API Testing

Introduction

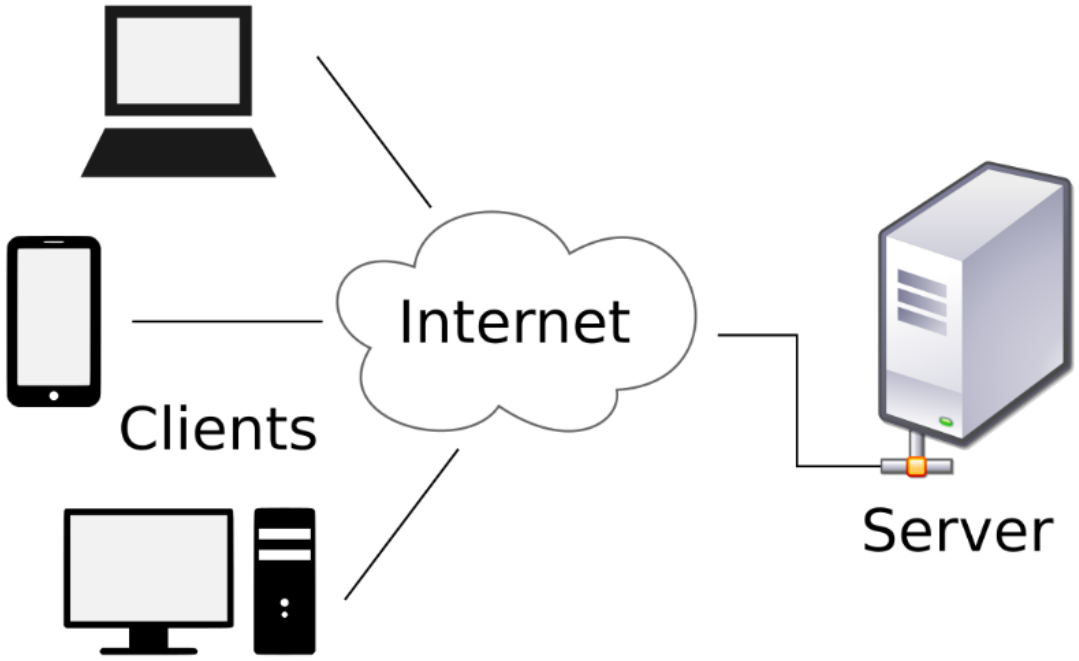
What is Client & Server?

Client is a device or software (like a computer, phone, or browser) that sends requests to get services or data.

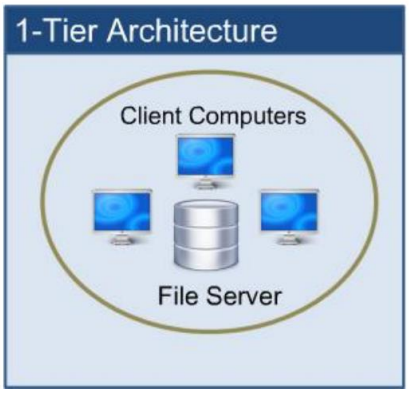
For example, when you use a browser to open a website, your browser acts as the client.

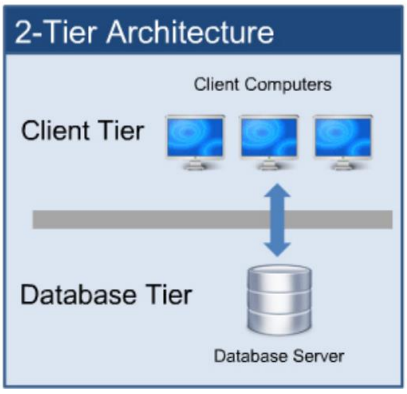
Server is a powerful computer or software that receives those requests, processes them, and sends back the requested data or service.

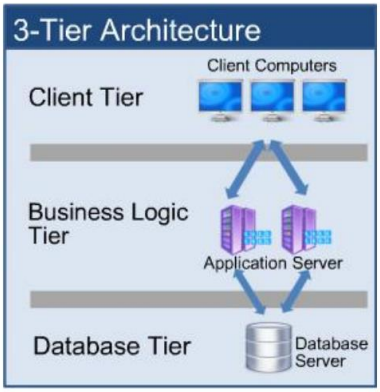
For instance, when you open a website, the server provides the webpage to your browser.

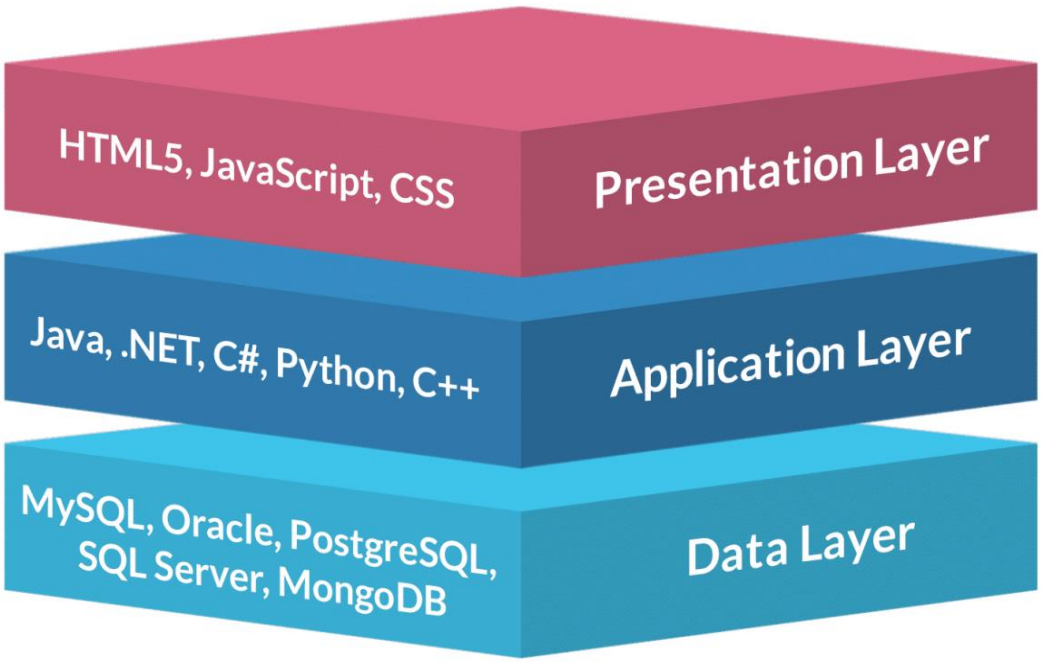


Client/Server Architecture









What is an API?

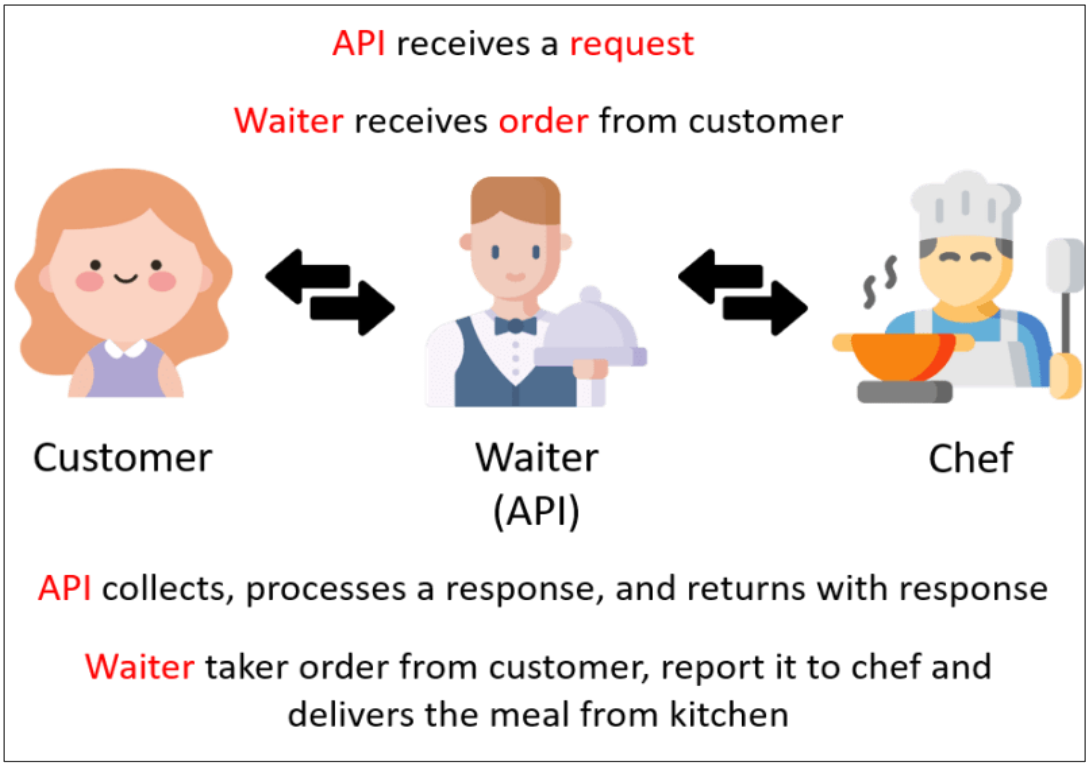
API (Application Programming Interface) is like a messenger that helps two programs or systems talk to each other and share information.

For example:

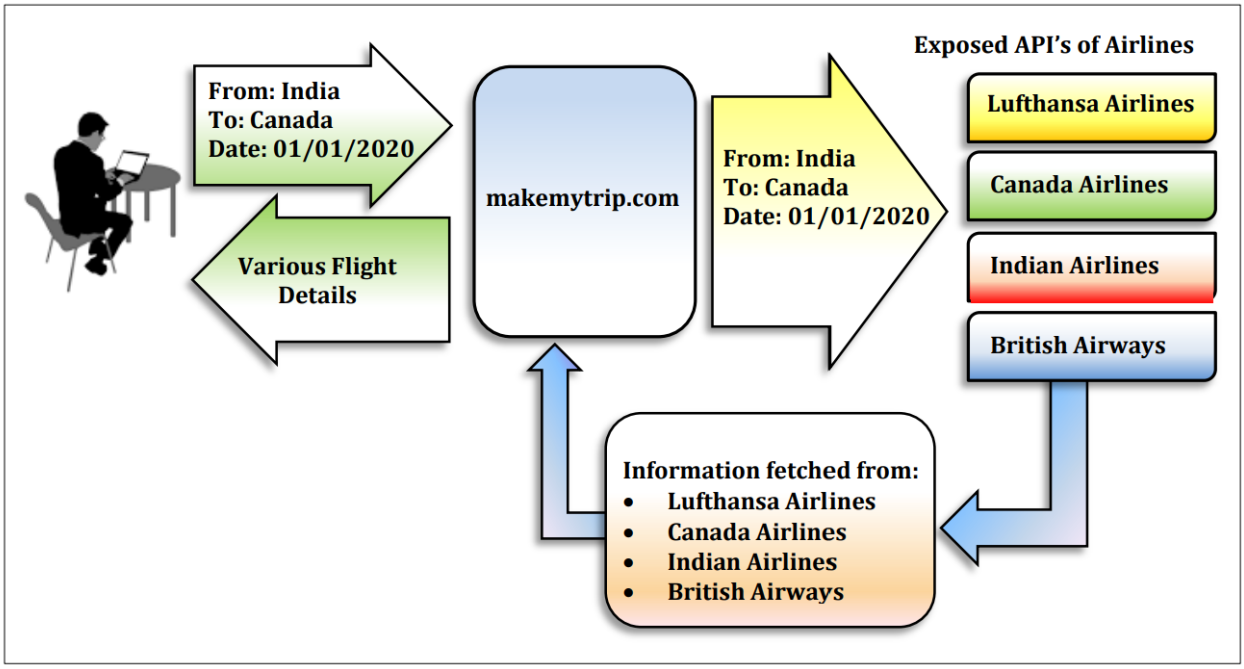
- Imagine you're using a **food delivery app**. When you select a restaurant and place an order, the app uses an API to communicate with the restaurant's system to check the menu, place your order, and confirm it.
- APIs define the rules for how this communication happens, like what data can be sent, how to ask for it, and what the response will look like.

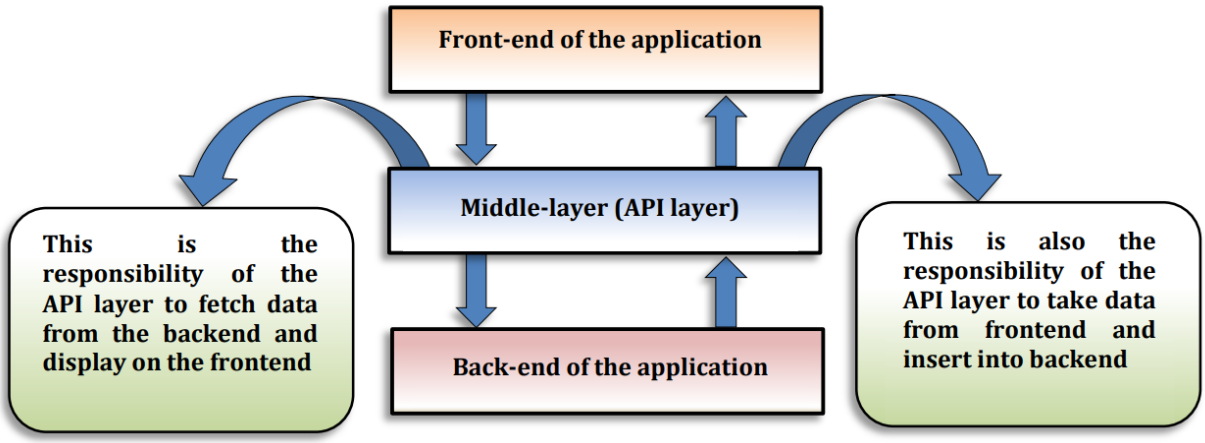
It's **like a waiter in a restaurant**, taking your request to the kitchen and bringing back your food. The API (waiter) ensures both sides (you and the kitchen) understand each other!

API - Restaurant Analogy



API - MakeMyTrip Analogy





**** API is a way of communication between two layers.**

Types Of API

APIs come in different types based on how and where they are used.

Open APIs (Public APIs):

- These are open for everyone to use. Developers can access them without restrictions (or with minimal requirements).
- Example: Google Maps API lets any app show maps and directions.

Internal APIs (Private APIs):

- These are used only within a company or organization. They help internal systems or teams communicate securely.
- Example: A company's HR system API to access employee data for payroll processing.

Partner APIs:

- These are shared with specific partners or businesses. They require special permissions or agreements to use.
- Example: A travel app using an airline's API to show flight details.

Composite APIs:

- These combine multiple APIs into one call, allowing you to get data from different sources in a single request.
- Example: An e-commerce app using a composite API to retrieve product info, pricing, and customer reviews all at once.

API Vs Webservice Vs Microservice

Here's a simple comparison of API, Web Service, and Microservice:

1.API (Application Programming Interface)

- **What it is:** A way for different software systems or applications to communicate with each other.
- **How it works:** It defines rules for how one app can request data or services from another. APIs don't have to use the internet—they can work locally, too.

- **Example:** Your phone's weather app uses an API to get weather data from a weather server.

2. Web Service

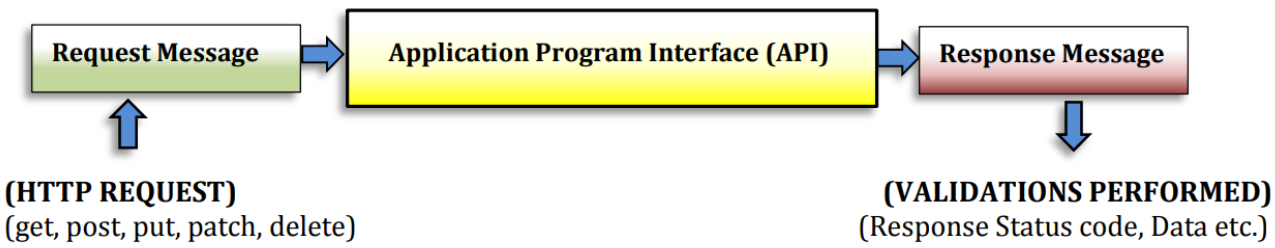
- **What it is:** A type of API that works specifically over the internet (using HTTP).
- **How it works:** It enables two systems to exchange data (often using XML or JSON).
- **Example:** A payment gateway like PayPal's web service allows e-commerce websites to process payments online.

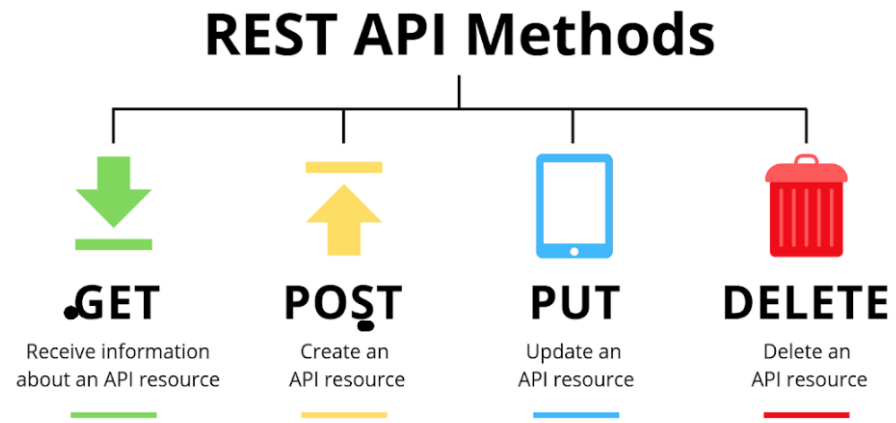
Key point: All web services are APIs, but not all APIs are web services.

3. Microservice

- **What it is:** A small, independent part of an application that does one thing well. A large app is built by combining many microservices.
- **How it works:** Each microservice communicates with other microservices via APIs. They are self-contained and can run independently.
- **Example:** In an online shopping app, separate microservices might handle user login, product catalogue**,** payments, and order tracking.

Rest API HTTP Methods





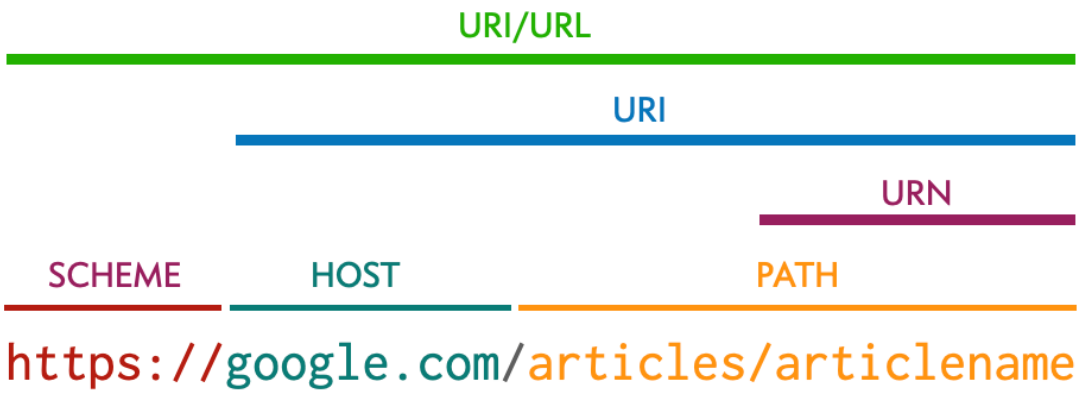
http Vs https

Terminologies

URI – Uniform Resource Identifier

URL – Uniform Resource Locator

URN – Uniform Resource Name



Feature & Resource

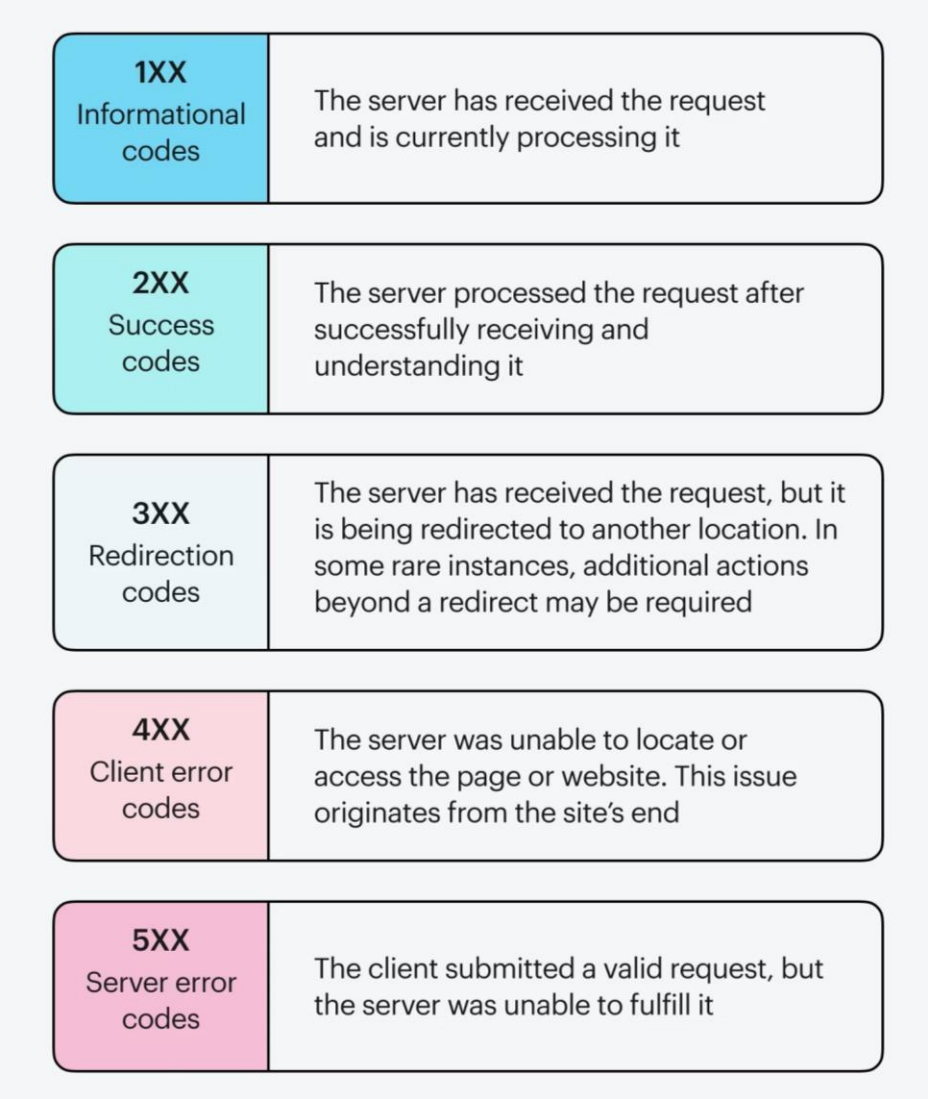
'**Feature'** is the term used in manual testing to test some functionality and similarly '**Resource'** is the term used in API Automation testing referring some functionality.

Payload

payload means **body in the HTTP request and response message**.

- Request Payload
- Response Payload

HTTP Status Codes



1XX Information	
100	Continue
101	Switching Protocols
102	Processing
103	Early Hints
2XX Success	
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
205	Reset Content
206	Partial Content
207	Multi-Status (WebDAV)
208	Already Reported (WebDAV)
226	IM Used (HTTP Delta Encoding)
3XX Redirection	
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
306	Unused
307	Temporary Redirect
308	Permanent Redirect
4XX Client Error	
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable

4XX Client (Continue)	
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Payload Too Large
414	URI Too Large
415	Unsupported Media Type
416	Range Not Satisfiable
417	Exception Failed
418	I'm a teapot
421	Misdirected Request
422	Unprocessable Entity (WebDAV)
423	Locked (WebDAV)
424	Failed Dependency (WebDAV)
425	Too Early
426	Upgrade Required
428	Precondition Required
429	Too Many Requests
431	Request Header Fields Too Large
451	Unavailable for Legal Reasons
499	Client Closed Request
5XX Server Error Responses	
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version Not Supported
507	Insufficient Storage (WebDAV)
508	Loop Detected (WebDAV)
510	Not Extended
511	Network Authentication Required
599	Network Connect Timeout Error