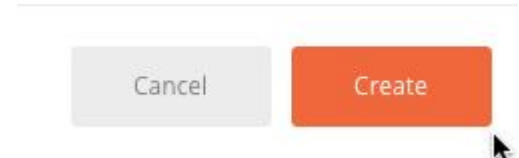
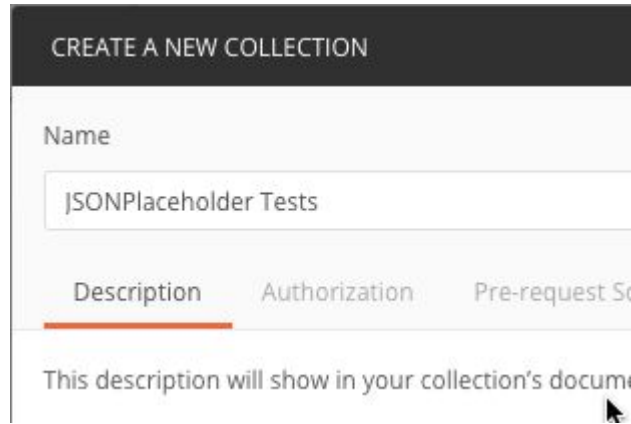
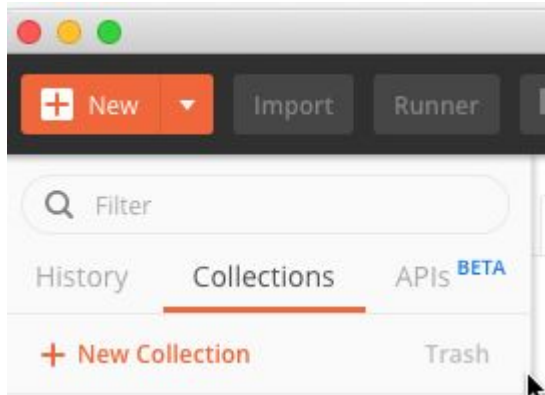


Exercise - testing in Postman

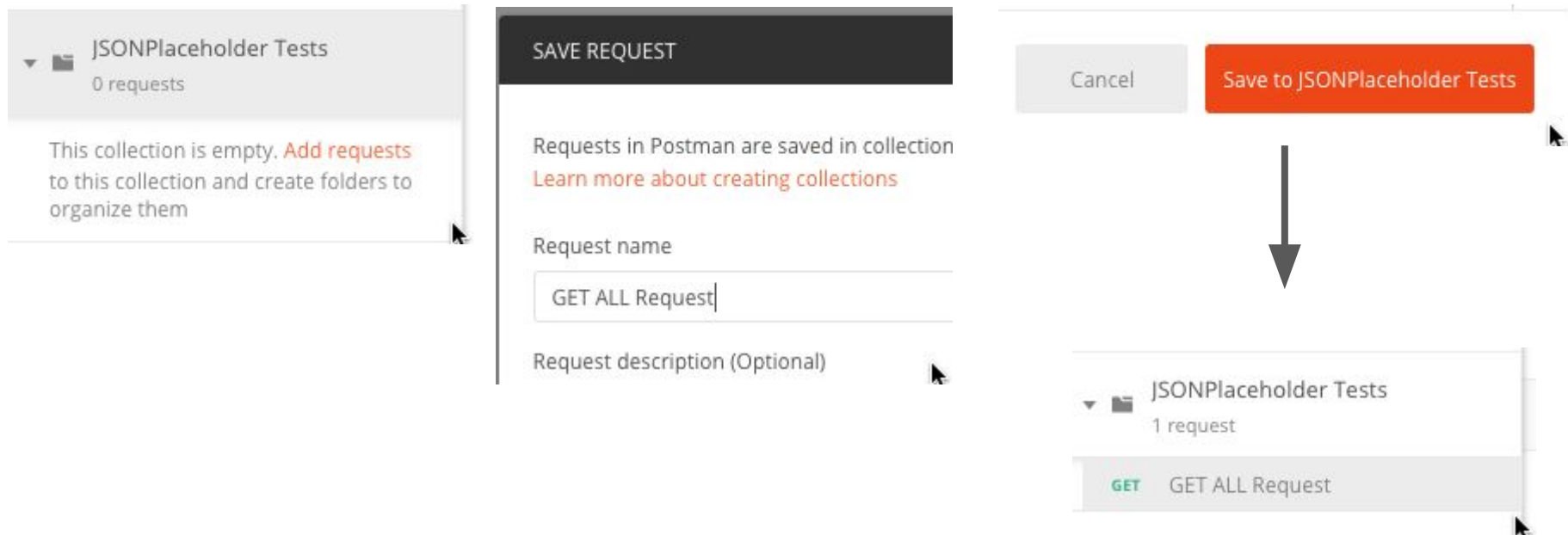
Postman Setup

1. Make a new collection



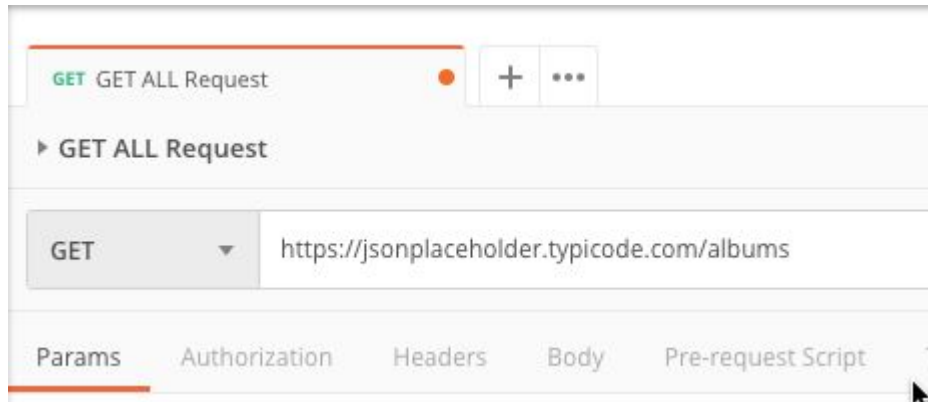
Postman Setup

2. Add a new request



Postman Setup - part 3

3. Put the URL you want to test into the box



Test Cases - Review

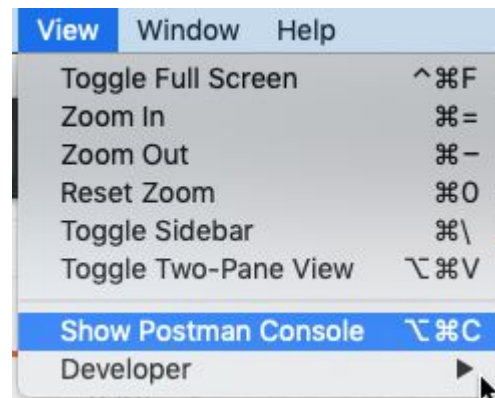
Test cases are written in Javascript

Use the built in Postman variables:

- `responseCode` → dictionary that gives you info about the status code
- `responseBody` → the actual data that comes from the API

Debug by using `console.log`

- View > Show Postman Console



Test case for GET ALL - Steps

// 1. test that response code = 200

// 2. test that the response data has 100 items

// 3. test that a single dictionary has these keys:

// userId

// id

// title

Implement step 1

```
console.log("I AM TESTING!");
```

```
console.log("Response Code: ");
```

```
console.log("=====");
```

```
console.log(responseCode);
```

```
console.log("Response Body: ");
```

```
console.log("=====");
```

```
console.log(responseBody);
```

What's in responseCode?

```
▼ Object: {}  
  code: 200  
  detail: "Standard response fo  
    d. In a GET request, the re  
    est the response will conta  
  name: "OK"  
  standardName: "OK"
```


Test case for status code

```
// 1. test that response code = 200
```

```
let actualStatusCode = responseCode["code"]
```

```
tests["Does status code == 200?"] = actualStatusCode === 200
```



Test for num items in response

```
// 2. test that the response data has 100 items
// 2a. convert the response data to a JSON Object type
let dataFromServer = JSON.parse(responseBody)

// 2b. get the number of items in the object
let actualNumItems = dataFromServer.length
tests["Response contains 100 items"] = actualNumItems === 100
```

Test that album has required keys

```
// 3a. Get one item of the response
```

```
let album = dataFromServer[0]
```

```
// 3b. Check if the albumn has the userId key
```

```
let containsUserIdKey = "userId" in album
```

```
tests["Album has a userId key"] = containsUserIdKey === true
```

```
// 3c. Check if the albumn has the id key
```

```
tests["Album has a id key"] = "id" in album
```

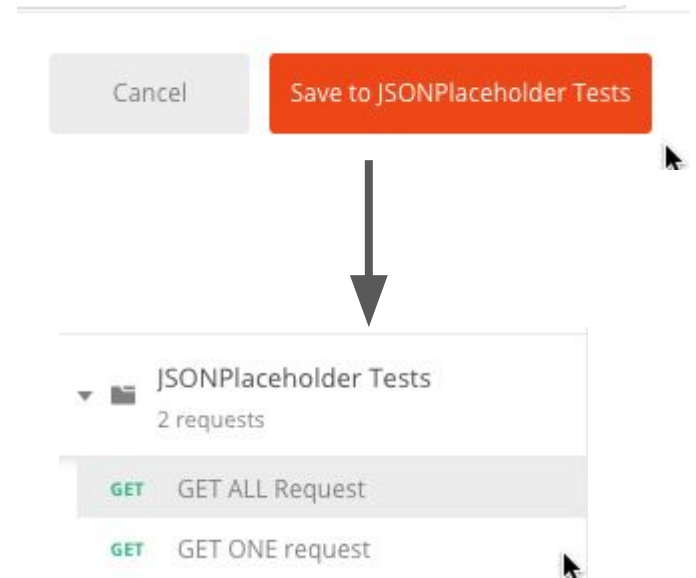
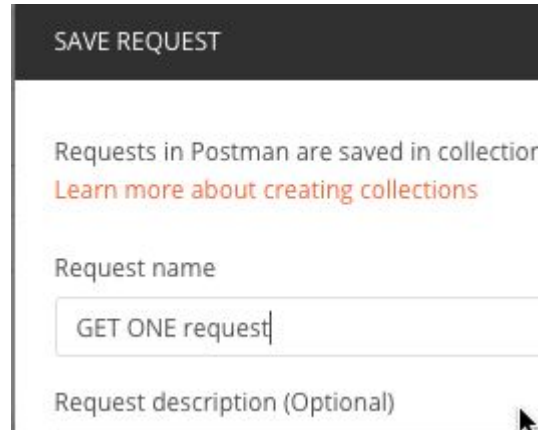
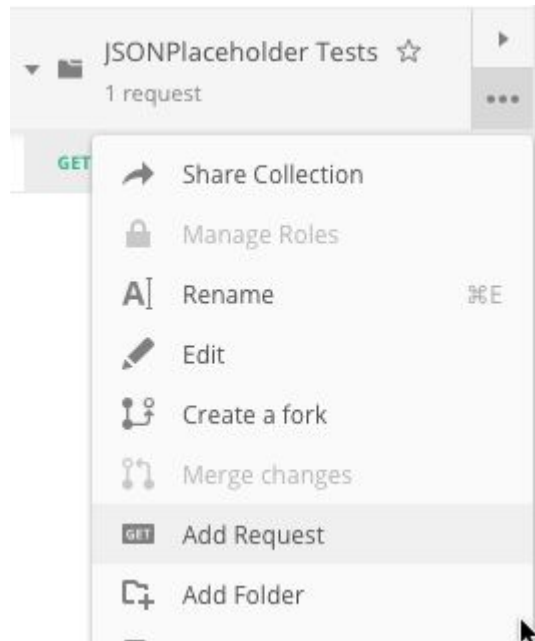
```
// 3d. Check if the albumn has the title key
```

```
tests["Album has a title key"] = "title" in album
```


GET ONE Request

Setup - add new request

1. Add a new request to your collection



Do same process as GET ALL request

Test POST erquest

Setup

1. Add teh POSt request to yoru collection

