

This assignment will involve some research on your own on how to use the Trello API to get things working.

Get familiar with Trello's REST API documentation (<https://developer.atlassian.com/cloud/trello/rest/>) to solve this assignment.

IMPORTANT: you will need an [account with trello.com](#), not with [developers.trello.com](#)

If you don't have an account, [create one](#) before starting this assignment.

Vooraleer ik vraag 1 ga uitvoeren moet ik eerst een nieuw bord aanmaken en de ID ervan meegeven in de global variabele

Trello / Create board

POST ▼ <https://api.trello.com/1/boards/?name=My new board&key={{keyapi}}&token={{tokentrello}}&defaultLists=false>

Params ● Authorization Headers (7) Body Pre-request Script **Tests ●** Settings

```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
4 pm.test("Board should be created", function () {
5   var jsonData = pm.response.json();
6   pm.expect(jsonData.name).to.eql("My new board");
7   pm.expect(jsonData.closed).to.eql(false);
8 });
9 pm.test("Board should be private", function () {
10  var jsonData = pm.response.json();
11  pm.expect(jsonData.prefs.permissionLevel).to.eql("private");
12 });
```

Body Cookies (2) Headers (43) Test Results (3/3)

Pretty Raw Preview Visualize JSON ▼ ≡

```
1 {
2   "id": "616430f0155a5407c508d8fb",
3   "name": "My new board",
4   "desc": "",
5   "descData": null,
6   "closed": false,
7   "idOrganization": "616424aa0bc8d00f17dbc407",
8   "idEnterprise": null,
9   "pinned": false,
10  "url": "https://trello.com/b/U7JZe0XM/my-new-board",
11  "shortUrl": "https://trello.com/b/U7JZe0XM",
12  "prefs": {
13    "permissionLevel": "private",
14  }
```

De eerste test was om te zien dat het bord is aangemaakt. Daarna geef ik het boardID als globale variabele. We zien dat de test results 3/3 is → succesvol uitgevoerd.

Globals		
Global variables for a workspace are a set of variables that are always available within the scope of that workspace. They can be viewed and edited by anyone in that workspace. Learn more about globals ➤		
	VARIABLE	CURRENT VALUE ⓘ
<input checked="" type="checkbox"/>	boardID	616430f0155a5407c508d8fb
<input checked="" type="checkbox"/>	keyapi	1c0b21980c6841c9e0e69936ece307c6
<input checked="" type="checkbox"/>	tokentrello	3fce7efe427ae299e909070c98d0d9779f7d93ab0389a3535d3a270dd6b05d69
	Add a new variable	

Variabele is nu opgeslagen.

1. Using trello's online API documentation (<https://developer.atlassian.com/cloud/trello/rest/>) **build another request which creates a new list called TODO** inside your already created board.

Write tests to check that the request performed as expected. As a minimum, please test the following:

- status is 200
- the name of the list is TODO
- should not be closed

Feel free to write any additional tests that you feel are needed.

Please share your collection or tests.

```
pm.test("Status code is 200", function () {
  pm.response.to.have.status(200);
});
pm.test("Check list name", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.name).to.eql("TODO");
});
pm.test("Check list is not closed", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.closed).to.eql(false);
});
let id = pm.response.json().id;
pm.globals.set("list_TODO_id", id);
```

POST ▼ `https://api.trello.com/1/lists?name=TODO&idBoard={{boardID}}&key={{keyapi}}&token={{tokentrello}}`

Params ● Authorization Headers (7) Body Pre-request Script **Tests ●** Settings

```
1 pm.test("Status code is 200", function () {
2   | pm.response.to.have.status(200);
3   | });
4 pm.test("Check list name", function () {
5   |   var jsonData = pm.response.json();
6   |   pm.expect(jsonData.name).to.eql("TODO");
7   | });
8 pm.test("Check list is not closed", function () {
9   |   var jsonData = pm.response.json();
10  |   pm.expect(jsonData.closed).to.eql(false);
11  | });
12 let id = pm.response.json().id;
13 pm.globals.set("list_TODO_id", id);
14
```

Body Cookies (2) Headers (42) **Test Results (3/3)**

All Passed Skipped Failed

PASS

 Status code is 200

PASS

 Check list name

PASS

 Check list is not closed

- ➔ Test succesvol
- ➔ ID van TODO-list opgeslagen in glob variabele

2. Additionally **create a new list** called **DONE** inside your already created board (in a similar way you have created the previous one).

Write tests to check that the request performed as expected. As a minimum, please test the following:

- status is 200
- the name of the list is DONE
- should not be closed

Optional:

- test that the list was created in the desired board

Please share your collection or tests.

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
pm.test("Check list name", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.name).to.eql("DONE");
});
pm.test("Check list is not closed", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.closed).to.eql(false);
});
pm.test("List is created in API_test board", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.idBoard).to.eql
    (pm.globals.get("board_id"));
});
let id = pm.response.json().id;
pm.globals.set("list_DONE_id", id);
```

POST

https://api.trello.com/1/lists?name=DONE&idBoard={{boardID}}&key={{keyapi}}&token={{tokentrello}}

Params ● Authorization Headers (7) Body Pre-request Script Tests ● Settings

```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
4 pm.test("Check list name", function () {
5   var jsonData = pm.response.json();
6   pm.expect(jsonData.name).to.eql("DONE");
7 });
8 pm.test("Check list is not closed", function () {
9   var jsonData = pm.response.json();
10  pm.expect(jsonData.closed).to.eql(false);
11 });
12 pm.test("List is created in API_test board", function () {
13   var jsonData = pm.response.json();
14   pm.expect(jsonData.idBoard).to.eql
15     (pm.globals.get("boardID"));
16 });
17 let id = pm.response.json().id;
18 pm.globals.set("list_DONE_id", id);
19
```

Body Cookies (2) Headers (42) Test Results (4/4)

All Passed Skipped Failed

PASS

Status code is 200

PASS

Check list name

PASS

Check list is not closed

PASS

List is created in API_test board

Test was succesvol, de id van de DONE-list is opgeslagen in globale vars.

3. Inside the TODO list, **create a new card** (which is like a task) with the name ***Learn Postman***.

Write tests to check that the request performed as expected.

As a **minimum**, please test the following:

- status is 200
- the name of the card is Learn Postman
- should not be closed

Optional:

- test that the card was created in the desired TODO list
- test that the card was created in the desired board
- test that the card has no votes or attachments

Please share your collection or tests.

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
pm.test("Check card name", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.name).to.eql("Learn Postman");
});
pm.test("Check card is not closed", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.closed).to.eql(false);
});
pm.test("card is created in TODO list", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.idList).to.eql
    (pm.globals.get("list_TODO_id"));
});
let id = pm.response.json().id;
pm.globals.set("card_learnpostman_id", id);
```

POST ▼ https://api.trello.com/1/cards?name=Learn Postman&idList={{list_TODO_id}}&key={{keyapi}}&token={{tokentrello}}

Params ● Authorization Headers (7) Body Pre-request Script **Tests ●** Settings

```

1  pm.test("Status code is 200", function () {
2    pm.response.to.have.status(200);
3  });
4  pm.test("Check card name", function () {
5    var jsonData = pm.response.json();
6    pm.expect(jsonData.name).to.eql("Learn Postman");
7  });
8  pm.test("Check card is not closed", function () {
9    var jsonData = pm.response.json();
10   pm.expect(jsonData.closed).to.eql(false);
11 });
12 pm.test("card is created in TODO list", function () {
13   var jsonData = pm.response.json();
14   pm.expect(jsonData.idList).to.eql
15     (pm.globals.get("list_TODO_id"));
16 });
17 let id = pm.response.json().id;
18 pm.globals.set("card_learnpostman_id", id);

```

Body Cookies (2) Headers (42) **Test Results (4/4)**

All Passed Skipped Failed

- PASS Status code is 200
- PASS Check card name
- PASS Check card is not closed
- PASS card is created in TODO list

Test was succesvol, de id van de card die net gemaakt is werd opnieuw in de globale variabele geplaatst.

Globals

Global variables for a workspace are a set of variables that are always available within the scope of that workspace. They can be viewed and edited by anyone in that workspace. [Learn more about globals](#) ↗

	VARIABLE	INITIAL VALUE ①	CURRENT VALUE ①
<input checked="" type="checkbox"/>	boardId		616430f0155a5407c508d8fb
<input checked="" type="checkbox"/>	keyapi		1c0b21980c6841c9e0e69936ece307c6
<input checked="" type="checkbox"/>	tokentrello		3fce7efe427ae299e909070c98d0d9779f7d93ab0389a3535d3a270dd6b05d69
<input checked="" type="checkbox"/>	list_TODO_id		6164330076774953822fe50f
<input checked="" type="checkbox"/>	list_DONE_id		616434894382f789df566f74
<input checked="" type="checkbox"/>	card_learnpostman_id		6164361f789c6a03ff863db5
	Add a new variable		

Dit zijn alle globale variabele die ik tot nu toe heb opgeslagen.

4. **Move the card** from the TODO list to the DONE list.

Write tests to check that the request performed as expected.

Please test the following:

- status is 200
- the name of the card is still Learn Postman
- test that the card was moved to the desired DONE list

Hint: You will be using the PUT method to update this information.

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
pm.test("Check card name", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.name).to.eql("Learn Postman");
});
pm.test("Check card is not closed", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.closed).to.eql(false);
});
pm.test("card is in DONE list", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.idList).to.eql
    (pm.globals.get("list_DONE_id"));
});
let id = pm.response.json().id;
pm.globals.set("card_learnpostman_id", id);
```


PUT

https://api.trello.com/1/cards/:id?idList={{list_DONE_id}}&key={{keyapi}}&token={{tokentrello}}

Params ● Authorization Headers (7) Body Pre-request Script **Tests ●** Settings

```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
4 pm.test("Check card name", function () {
5   var jsonData = pm.response.json();
6   pm.expect(jsonData.name).to.eql("Learn Postman");
7 });
8 pm.test("Check card is not closed", function () {
9   var jsonData = pm.response.json();
10  pm.expect(jsonData.closed).to.eql(false);
11 });
12 pm.test("card is in DONE list", function () {
13   var jsonData = pm.response.json();
14   pm.expect(jsonData.idList).to.eql
15     (pm.globals.get("list_DONE_id"));
16 });
17 let id = pm.response.json().id;
18 pm.globals.set("card_learnpostman_id", id);
```

Body Cookies (2) Headers (42) **Test Results (4/4)**

All Passed Skipped Failed

PASS

Status code is 200

PASS

Check card name

PASS

Check card is not closed

PASS

card is in DONE list

Alle testen zijn succesvol.

000 Tableau ▾ My new board ☆ Espace de travail de Efrem Gokinan Privé EG Inviter

TODO ...
+ Ajouter une carte

DONE ...
Learn Postman
+ Ajouter une carte

+ Ajoutez une autre liste

➔ Resultaat van de 4 testen

5. It is time to clean things up. As you do not need the task, the lists and the board, make sure you **delete the board**, so that your Trello account does not get filled with useless boards.

Please test the following:

- status is 200

Optional: figure a way on how you could test that the board does not exist anymore (hint: using a GET request)

Hint: You will be using the **DELETE** method in order to **delete the board**.

```
pm.test("Status code is 200", function () {  
    pm.response.to.have.status(200);  
});  
let id = pm.globals.get("board_id");
```

The screenshot shows the Postman interface for a DELETE request. The top bar indicates the method is 'DELETE' and the URL is 'https://api.trello.com/1/boards/:id?key={{keyapi}}&token={{tokentrello}}'. Below the URL bar, there are tabs for 'Params', 'Authorization', 'Headers (6)', 'Body', 'Pre-request Script', 'Tests', and 'Settings'. The 'Tests' tab is selected, showing the same JavaScript test code as in the previous block. Below the test code, there are tabs for 'Body', 'Cookies (2)', 'Headers (42)', and 'Test Results (1/1)'. The 'Test Results' tab is selected, showing a 'PASS' status and the message 'Status code is 200'.

Test was opnieuw succesvol. Alle testen van de opdracht zijn nu uitgevoerd en alles werkt.

<https://www.getpostman.com/collections/6e045229e79b8175e4eb>

➔ Link naar mijn collectie van testen

6. How was this assignment? How much time did you need to finish it?

Leuke taak, ik moest alles zelf gaan zoeken welke testen ik wou uitvoeren. 2uur