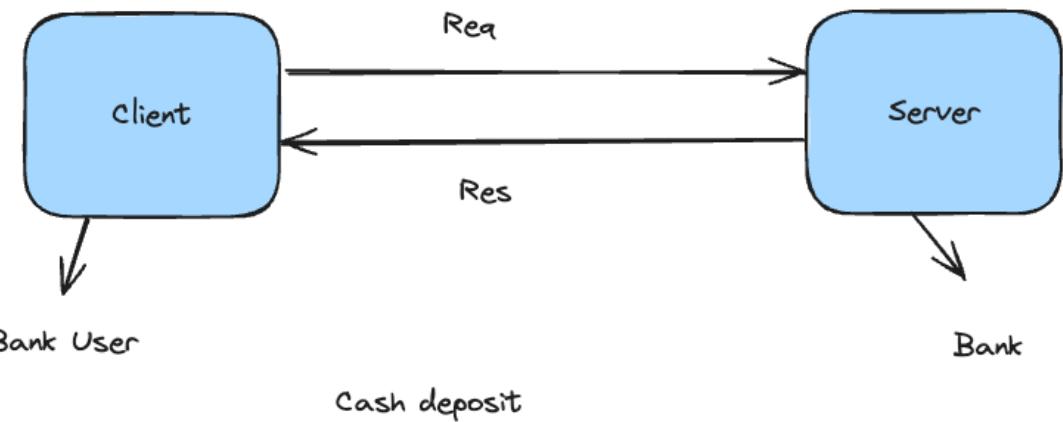
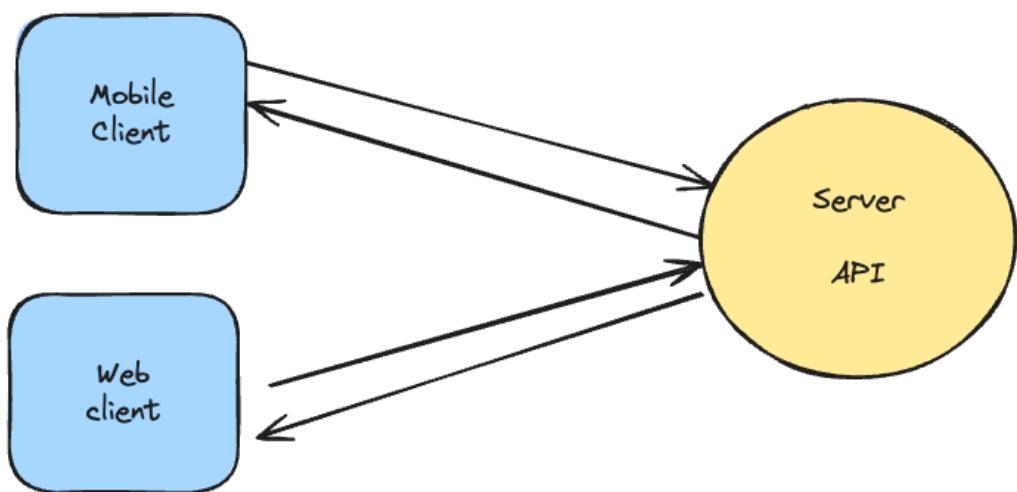


APIs

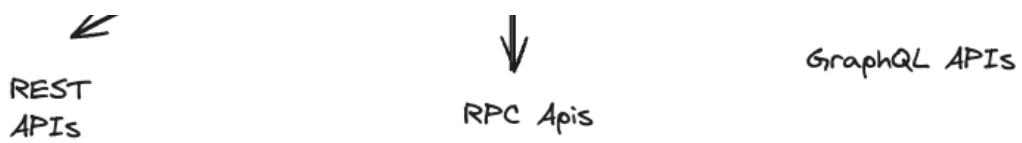


API is nothing but a contract setup by the server which a client needs to follow in order to get a task done.



How to make APIs





1. To send or receive data we use **JSON** (javascript object notation).
2. REST APIs are implemented around **resources**. Resources are the main identity of a REST API. The urls of the rest api depends on the resources

`ipaddr:port/orders`

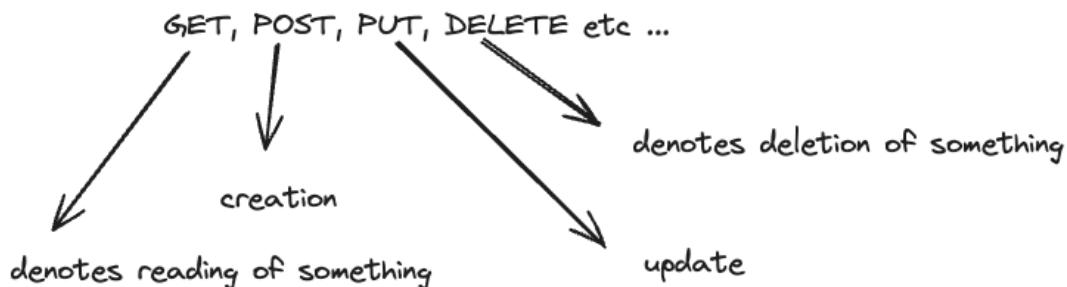
`ipaddr:port/users/2/bookings`

constant

variable: url params

`ipaddr:port/users/:userid/bookings/:bookingid`

3. REST recommends you to use **HTTP protocol** for network interaction. And for defining an API we should use an **HTTP verb**.



Resource	POST	GET	PUT	DELETE
/customers	Create a new customer	Retrieve all customers	Bulk update of customers	Remove all customers
/customers/1	Error	Retrieve the details for customer 1	Update the details of customer 1 if it exists	Remove customer 1
/customers/1/orders	Create a new order for customer 1	Retrieve all orders for customer 1	Bulk update of orders for customer 1	Remove all orders for customer 1

`/customers/13/orders/2`

GET --> we want to fetch order with id 2 which belongs to customer with id 13

POST --> Error

DELETE --> we want to delete order with id 2 that belongs to customer 13

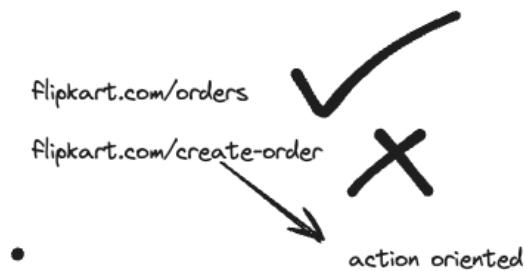
PUT --> we want to update order with id 2 whose customer is 13

- Generally the resources are mentioned in plural form in the urls.

- In the response sent by the server, we should attach a response code (int) which denotes what happened with a request.

- if you want to send big information or sensitive information, send it in req body

- if you want to send some filtering related data then use query params.
flipkart.com/product?min_price=2000&max_price=20000&company=samsung



🌐 https://www.flipkart.com/search?sid=b5g&otracker=CLP_Filters&p%5B%5D=facets.price_range.from%3D20000&p%5B%5D=facets.price_range.to%3D60000&p%5B%5D=facets.rating%255B%255D%3D4%25E2%2598%2585%2B%2526%2Babove

🌐 <https://learn.microsoft.com/en-us/azure/architecture/best-practices/api-design>