



# Pagination.js 2.1.5

A jQuery plugin to provide simple yet fully customisable pagination.  
Almost all the ways you can think of on pagination.

Pagination.js

Pagination.min.js 16.0K

## Normal

1

2

3

4

5

6

7

8

9

10

«

1

2

3

4

5

...

20

»

```
1. $('#demo').pagination({
2.   dataSource: [1, 2, 3, 4, 5, 6, 7, ... , 195],
3.   callback: function(data, pagination) {
4.     // template method of yourself
5.     var html = template(data);
6.     dataContainer.html(html);
7.   }
8. })
```

## Only page numbers

1

2

3

4

5

1

2

3

4

5

...

20

```
1. $('#demo').pagination({
2.   dataSource: [1, 2, 3, 4, 5, 6, 7, ... , 100],
3.   pageSize: 5,
4.   showPrevious: false,
5.   showNext: false,
6.   callback: function(data, pagination) {
7.     // template method of yourself
8.     var html = template(data);
9.     dataContainer.html(html);
10.  }
11. })
```

## Show "go" input & button

1

2

3

4

Top

5

«

1

2

3

4

5

...

8

»

Go

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 40 ],
3.   pageSize: 5,
4.   showGoInput: true,
5.   showGoButton: true,
6.   callback: function( data, pagination ) {
7.     // template method of yourself
8.     var html = template( data );
9.     dataContainer.html( html );
10.  }
11. })
```

## Auto hide previous & next button

1

2

3

4

5

1

2

3

4

5

6

7

»

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 35 ],
3.   pageSize: 5,
4.   autoHidePrevious: true,
5.   autoHideNext: true,
6.   callback: function( data, pagination ) {
7.     // template method of yourself
8.     var html = template( data );
9.     dataContainer.html( html );
10.  }
11. })
```

## Mini

1

2

3

4

5

«

»

1 / 10

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 50 ],
3.   pageSize: 5,
4.   showPageNumbers: false,
5.   showNavigator: true,
6.   callback: function( data, pagination ) {
7.     // template method of yourself
8.     var html = template( data );
9.     dataContainer.html( html );
10.  }
11. })
```

## Show all pages

1

2

3

4

5

«

1

2

3

4

5

6

7

8

9

10

»

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 50 ],
3.   pageSize: 5,
4.   pageRange: null,
5.   showPageNumbers: true,
6.   callback: function( data, pagination ) {
7.     // template method of yourself
8.     var html = template( data );
9.     dataContainer.html( html );
10.  }
11. })
```

## Asynchronous or JSONP

- [DSC\\_0445](#)
- [DSC\\_0354](#)
- [DSC\\_0377](#)
- [DSC\\_0605](#)
- [DSC\\_0305](#)



[DSC\\_0586](#)  
[DSC\\_0192](#)  
[DSC\\_0175](#)

«	1	2	3	4	5	6	»
---	---	---	---	---	---	---	---

```
1. $('#demo').pagination({
2.   dataSource: 'https://api.flickr.com/services/feeds/photos_
3.   locator: 'items',
4.   totalNumber: 120,
5.   pageSize: 20,
6.   ajax: {
7.     beforeSend: function() {
8.       dataContainer.html('Loading data from flickr.com .
9.     }
10.  },
11.  callback: function(data, pagination) {
12.    // template method of yourself
13.    var html = template(data);
14.    dataContainer.html(html);
15.  }
16. })
```

## Asynchronous & Dynamic total number

[DSC\\_0445](#)  
[DSC\\_0354](#)  
[DSC\\_0377](#)  
[DSC\\_0605](#)  
[DSC\\_0305](#)  
[DSC\\_0586](#)  
[DSC\\_0192](#)  
[DSC\\_0175](#)

«	1	2	3	4	5	...	15	»
---	---	---	---	---	---	-----	----	---

```
1. $('#demo').pagination({
2.   dataSource: 'https://api.flickr.com/services/feeds/photos_
3.   locator: 'items',
4.   totalNumberLocator: function(response) {
5.     // you can return totalNumber by analyzing response cor
6.     return Math.floor(Math.random() * (1000 - 100)) + 100;
7.   },
8.   pageSize: 20,
9.   ajax: {
10.    beforeSend: function() {
11.      dataContainer.html('Loading data from flickr.com .
12.    }
13.  },
14.  callback: function(data, pagination) {
15.    // template method of yourself
16.    var html = template(data);
17.    dataContainer.html(html);
18.  }
19. })
```

## Specify default

11

12

13

14

15

«	1	2	3	4	5	6	7	»
---	---	---	---	---	---	---	---	---

```
1. $('#demo').pagination({
2.   dataSource: [1, 2, 3, 4, 5, 6, 7, ... , 35],
3.   pageSize: 5,
4.   pageNumber: 3,
5.   callback: function(data, pagination) {
6.     // template method of yourself
7.     var html = template(data);
8.     dataContainer.html(html);
9.   }
10. })
```

## Format result data

- 1 - good guys
- 2 - good guys
- 3 - good guys
- 4 - good guys
- 5 - good guys
- 6 - good guys

7 - good guys

8 - good guys

«	1	2	3	4	5	...	13	»
---	---	---	---	---	---	-----	----	---

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 100],
3.   pageSize: 8,
4.   formatResult: function(data) {
5.     var result = [];
6.     for (var i = 0, len = data.length; i < len; i++) {
7.       result.push(data[i] + ' - good guys');
8.     }
9.     return result;
10.  },
11.  callback: function(data, pagination) {
12.    // template method of yourself
13.    var html = template(data);
14.    dataContainer.html(html);
15.  }
16. })
```

## Another format result data

1 - bad guys

2 - bad guys

3 - bad guys

4 - bad guys

5 - bad guys

«	1	2	3	4	5	...	20	»
---	---	---	---	---	---	-----	----	---

```
1. $( '#demo' ).pagination({
2.   dataSource: [{a :1}, {a :2}, {a :3}, {a :4}, ... , {a :50}],
3.   pageSize: 8,
4.   formatResult: function(data) {
5.     for (var i = 0, len = data.length; i < len; i++) {
6.       data[i].a = data[i].a + ' - bad guys';
7.     }
8.   },
9.   callback: function(data, pagination) {
10.    // template method of yourself
11.    var html = template(data);
12.    dataContainer.html(html);
13.  }
14. })
```

## Format navigator

«	1	2	3	»	1 st/rd/th, 3 pages, 15 entries
---	---	---	---	---	---------------------------------

- 1
- 2
- 3
- 4
- 5

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 15],
3.   pageSize: 5,
4.   showNavigator: true,
5.   formatNavigator: '<span style="color: #f00"><%= currentPage
6.   position: 'top',
7.   callback: function(data, pagination) {
8.     // template method of yourself
9.     var html = template(data);
10.    dataContainer.html(html);
11.  }
12. })
```

## Format "go" input

- 1
- 2
- 3
- 4
- 5

«	1	2	3	4	5	»	go to	<input type="text"/>	st/rd/th	Go
---	---	---	---	---	---	---	-------	----------------------	----------	----

```
1. $( '#demo' ).pagination({
2.   dataSource: [ 1, 2, 3, 4, 5, 6, 7, ... , 25],
3.   pageSize: 5,
4.   showGoInput: true,
5.   showGoButton: true,
6.   formatGoInput: 'go to <%= input %> st/rd/th',
7.   callback: function(data, pagination) {
8.     // template method of yourself
9.     var html = template(data);
10.    dataContainer.html(html);
11.  }
12. })
```

# Methods & Events

1

2

3

4

5

Previous

Next

Go (5)

Disable

Enable

Destroy

Show

Hide

is disabled?

Total page

Selected page

Selected page data

«

1

2

3

4

5

...

20

»

Go

☒ Choose events to be logged:

- ☒ beforeInit
- ☒ beforeRender
- ☒ beforePaging
- ☒ beforeDestroy
- ☒ beforeDisable
- ☒ beforeEnable
- ☒ beforePreviousOnClick
- ☒ beforePageOnClick
- ☒ beforeNextOnClick
- ☒ beforeGoInputOnEnter
- ☒ beforeGoButtonOnClick
- ☒ afterInit
- ☒ afterRender
- ☒ afterPaging
- ☒ afterDestroy
- ☒ afterDisable
- ☒ afterEnable
- ☒ afterPreviousOnClick
- ☒ afterPageOnClick
- ☒ afterNextOnClick
- ☒ afterGoInputOnEnter
- ☒ afterGoButtonOnClick
- ☒ afterIsFirstPage
- ☒ afterIsLastPage

Events log

Clear