

# API and webservices testing

Cybertek

# PLAN

What is API?

Web services

Application architecture

Manual testing with postman

Request

Response

Web services testing

# What does it mean?

---

## Application user interface

API is an interface or communication protocol between different applications or parts of same application





## What does it do?



API is a means of communication. Humans communicate using speech, body language. Applications talk to each other using API.

Cybertek

Using API we gain control to the operations of the application.

Any Libraries that we use in our project to establish connection to anything/ or to make our interaction with certain set of resources easier, we can call it an API.

# What does it look like?

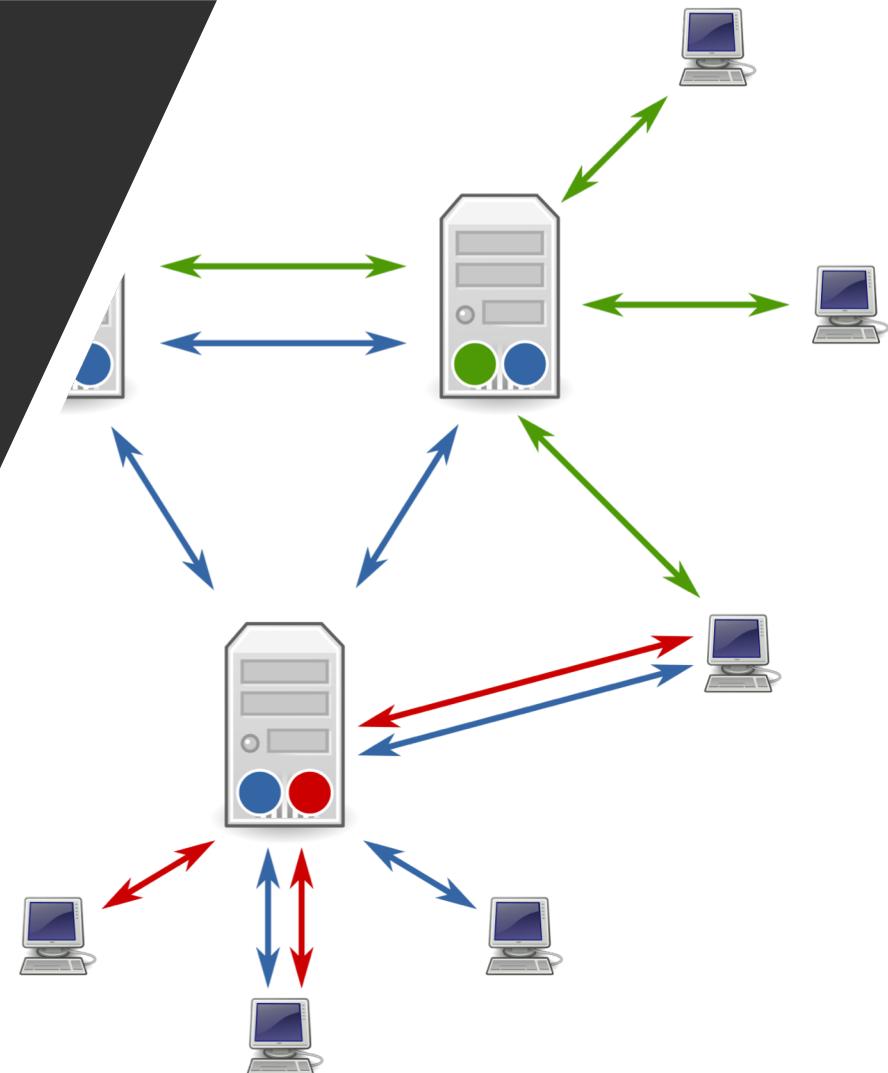
- Jar files
- DLL libraries
- Operating System
- Web services

```
WebDriver driver = new ChromeDriver();
driver.get("https://cybertekschool.com");
System.out.println(driver.getTitle());
driver.quit();
```

Cybertek

# Web service

- Just another form of API
- Type of API that we can access over the network
- Can be based on REST or SOAP



# SOAP and REST

REST	SOAP
It is an architectural style that is protocol independent	XML based message protocol
Documentation is easy to understand	Documentation is complex and hard to understand
requires use of HTTP	Language, platform, and transport independent, can use SMTP, FTP etc
More efficient and faster	Slower compared to rest
Requires less bandwidth and resources	Requires more bandwidth and resources
Less secure	More secure
REST can use the data JSON, XML, CSV, YAML, plain text, etc	SOAP relies exclusively on XML
Uses URI to expose business logic	Uses services interfaces to expose business logic

Schemes  
HTTP

**pet** Everything about your Pets

**POST** /pet Add a new pet to the store

**PUT** /pet Update an existing pet

**GET** /pet/findByStatus Finds Pets by status

**GET** /pet/findByTags Finds Pets by tags

**GET** /pet/{petId} Find pet by ID

**POST** /pet/{petId} Updates a pet in the store with form data

**DELETE** /pet/{petId} Deletes a pet

**POST** /pet/{petId}/uploadImage uploads an image

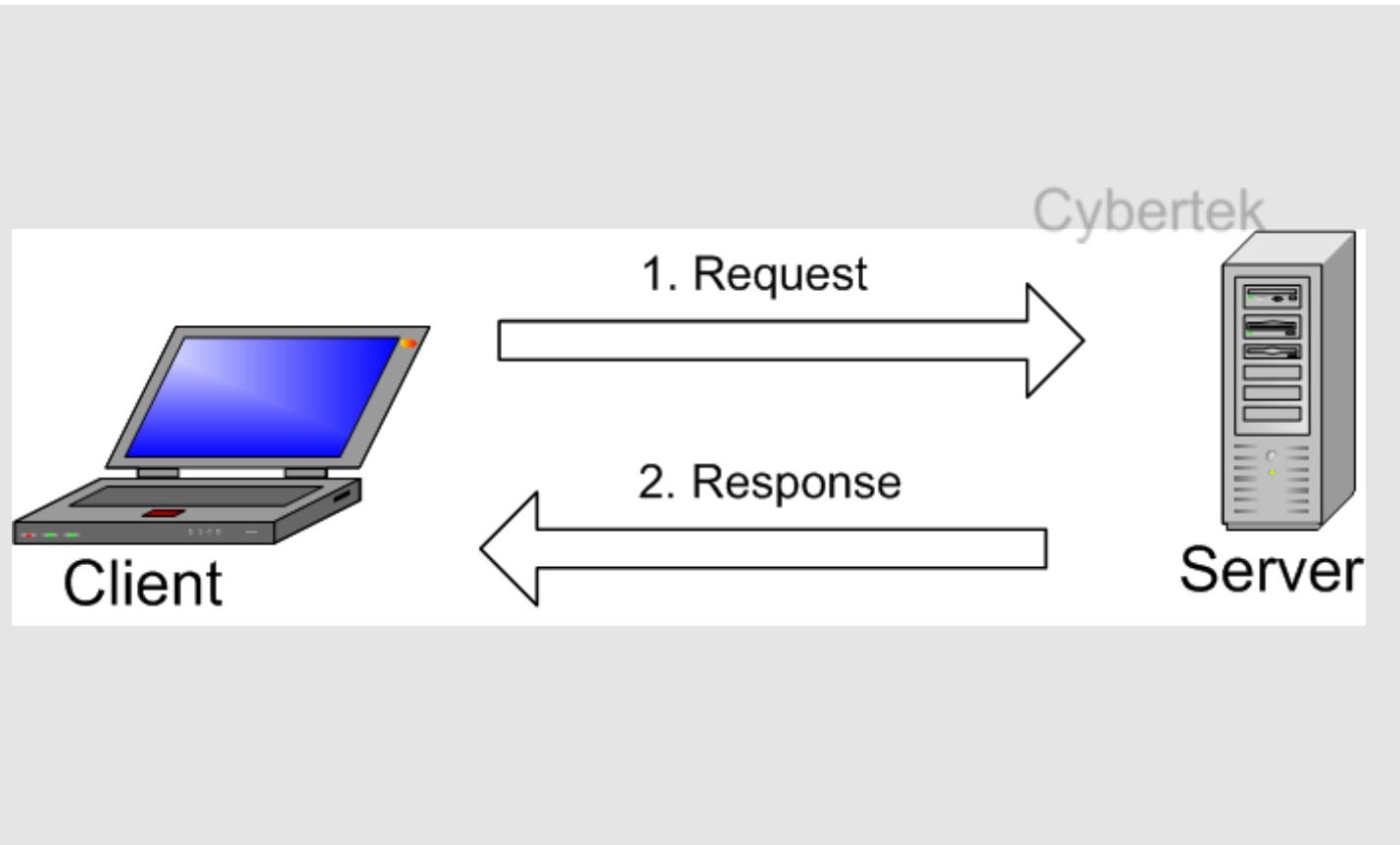
**store** Access to Petstore orders

# API documentation

Cybertek

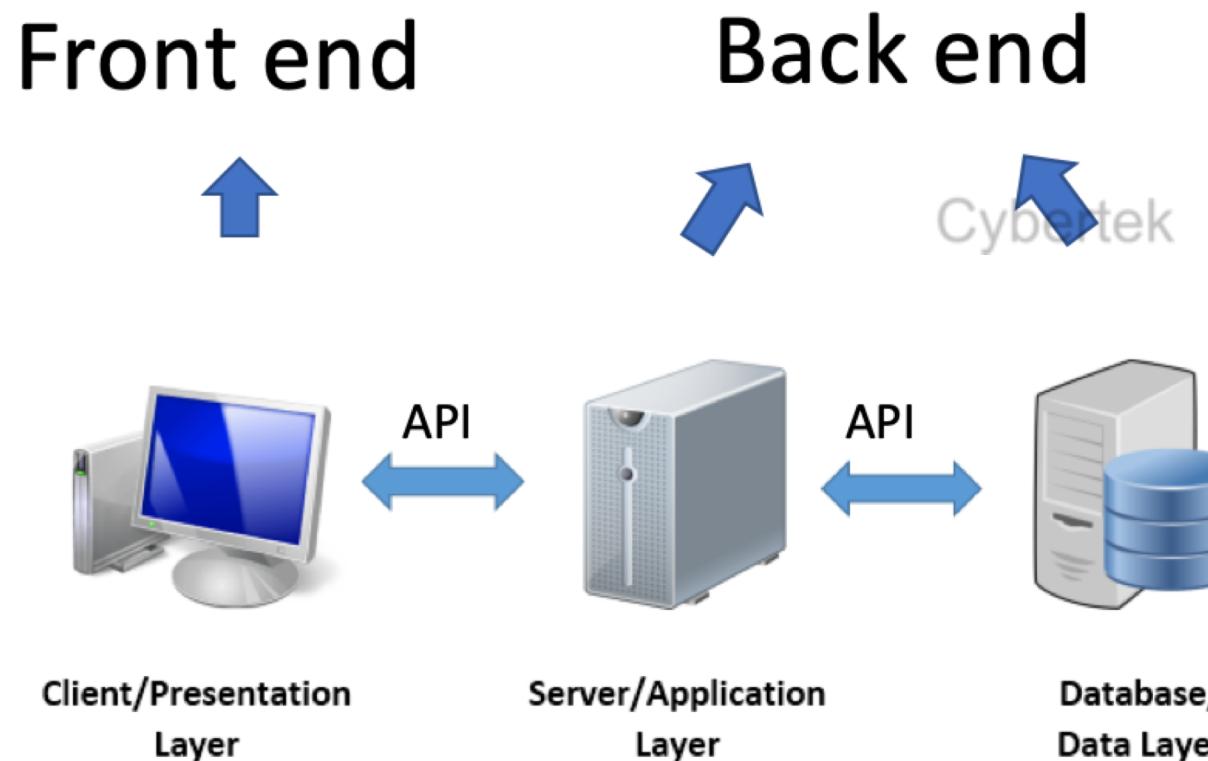
- Quick and concise reference containing what you need to know to use certain API
- Most companies produce documentation for their API
- Public API documentations are published in company website while private API documentation are only accessible internally.
- Swagger is a tool used to create REST web services documentation

# How web services work?



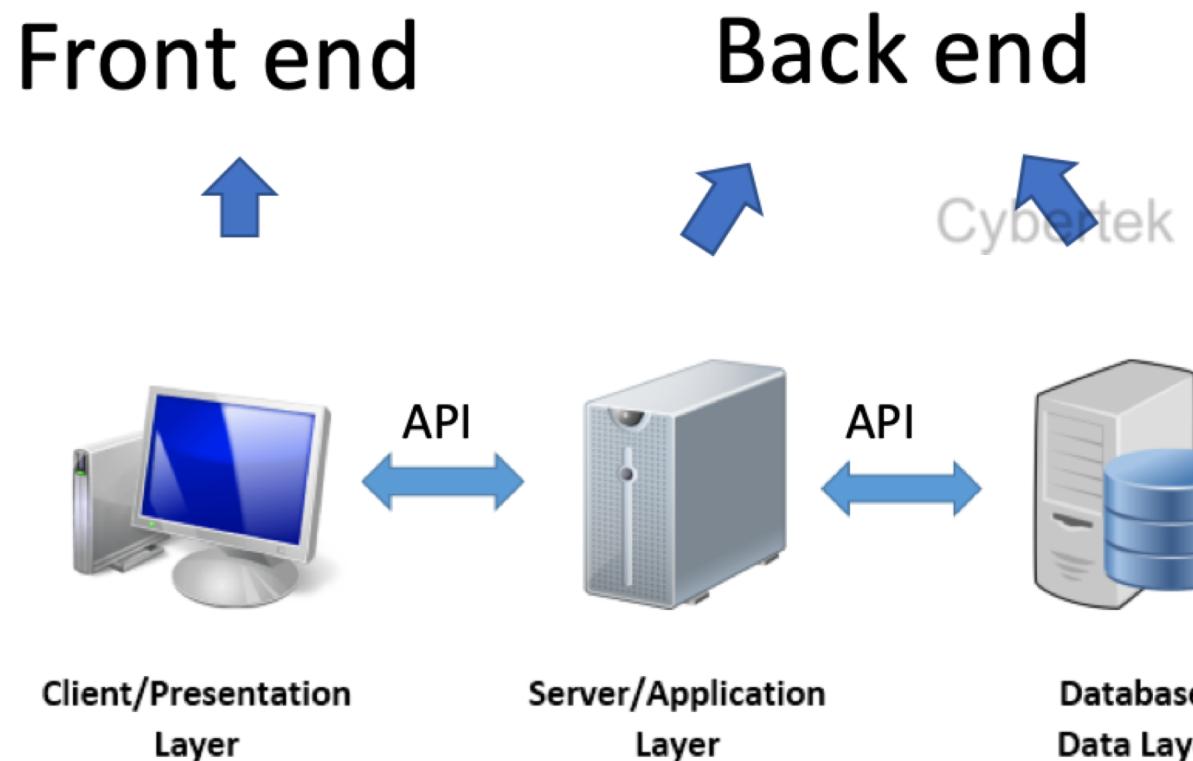
- When we want something from a web service, we send a request
- Whoever (or whatever) sends the request is called the client
- Our request is consumed and processed by server
- The server processes the request and returns us a response

# 3 tier architecture



- Presentation layer is front end layer that consists of the user interface accessible through a web browser
- Application layer aka business layer contains the functional business logic which drives an application's core capabilities
- Data layer is a database management system that provides access to the data

# 3 tier architecture

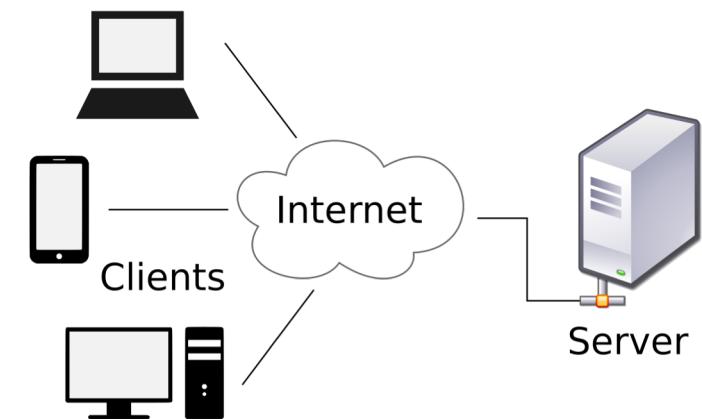


- Presentation layer is built using HTML, JavaScript and CSS by frameworks like React, Angular
- Application layer can be written in C#, Java, C++, Python, Ruby
- Data layer can be any relational or non relational database such as MySQL, Oracle, PostgreSQL or Mongo

# Client server application

- Client–server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients
- Client connects to server using the network
- Web applications are also client server applications
- In web applications the browser is the client

Cybertek



# REST API CRUD operations

Operation	SQL	HTTP	Action
Create	insert	<b>POST</b>	<i>Create</i>
Read	select	<b>GET</b>	<i>Retrieve</i>
Update	update	<b>PUT</b> <b>PATCH</b>	<i>Update/Replace</i> <i>Update/Modify</i>
Delete	delete	<b>DELETE</b>	<i>Remove</i>

# Request. What is it made of?



---

**ENDPOINT**

---

**METHOD**  
Cybertek

---

**HEADERS**

---

**PARAMETERS**

---

**BODY**

---

# Request. What is it made of?

---



## STATUS CODE

---

Cybertek

## HEADER

---

## BODY

---

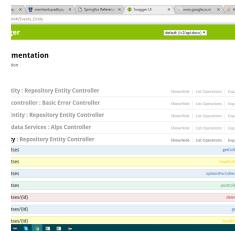
Hold on. One by one please.

Cybertek

# Endpoint



Specific URL where we send our requests to a certain web service.

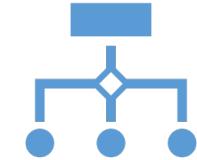


We can learn about the endpoints in the documentation

Cybertek



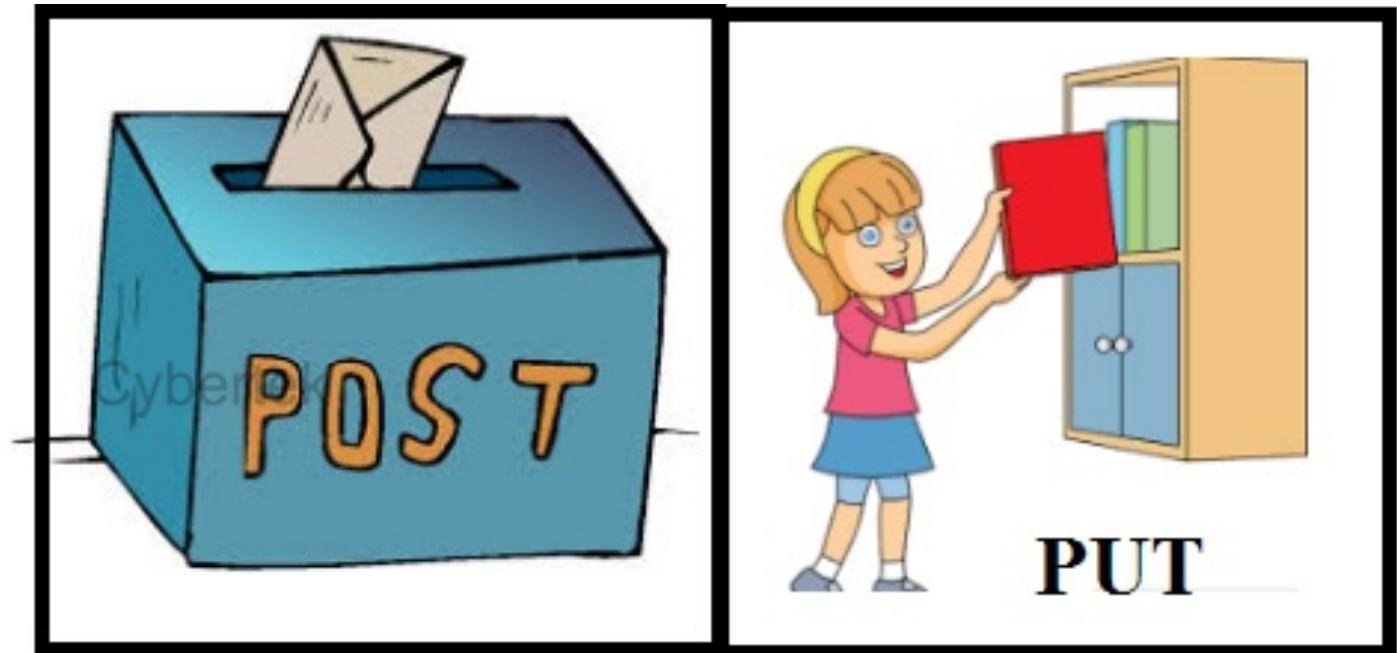
End point contains the base URL and path to specific resource.



Resource is the data is returned as a result of the request to a specific endpoint.

# Methods

- Get
- Post
- Put
- Patch
- Delete





## Status code

- Status codes are issued by a server in response to a client's request
- HTTP response status codes indicate whether a specific HTTP request has been successfully completed.

# Header

**Request Header**

Name	Content-type
Value	application/json
<input type="checkbox"/> Save to favorite	<b>Okay</b>

Represents the metadata associated with the request or response. They define the operating parameters of request/response.

Cybertek

- **Accept-Charset:** This is a header which is set with the request and tells the server about which character sets are acceptable by the client.
- **Content-Type:** Indicates the media type (text/html or text/JSON) of the response sent to the client by the server, this will help the client in processing the response body correctly.

# Parameters



- Parameters are options that can be passed with the endpoint to influence the response
- Parameters help the caller to further specify what specific resource needs to be returned or what sceptic specific action to be taken by the server
- We can learn about the what parameters available and how to use them in the documentation

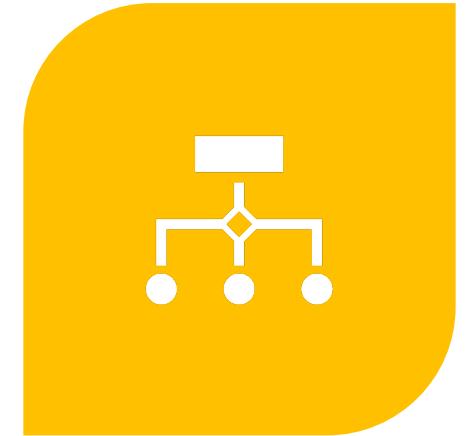
# Body



BODY CONTAINS THE IS SENT BY REQUEST OR RECEIVED BY RESPONSE.

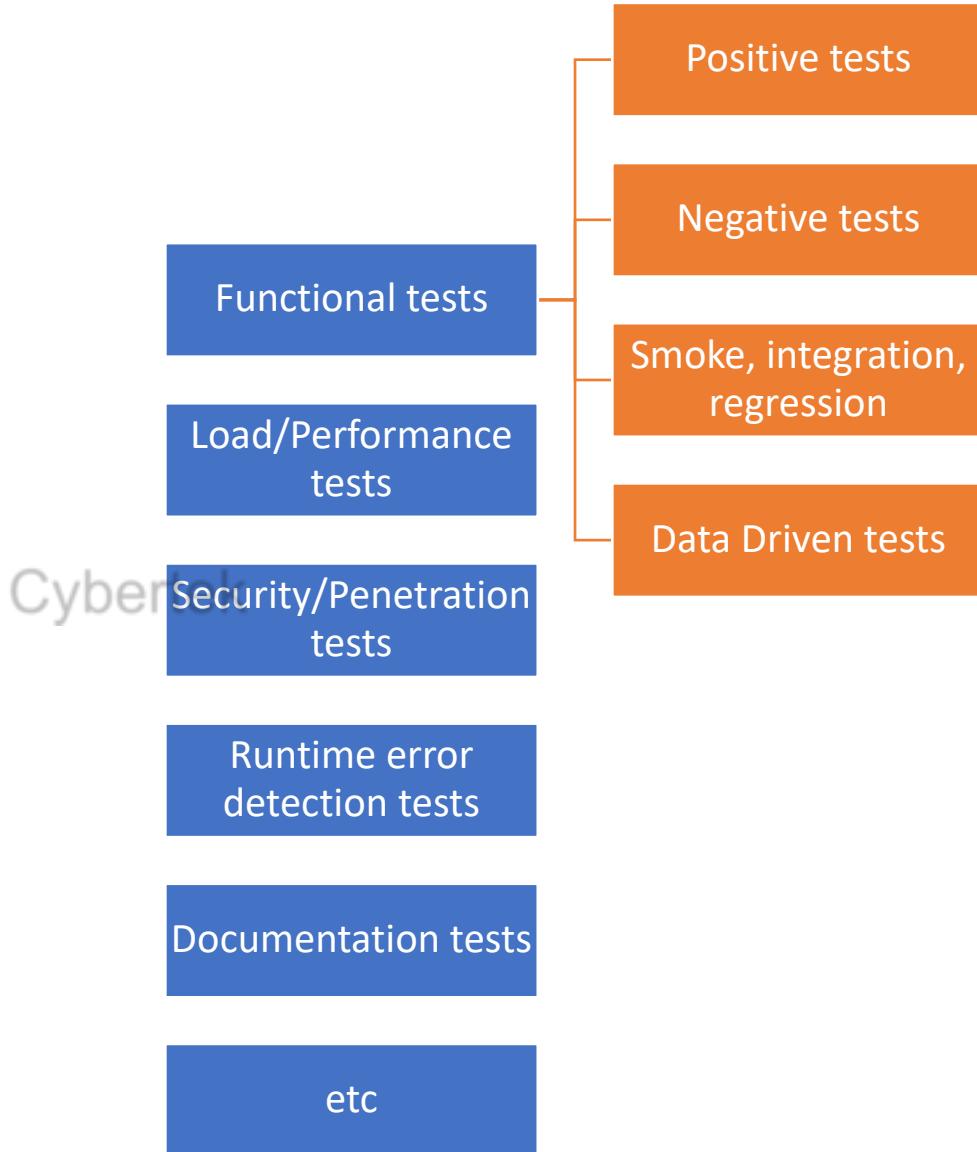


HTTP BODY DATA IS THE DATA BYTES TRANSMITTED IN AN HTTP TRANSACTION MESSAGE IMMEDIATELY FOLLOWING THE HEADERS



DATA IN THE BODY CAN BE IN DIFFERENT FORMAT

# What tests can we do on a webservice?



# How do we test a web service?

