

REST

- REPRESENTATIONAL STATE TRANSFER — APPLICATION PROGRAMMING INTERFACE



REST ARCHITECTURAL STYLE

- UNIFORM INTERFACE. (STANDARD)
- RESOURCEORIENTED, EVERYTHING IS A RESOURCE.
- WELL DOCUMENTED ENDPOINTS (SWAGGER ETC)
- CLIENT-SERVER DECOUPLING.
- STATELESSNESS. EACH REQUEST IS TREATED INDEPENDENTLY
- CACHEABILITY.
- LAYERED SYSTEM ARCHITECTURE.
- CODE ON DEMAND

Request

REST API Endpoint URL + API Method + Parameters

<http://dummy.restapiexample.com/api/v1/create>

Response

Client



WebServer



Representation of resource (json, xml, image, etc)

HTTP(S) REQUEST STRUCTURE

- REQUEST METHOD: GET, POST, PUT, DELETE ETC.
- REQUEST URL: THE URL OF THE RESOURCE
- REQUEST HEADER: ACCEPT-LANGUAGE, ACCEPTENCODING, USER-AGENT, HOST ETC.
- REQUEST BODY ==> THIS IS THE DATA TO BE SENT TO THE

REST REQUEST

- VERB (POST, GET, PUT, DELETE ETC.)
- URL (HTTP://DUMMY.RESTAPIEXAMPLE.COM/API/V1/EMPLOYEES)

There are following public apis

#	Route	Method	Type	Full route	Description	Details
1	/employee	GET	JSON	http://dummy.restapiexample.com/api/v1/employees	Get all employee data	Details
2	/employee/{id}	GET	JSON	http://dummy.restapiexample.com/api/v1/employee/1	Get a single employee data	Details
3	/create	POST	JSON	http://dummy.restapiexample.com/api/v1/create	Create new record in database	Details
4	/update/{id}	PUT	JSON	http://dummy.restapiexample.com/api/v1/update/21	Update an employee record	Details
5	/delete/{id}	DELETE	JSON	http://dummy.restapiexample.com/api/v1/update/2	Delete an employee record	Details

HANDS ON

- INSTALL POSTMAN

LINKS

- <http://dummy.restapiexample.com/>
- <https://www.frank-rahn.de/restful-webservices/>