

Parameter Examples:

Here is an example of a parameter value:

```
1. parameters:
2.   - in: query
3.     name: status
4.     schema:
5.       type: string
6.       enum: [approved, pending, closed, new]
7.       example: approved    # Example of a parameter value
```

Multiple examples for a parameter:

```
1. parameters:
2.   - in: query
3.     name: limit
4.     schema:
5.       type: integer
6.       maximum: 50
7.       examples:    # Multiple examples
8.         zero:      # Distinct name
9.           value: 0  # Example value
10.        summary: A sample limit value # Optional description
11.        max: # Distinct name
12.        value: 50  # Example value
13.        summary: A sample limit value # Optional description
```

As you can see, each example has a distinct key name. Also, in the code above, we used an optional **summary** keys with description. **Note:** the sample values you specify should match the parameter data type.

Request and Response Body Examples:

Here is an example of the **example** keyword in a request body:

```
1. paths:
2.   /users:
3.     post:
```

```

4.    summary: Adds a new user
5.    requestBody:
6.      content:
7.        application/json:
8.          schema: # Request body contents
9.            type: object
10.           properties:
11.             id:
12.               type: integer
13.             name:
14.               type: string
15.           example: # Sample object
16.             id: 10
17.             name: Jessica Smith
18.   responses:
19.     '200':
20.       description: OK

```

Note that in the code above, **example** is a child of **schema**. If **schema** refers to some object defined in the **components** section, then you should make **example** a child of the media type keyword:

```

1. paths:
2.   /users:
3.     post:
4.       summary: Adds a new user
5.       requestBody:
6.         content:
7.           application/json: # Media type
8.             schema: # Request body contents
9.               $ref: '#/components/schemas/User' # Reference to an object
10.            example: # Child of media type because we use $ref above
11.              # Properties of a referenced object
12.              id: 10
13.              name: Jessica Smith
14.     responses:
15.       '200':
16.         description: OK

```

This is needed because `$ref` overwrites all the siblings alongside it. If needed, you can use multiple examples:

```
1. paths:
2.   /users:
3.     post:
4.       summary: Adds a new user
5.       requestBody:
6.         content:
7.           application/json: # Media type
8.             schema: # Request body contents
9.               $ref: '#/components/schemas/User' # Reference to an object
10.            examples: # Child of media type
11.              Jessica: # Example 1
12.                value:
13.                  id: 10
14.                  name: Jessica Smith
15.              Ron: # Example 2
16.                value:
17.                  id: 11
18.                  name: Ron Stewart
19.            responses:
20.              '200':
21.                description: OK
```

Here is an example of the `example` in response bodies:

```
1. responses:
2.   '200':
3.     description: A user object.
4.     content:
5.       application/json:
6.         schema:
7.           $ref: '#/components/schemas/User' # Reference to an object
8.         example:
9.           # Properties of the referenced object
10.          id: 10
11.          name: Jessica Smith
```

Multiple examples in response bodies:

```
1. responses:
2.   '200':
3.     description: A user object.
4.     content:
5.       application/json:
6.         schema:
7.           $ref: '#/components/schemas/User' # Reference to an object
8.         examples:
9.           Jessica:
10.            value:
11.              id: 10
12.              name: Jessica Smith
13.           Ron:
14.            value:
15.              id: 20
16.              name: Ron Stewart
```

Note: The examples in response and request bodies are free-form, but are expected to be compatible with the body schema.

Object and Property Examples:

You can also specify examples for objects and individual properties in the **components** section.

- Property-level example:

```
• components:
•   schemas:
•     User: # Schema name
•     type: object
•     properties:
•       id:
•         type: integer
•         format: int64
•         example: 1 # Property example
•     name:
•       type: string
```

- example: New order # Property example
- Object-level example:

```
• components:
• schemas:
• User: # Schema name
• type: object
• properties:
• id:
• type: integer
• name:
• type: string
• example: # Object-level example
• id: 1
• name: Jessica Smith
```

Note that schemas and properties support single **example** but not multiple **examples**.

Array Example:

You can add an example of an individual array item:

```
1. components:
2. schemas:
3. ArrayOfInt:
4. type: array
5. items:
6. type: integer
7. format: int64
8. example: 1
```

or an array-level example containing multiple items:

```
1. components:
2. schemas:
3. ArrayOfInt:
4. type: array
5. items:
6. type: integer
7. format: int64
```

```
8.   example: [1, 2, 3]
```

If the array contains objects, you can specify a multi-item example as follows:

```
1. components:
2.   schemas:
3.     ArrayOfUsers:
4.       type: array
5.       items:
6.         type: object
7.         properties:
8.           id:
9.             type: integer
10.          name:
11.            type: string
12.        example:
13.          - id: 10
14.            name: Jessica Smith
15.          - id: 20
16.            name: Ron Stewart
```

Note that arrays and array items support single **example** but not multiple **examples**.

Examples for XML and HTML Data

To describe an example value that cannot be presented in JSON or YAML format, specify it as a string:

```
1. content:
2.   application/xml:
3.     schema:
4.       $ref: '#/components/schemas/xml'
5.     examples:
6.       xml:
7.         summary: A sample XML response
8.         value:
9.           '<objects><object><id>1</id><name>new</name></object><object><id>2</id></object></objects>'
```

```

10. schema:
11.   type: string
12. examples:
13.   html:
14.     summary: A list containing two items
15.     value: '<html><body><ul><li>item 1</li><li>item
              2</li></ul></body></html>'

```

You can find information on writing multiline string in YAML in this Stack Overflow post: <https://stackoverflow.com/questions/3790454/in-yaml-how-do-i-break-a-string-over-multiple-lines>.

External Examples:

If a sample value cannot be inserted into your specification for some reason, for instance, it is neither YAML-, nor JSON-conformant, you can use the `externalValue` keyword to specify the URL of the example value. The URL should point to the resource that contains the literal example contents (an object, file or image, for example):

```

1. content:
2.   application/json:
3.     schema:
4.       $ref: '#/components/schemas/MyObject'
5.     examples:
6.       jsonObject:
7.         summary: A sample object
8.         externalValue: 'http://example.com/examples/object-example.json'
9.   application/pdf:
10.    schema:
11.      type: string
12.      format: binary
13.    examples:
14.      sampleFile:
15.        summary: A sample file
16.        externalValue: 'http://example.com/examples/example.pdf'

```

Reusing Examples:

You can define common examples in the [components/examples](#) section of your specification and then re-use them in various parameter descriptions, request and response body descriptions, objects and properties:

```
1. content:
2.   application/json:
3.     schema:
4.       $ref: '#/components/schemas/MyObject'
5.     examples:
6.       objectExample:
7.         $ref: '#/components/examples/objectExample'
8.   ...
9. components:
10.  examples:
11.    objectExample:
12.      value:
13.        id: 1
14.        name: new object
15.      summary: A sample object
```