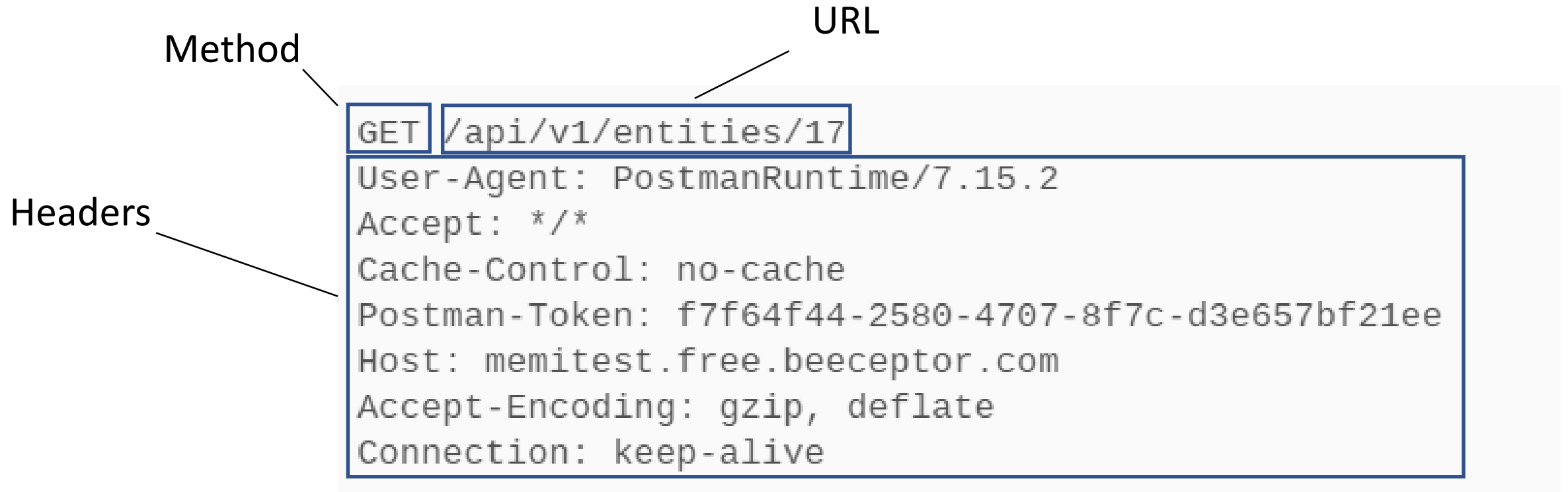


URL Structure

Memilavi
www.memilavi.com



URL Structure



URL Structure

- Defines the structure of the API's URL
- The most important part of the API
- **MUST** be well thought of

URL Structure


- Should be:
 - Self explanatory
 - Consistent across the API
 - Predictable

Or No One Will Use It!

The Domain Name

- The first part of the URL
- Usually contains FQDN (Fully Qualified Domain Name) or a server name

GET
`https://api.ebay.com/sell/account/v1/payments_program/{marketplace_id}/{payments_program_type}`

A blue rectangular box highlights the text 'api.ebay.com' in the URL. A thin black line points from the label 'FQDN' to the top-right corner of this box.

FQDN

The Domain Name

- Protocol should be HTTPS
- For FQDN – should have *api* in its name
- I.e. `https://api.myapp.com`

The API Word

- If the domain does not include *api*, it comes now
- I.e. <https://www.myapp.com/api>
- Emphasizes this URL is for the API, not the website
- Can be part of the domain name itself:

```
"https://language.googleapis.com/v1/documents:analyzeEntities"
```

Version

- If the version strategy is URL – it comes now
- Only natural number
 - Positive
 - No decimals
- Could be prefixed with `v`
- I.e. `https://www.myapp.com/api/v1`

Entity

- REST deals with resources, or entities
- The next part is the entity we're dealing with
- Should be one word, no more
- NEVER a verb
 - This is the HTTP Verb role

Entity Examples

- `/api/v1/order`
- `/api/v2/employees`
- `/api/v2/travel`

ID Parameter

- When dealing with a specific entity – next is the entity ID
- The ID is part of the URL
- `/api/v1/order/17`

ID Parameter

- Relevant for:
 - GET
 - PUT
 - DELETE
- Not POST – No ID for new entity

Sub Entity

- Used for accessing entities that are dependent on other entities
- Example: Get all the items of order no. 17

Sub Entity

- Common mistake:
 - `/api/v1/ItemsInOrder/17`
- Not readable
- Not simple

Sub Entity

- Sub Entity should come after the ID parameter
 - `/api/v1/order/17/items`
- URL is Hierarchical, easy to understand
- Same goes for sub-sub-entity, etc.

Query Parameters

- Used to query the entities in GET method
- Come at the end of the URL, after ?
- Concatenated with &
 - `/api/v1/orders?fromDate=12/12/2018&toDate=2/2/2019`

Query Parameters vs ID Parameter

	Query Parameters	ID Parameter
Location		
Param count		
Example		
Return value		
If not found		

Query Parameters vs ID Parameter

	Query Parameters	ID Parameter
Location	At the end of the URL	End or middle of URL
Param count		
Example		
Return value		
If not found		

Query Parameters vs ID Parameter

	Query Parameters	ID Parameter
Location	At the end of the URL	End or middle of URL
Param count	0..N parameters	1 parameter
Example		
Return value		
If not found		

Query Parameters vs ID Parameter

	Query Parameters	ID Parameter
Location	At the end of the URL	End or middle of URL
Param count	0..N parameters	1 parameter
Example	<code>/api/v1/orders?user=john</code>	<code>/api/v1/order/17</code>
Return value		
If not found		

Query Parameters vs ID Parameter

	Query Parameters	ID Parameter
Location	At the end of the URL	End or middle of URL
Param count	0..N parameters	1 parameter
Example	/api/v1/orders? user=john	/api/v1/order/ 17
Return value	May return 0...N entities	Must return exactly 1 entity
If not found		

Query Parameters vs ID Parameter

	Query Parameters	ID Parameter
Location	At the end of the URL	End or middle of URL
Param count	0..N parameters	1 parameter
Example	/api/v1/orders? user=john	/api/v1/order/ 17
Return value	May return 0...N entities	Must return exactly 1 entity
If not found	That's fine...	Error!


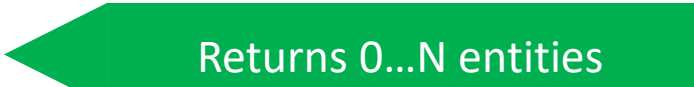
Singular vs Plural

- The dilemma:
 - `/api/v1/order/17`
 - Or...
 - `/api/v1/orders/17`

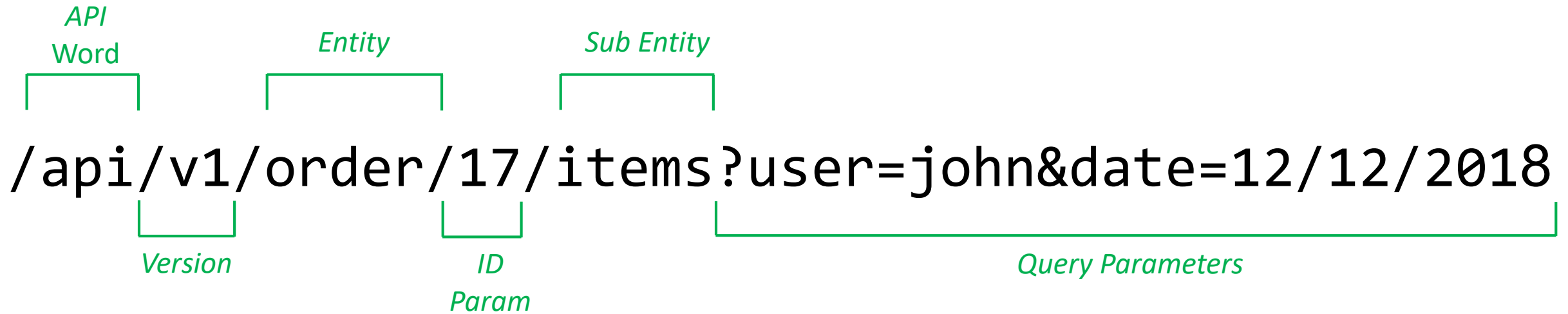
Singular vs Plural

- The dilemma #2:
 - `/api/v1/order?user=john`
 - Or...
 - `/api/v1/orders?user=john`

Singular vs Plural

- No concrete answer
- Best practice: Always prefer readability and ease of use
- `/api/v1/order/17`  Returns single entity
- `/api/v1/orders?user=john`  Returns 0...N entities

Summary



Remember: Some components are optional