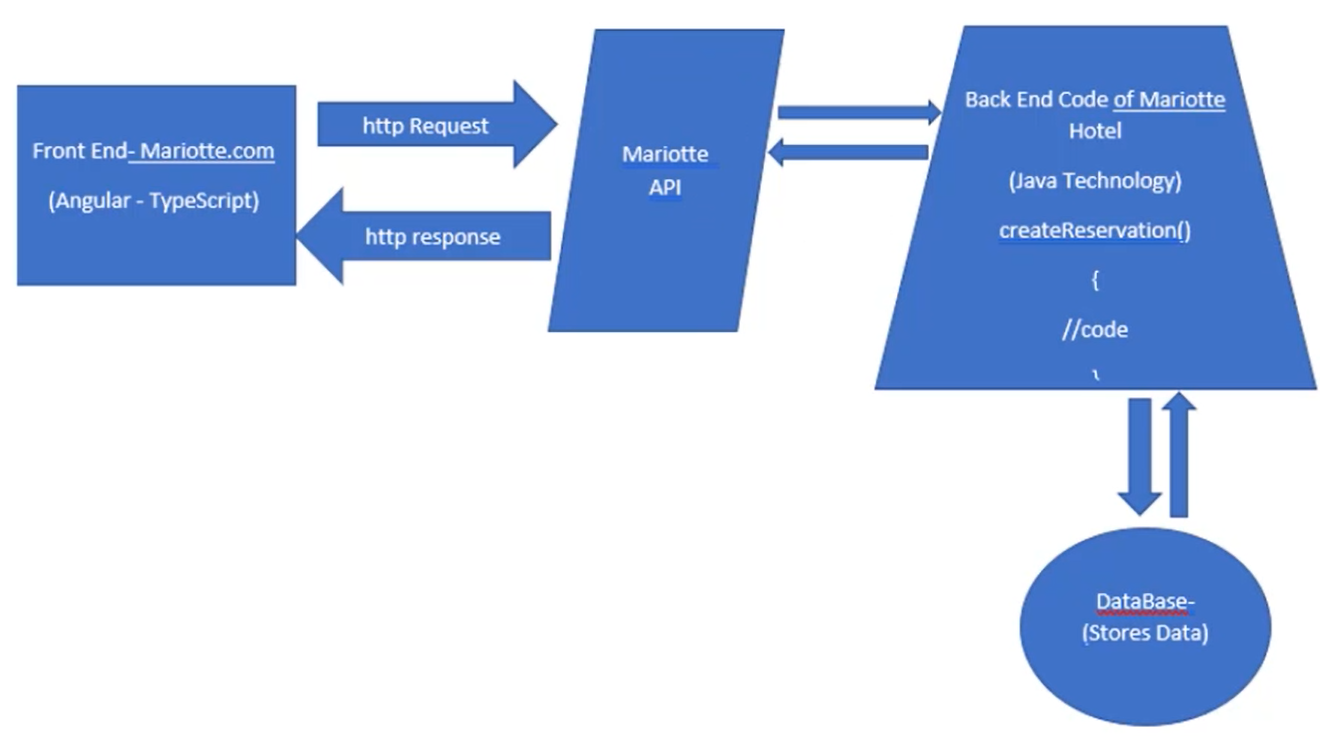
For any website, we have a frontend i.e. UI (we call it as a client). We also have a backend i.e. server and we also use databases. So, if a user wants to book a hotel room in Marriot hotel using Marriot hotel’s own website. He enters details like date and time on UI, these inputs are used by a function / method named as BookA Room() in backend. This method has a code to reserve a room. It books a room and the booking id is stored in database.

When another user wants to book another room in Marriot hotel using makemy trip website, then makemy trip website contacts Marriot hotel’s website (it actually communicates with backend of Marriot’ website where the code to book a room is written. It never communicates with the UI of Marriott’s website. Backend checks in database and tell makemytrip how many rooms left in the hotel now. So, makemytrip retrieves all the info from Marriot’s backend.) But, marriot will not expose their code and database to makemy trip. So, here we use APIs to resolve this problem.

Now, marriot.com will host one API that directly calls BookARoom() method. So, API is between UI and server. Even if someone tries to book a room on Marriot’s own website OR through makemy trip, all the data/info. entered by user on UI, will be passed to API through HTTP protocol(request) using JSON or XML. Once the API gets the details, it will send it to BookARoom() method. And, BookARoom() books a room and store bookingID in DB. After successfully boking a room, the method returns a response to API, and then API uses JSON or XML to send a response back to UI over HTTP protocol(response). Then the UI will parse the JSON / XML response, and show on UI that the reservation is conform and it also shows the info such as booking id, confirmation id etc. While we are sending information, we send it in a format : either JSON or XML.



The most API everybody uses is REST API. Similary we have SOAP APIs / soap webservices. Mostly legacy applications use soap, latest applications use REST APIs.

Rest is one of the architectural design to create API.