add header param in operation settings under Inbound Processing

## 6. How does versioning work in APIM for APIs?

Supported methods:

- **Path-based** (e.g., /v1/orders)
- **Query string** (e.g., ?api-version=1.0)
- **Header-based** (e.g., X-Version: 1.0)

APIM supports version sets to group versions under a single API entity.

---

## 7. How do you document APIs using OpenAPI (Swagger) in APIM?

Import an OpenAPI file to auto-generate operations and documentation.

OpenAPI description appears in both the Azure Portal and developer portal.

Supports Swagger UI rendering in developer portal.

---

## 8. What is the role of the developer portal in API documentation?

Provides a public or private interface for users to:

- Discover and test APIs
  - View request/response details
  - Access API keys
  - Read autogenerated documentation
- 

## 9. How do you secure developer portal access?

- Enable **require sign-in** in portal settings
  - Configure identity providers (AAD, Facebook, etc.)
  - Assign users to products with subscription-based access
- 

## 10. What tools support automated API documentation publishing to APIM?

- **Swagger/OpenAPI CLI**
- **Postman** → export to OpenAPI
- **Azure DevOps/GitHub Actions** → automate import via `az apim api import`
- Supports CI/CD for keeping API definitions in sync