

- [\[User Guide\]\(/3.3/guide\)](#)
- [\[API Browser\]\(/3.3/guide-api\)](#)
- **Kohana**
- Helpers
- Form

## Modules

### 1. Kohana

- 1.
2. [\[Debug\]\(/3.3/guide-api/Debug\)](#)
3. [\[HTTP\\_Cache\]\(/3.3/guide-api/HTTP\\_Cache\)](#)
4. [\[I18n\]\(/3.3/guide-api/I18n\)](#)
5. [\[Kohana\]\(/3.3/guide-api/Kohana\)](#)
6. [\[Kohana\\_Config\\_Writer\]\(/3.3/guide-api/Kohana\\_Config\\_Writer\)](#)
7. [\[Request\]\(/3.3/guide-api/Request\)](#)
8. [\[Request\\_Client\]\(/3.3/guide-api/Request\\_Client\)](#)
9. [\[Request\\_Client\\_Curl\]\(/3.3/guide-api/Request\\_Client\\_Curl\)](#)
10. [\[Request\\_Client\\_External\]\(/3.3/guide-api/Request\\_Client\\_External\)](#)
11. [\[Request\\_Client\\_HTTP\]\(/3.3/guide-api/Request\\_Client\\_HTTP\)](#)
12. [\[Request\\_Client\\_Internal\]\(/3.3/guide-api/Request\\_Client\\_Internal\)](#)
13. [\[Request\\_Client\\_Stream\]\(/3.3/guide-api/Request\\_Client\\_Stream\)](#)
14. [\[Response\]\(/3.3/guide-api/Response\)](#)
15. [\[Route\]\(/3.3/guide-api/Route\)](#)
16. [\[UTF8\]\(/3.3/guide-api/UTF8\)](#)
17. [\[View\]\(/3.3/guide-api/View\)](#)
18. Configuration
  1. [\[Config\]\(/3.3/guide-api/Config\)](#)
  2. [\[Config\\_File\]\(/3.3/guide-api/Config\\_File\)](#)
  3. [\[Config\\_Group\]\(/3.3/guide-api/Config\\_Group\)](#)
  4. [\[Kohana\\_Config\\_File\\_Reader\]\(/3.3/guide-api/Kohana\\_Config\\_File\\_Reader\)](#)
  5. [\[Kohana\\_Config\\_Reader\]\(/3.3/guide-api/Kohana\\_Config\\_Reader\)](#)
  6. [\[Kohana\\_Config\\_Source\]\(/3.3/guide-api/Kohana\\_Config\\_Source\)](#)

### 19. Controller

1. [\[Controller\]\(/3.3/guide-api/Controller\)](#)
2. [\[Controller\\_Template\]\(/3.3/guide-api/Controller\\_Template\)](#)
3. [\[Controller\\_Welcome\]\(/3.3/guide-api/Controller\\_Welcome\)](#)

### 20. Exceptions

1. [\[HTTP\\_Exception\]\(/3.3/guide-api/HTTP\\_Exception\)](#)
2. [\[HTTP\\_Exception\\_300\]\(/3.3/guide-api/HTTP\\_Exception\\_300\)](#)
3. [\[HTTP\\_Exception\\_301\]\(/3.3/guide-api/HTTP\\_Exception\\_301\)](#)
4. [\[HTTP\\_Exception\\_302\]\(/3.3/guide-api/HTTP\\_Exception\\_302\)](#)
5. [\[HTTP\\_Exception\\_303\]\(/3.3/guide-api/HTTP\\_Exception\\_303\)](#)
6. [\[HTTP\\_Exception\\_304\]\(/3.3/guide-api/HTTP\\_Exception\\_304\)](#)
7. [\[HTTP\\_Exception\\_305\]\(/3.3/guide-api/HTTP\\_Exception\\_305\)](#)
8. [\[HTTP\\_Exception\\_307\]\(/3.3/guide-api/HTTP\\_Exception\\_307\)](#)
9. [\[HTTP\\_Exception\\_400\]\(/3.3/guide-api/HTTP\\_Exception\\_400\)](#)
10. [\[HTTP\\_Exception\\_401\]\(/3.3/guide-api/HTTP\\_Exception\\_401\)](#)
11. [\[HTTP\\_Exception\\_402\]\(/3.3/guide-api/HTTP\\_Exception\\_402\)](#)
12. [\[HTTP\\_Exception\\_403\]\(/3.3/guide-api/HTTP\\_Exception\\_403\)](#)
13. [\[HTTP\\_Exception\\_404\]\(/3.3/guide-api/HTTP\\_Exception\\_404\)](#)
14. [\[HTTP\\_Exception\\_405\]\(/3.3/guide-api/HTTP\\_Exception\\_405\)](#)
15. [\[HTTP\\_Exception\\_406\]\(/3.3/guide-api/HTTP\\_Exception\\_406\)](#)
16. [\[HTTP\\_Exception\\_407\]\(/3.3/guide-api/HTTP\\_Exception\\_407\)](#)
17. [\[HTTP\\_Exception\\_408\]\(/3.3/guide-api/HTTP\\_Exception\\_408\)](#)
18. [\[HTTP\\_Exception\\_409\]\(/3.3/guide-api/HTTP\\_Exception\\_409\)](#)
19. [\[HTTP\\_Exception\\_410\]\(/3.3/guide-api/HTTP\\_Exception\\_410\)](#)
20. [\[HTTP\\_Exception\\_411\]\(/3.3/guide-api/HTTP\\_Exception\\_411\)](#)
21. [\[HTTP\\_Exception\\_412\]\(/3.3/guide-api/HTTP\\_Exception\\_412\)](#)
22. [\[HTTP\\_Exception\\_413\]\(/3.3/guide-api/HTTP\\_Exception\\_413\)](#)
23. [\[HTTP\\_Exception\\_414\]\(/3.3/guide-api/HTTP\\_Exception\\_414\)](#)

- 24. [\[HTTP\\_Exception\\_415\]\(/3.3/guide-api/HTTP\\_Exception\\_415\)](#)
- 25. [\[HTTP\\_Exception\\_416\]\(/3.3/guide-api/HTTP\\_Exception\\_416\)](#)
- 26. [\[HTTP\\_Exception\\_417\]\(/3.3/guide-api/HTTP\\_Exception\\_417\)](#)
- 27. [\[HTTP\\_Exception\\_500\]\(/3.3/guide-api/HTTP\\_Exception\\_500\)](#)
- 28. [\[HTTP\\_Exception\\_501\]\(/3.3/guide-api/HTTP\\_Exception\\_501\)](#)
- 29. [\[HTTP\\_Exception\\_502\]\(/3.3/guide-api/HTTP\\_Exception\\_502\)](#)
- 30. [\[HTTP\\_Exception\\_503\]\(/3.3/guide-api/HTTP\\_Exception\\_503\)](#)
- 31. [\[HTTP\\_Exception\\_504\]\(/3.3/guide-api/HTTP\\_Exception\\_504\)](#)
- 32. [\[HTTP\\_Exception\\_505\]\(/3.3/guide-api/HTTP\\_Exception\\_505\)](#)
- 33. [\[HTTP\\_Exception\\_Expected\]\(/3.3/guide-api/HTTP\\_Exception\\_Expected\)](#)
- 34. [\[HTTP\\_Exception\\_Redirect\]\(/3.3/guide-api/HTTP\\_Exception\\_Redirect\)](#)
- 35. [\[Kohana\\_Exception\]\(/3.3/guide-api/Kohana\\_Exception\)](#)
- 36. [\[Request\\_Client\\_Recursion\\_Exception\]\(/3.3/guide-api/Request\\_Client\\_Recursion\\_Exception\)](#)
- 37. [\[Request\\_Exception\]\(/3.3/guide-api/Request\\_Exception\)](#)
- 38. [\[Session\\_Exception\]\(/3.3/guide-api/Session\\_Exception\)](#)
- 39. [\[UTF8\\_Exception\]\(/3.3/guide-api/UTF8\\_Exception\)](#)
- 40. [\[Validation\\_Exception\]\(/3.3/guide-api/Validation\\_Exception\)](#)
- 41. [\[View\\_Exception\]\(/3.3/guide-api/View\\_Exception\)](#)

## 21. HTTP

- 1. [\[HTTP\]\(/3.3/guide-api/HTTP\)](#)
- 2. [\[HTTP\\_Header\]\(/3.3/guide-api/HTTP\\_Header\)](#)
- 3. [\[HTTP\\_Message\]\(/3.3/guide-api/HTTP\\_Message\)](#)
- 4. [\[HTTP\\_Request\]\(/3.3/guide-api/HTTP\\_Request\)](#)
- 5. [\[HTTP\\_Response\]\(/3.3/guide-api/HTTP\\_Response\)](#)

## 22. Helpers

- 1. [\[Arr\]\(/3.3/guide-api/Arr\)](#)
- 2. [\[Cookie\]\(/3.3/guide-api/Cookie\)](#)
- 3. [\[Date\]\(/3.3/guide-api/Date\)](#)
- 4. [\[Feed\]\(/3.3/guide-api/Feed\)](#)
- 5. [\[File\]\(/3.3/guide-api/File\)](#)
- 6. [\[Form\]\(/3.3/guide-api/Form\)](#)
- 7. [\[Fragment\]\(/3.3/guide-api/Fragment\)](#)
- 8. [\[HTML\]\(/3.3/guide-api/HTML\)](#)
- 9. [\[Inflector\]\(/3.3/guide-api/Inflector\)](#)
- 10. [\[Num\]\(/3.3/guide-api/Num\)](#)
- 11. [\[Profiler\]\(/3.3/guide-api/Profiler\)](#)
- 12. [\[Task\\_Help\]\(/3.3/guide-api/Task\\_Help\)](#)
- 13. [\[Text\]\(/3.3/guide-api/Text\)](#)
- 14. [\[URL\]\(/3.3/guide-api/URL\)](#)
- 15. [\[Upload\]\(/3.3/guide-api/Upload\)](#)

## 23. Logging

- 1. [\[Log\]\(/3.3/guide-api/Log\)](#)
- 2. [\[Log\\_File\]\(/3.3/guide-api/Log\\_File\)](#)
- 3. [\[Log\\_StdErr\]\(/3.3/guide-api/Log\\_StdErr\)](#)
- 4. [\[Log\\_StdOut\]\(/3.3/guide-api/Log\\_StdOut\)](#)
- 5. [\[Log\\_Syslog\]\(/3.3/guide-api/Log\\_Syslog\)](#)
- 6. [\[Log\\_Writer\]\(/3.3/guide-api/Log\\_Writer\)](#)

## 24. Minion

- 1. [\[Minion\\_Exception\]\(/3.3/guide-api/Minion\\_Exception\)](#)
- 2. [\[Minion\\_Exception\\_InvalidTask\]\(/3.3/guide-api/Minion\\_Exception\\_InvalidTask\)](#)

## 25. Models

- 1. [\[Model\]\(/3.3/guide-api/Model\)](#)

## 26. Security

- 1. [\[Encrypt\]\(/3.3/guide-api/Encrypt\)](#)
- 2. [\[Security\]\(/3.3/guide-api/Security\)](#)
- 3. [\[Valid\]\(/3.3/guide-api/Valid\)](#)
- 4. [\[Validation\]\(/3.3/guide-api/Validation\)](#)

## 27. Session

- 1. [\[Session\]\(/3.3/guide-api/Session\)](#)
- 2. [\[Session\\_Cookie\]\(/3.3/guide-api/Session\\_Cookie\)](#)
- 3. [\[Session\\_Native\]\(/3.3/guide-api/Session\\_Native\)](#)

## 2. Kohana/Auth

- 1.
2. [\[Auth\]/\(3.3/guide-api/Auth\)](#)
3. [\[Auth\\_File\]/\(3.3/guide-api/Auth\\_File\)](#)
4. [\[Auth ORM\]/\(3.3/guide-api/Auth ORM\)](#)
5. [\[Model\\_Auth\\_Role\]/\(3.3/guide-api/Model\\_Auth\\_Role\)](#)
6. [\[Model\\_Auth\\_User\]/\(3.3/guide-api/Model\\_Auth\\_User\)](#)
7. [\[Model\\_Auth\\_User\\_Token\]/\(3.3/guide-api/Model\\_Auth\\_User\\_Token\)](#)
8. [\[Model\\_Role\]/\(3.3/guide-api/Model\\_Role\)](#)
9. [\[Model\\_User\]/\(3.3/guide-api/Model\\_User\)](#)
10. [\[Model\\_User\\_Token\]/\(3.3/guide-api/Model\\_User\\_Token\)](#)

### 3. Kohana/Cache

- 1.
2. [\[Cache\]/\(3.3/guide-api/Cache\)](#)
3. [\[Cache\\_Apc\]/\(3.3/guide-api/Cache\\_Apc\)](#)
4. [\[Cache\\_Arithmetic\]/\(3.3/guide-api/Cache\\_Arithmetic\)](#)
5. [\[Cache\\_Exception\]/\(3.3/guide-api/Cache\\_Exception\)](#)
6. [\[Cache\\_File\]/\(3.3/guide-api/Cache\\_File\)](#)
7. [\[Cache\\_GarbageCollect\]/\(3.3/guide-api/Cache\\_GarbageCollect\)](#)
8. [\[Cache\\_Memcache\]/\(3.3/guide-api/Cache\\_Memcache\)](#)
9. [\[Cache\\_MemcacheTag\]/\(3.3/guide-api/Cache\\_MemcacheTag\)](#)
10. [\[Cache\\_Sqlite\]/\(3.3/guide-api/Cache\\_Sqlite\)](#)
11. [\[Cache\\_Tagging\]/\(3.3/guide-api/Cache\\_Tagging\)](#)
12. [\[Cache\\_Wincache\]/\(3.3/guide-api/Cache\\_Wincache\)](#)

### 4. Kohana/Codebench

- 1.
2. [\[Codebench\]/\(3.3/guide-api/Codebench\)](#)
3. Controllers
  1. [\[Controller\\_Codebench\]/\(3.3/guide-api/Controller\\_Codebench\)](#)
4. Tests
  1. [\[Bench\\_ArrCallback\]/\(3.3/guide-api/Bench\\_ArrCallback\)](#)
  2. [\[Bench\\_AutoLinkEmails\]/\(3.3/guide-api/Bench\\_AutoLinkEmails\)](#)
  3. [\[Bench\\_DateSpan\]/\(3.3/guide-api/Bench\\_DateSpan\)](#)
  4. [\[Bench\\_ExplodeLimit\]/\(3.3/guide-api/Bench\\_ExplodeLimit\)](#)
  5. [\[Bench\\_GruberURL\]/\(3.3/guide-api/Bench\\_GruberURL\)](#)
  6. [\[Bench\\_LtrimDigits\]/\(3.3/guide-api/Bench\\_LtrimDigits\)](#)
  7. [\[Bench\\_MDDoBaseURL\]/\(3.3/guide-api/Bench\\_MDDoBaseURL\)](#)
  8. [\[Bench\\_MDDoImageURL\]/\(3.3/guide-api/Bench\\_MDDoImageURL\)](#)
  9. [\[Bench\\_MDDoIncludeViews\]/\(3.3/guide-api/Bench\\_MDDoIncludeViews\)](#)
  10. [\[Bench\\_StripNullBytes\]/\(3.3/guide-api/Bench\\_StripNullBytes\)](#)
  11. [\[Bench\\_Transliterate\]/\(3.3/guide-api/Bench\\_Transliterate\)](#)
  12. [\[Bench\\_URLSite\]/\(3.3/guide-api/Bench\\_URLSite\)](#)
  13. [\[Bench\\_UserFuncArray\]/\(3.3/guide-api/Bench\\_UserFuncArray\)](#)
  14. [\[Bench\\_ValidColor\]/\(3.3/guide-api/Bench\\_ValidColor\)](#)
  15. [\[Bench\\_ValidURL\]/\(3.3/guide-api/Bench\\_ValidURL\)](#)

### 5. Kohana/Database

- 1.
2. [\[DB\]/\(3.3/guide-api/DB\)](#)
3. [\[Database\]/\(3.3/guide-api/Database\)](#)
4. [\[Database\\_Expression\]/\(3.3/guide-api/Database\\_Expression\)](#)
5. Configuration
  1. [\[Config\\_Database\]/\(3.3/guide-api/Config\\_Database\)](#)
  2. [\[Config\\_Database\\_Reader\]/\(3.3/guide-api/Config\\_Database\\_Reader\)](#)
  3. [\[Config\\_Database\\_Writer\]/\(3.3/guide-api/Config\\_Database\\_Writer\)](#)
6. Drivers
  1. [\[Database\\_MySQL\]/\(3.3/guide-api/Database\\_MySQL\)](#)
  2. [\[Database\\_MySQLi\]/\(3.3/guide-api/Database\\_MySQLi\)](#)
  3. [\[Database\\_PDO\]/\(3.3/guide-api/Database\\_PDO\)](#)
7. Exceptions
  1. [\[Database\\_Exception\]/\(3.3/guide-api/Database\\_Exception\)](#)
8. Models
  1. [\[Model\\_Database\]/\(3.3/guide-api/Model\\_Database\)](#)
9. Query

1. [\[Database\\_Query\]/\(3.3/guide-api/Database\\_Query\)](#)
2. [\[Database\\_Query\\_Builder\]/\(3.3/guide-api/Database\\_Query\\_Builder\)](#)
3. [\[Database\\_Query\\_Builder\\_Delete\]/\(3.3/guide-api/Database\\_Query\\_Builder\\_Delete\)](#)
4. [\[Database\\_Query\\_Builder\\_Insert\]/\(3.3/guide-api/Database\\_Query\\_Builder\\_Insert\)](#)
5. [\[Database\\_Query\\_Builder\\_Join\]/\(3.3/guide-api/Database\\_Query\\_Builder\\_Join\)](#)
6. [\[Database\\_Query\\_Builder\\_Select\]/\(3.3/guide-api/Database\\_Query\\_Builder\\_Select\)](#)
7. [\[Database\\_Query\\_Builder\\_Update\]/\(3.3/guide-api/Database\\_Query\\_Builder\\_Update\)](#)
8. [\[Database\\_Query\\_Builder\\_Where\]/\(3.3/guide-api/Database\\_Query\\_Builder\\_Where\)](#)
10. Query/Result
  1. [\[Database\\_MySQL\\_Result\]/\(3.3/guide-api/Database\\_MySQL\\_Result\)](#)
  2. [\[Database\\_MySQLi\\_Result\]/\(3.3/guide-api/Database\\_MySQLi\\_Result\)](#)
  3. [\[Database\\_Result\]/\(3.3/guide-api/Database\\_Result\)](#)
  4. [\[Database\\_Result\\_Cached\]/\(3.3/guide-api/Database\\_Result\\_Cached\)](#)
11. Session
  1. [\[Session\\_Database\]/\(3.3/guide-api/Session\\_Database\)](#)
6. **Kohana/Image**
  - 1.
  2. [\[Image\]/\(3.3/guide-api/Image\)](#)
  3. Drivers
    1. [\[Image\\_GD\]/\(3.3/guide-api/Image\\_GD\)](#)
    2. [\[Image\\_Imagick\]/\(3.3/guide-api/Image\\_Imagick\)](#)
7. **Kohana/ORM**
  - 1.
  2. [\[ORM\]/\(3.3/guide-api/ORM\)](#)
  3. [\[ORM\\_Validation\\_Exception\]/\(3.3/guide-api/ORM\\_Validation\\_Exception\)](#)
8. **Kohana/UnitTest**
  - 1.
  2. [\[Unittest\\_Database\\_TestCase\]/\(3.3/guide-api/Unittest\\_Database\\_TestCase\)](#)
  3. [\[Unittest\\_Tests\]/\(3.3/guide-api/Unittest\\_Tests\)](#)
9. **Kohana/Userguide**
  - 1.
  2. [\[Kodoc\]/\(3.3/guide-api/Kodoc\)](#)
  3. [\[Kodoc\\_Class\]/\(3.3/guide-api/Kodoc\\_Class\)](#)
  4. [\[Kodoc\\_Markdown\]/\(3.3/guide-api/Kodoc\\_Markdown\)](#)
  5. [\[Kodoc\\_Method\]/\(3.3/guide-api/Kodoc\\_Method\)](#)
  6. [\[Kodoc\\_Method\\_Param\]/\(3.3/guide-api/Kodoc\\_Method\\_Param\)](#)
  7. [\[Kodoc\\_Property\]/\(3.3/guide-api/Kodoc\\_Property\)](#)
  8. Controller
    1. [\[Controller\\_Userguide\]/\(3.3/guide-api/Controller\\_Userguide\)](#)
  9. Undocumented
    1. [\[Kodoc\\_Missing\]/\(3.3/guide-api/Kodoc\\_Missing\)](#)
10. **[Unknown]**
  - 1.
  2. [\[Minion\\_CLI\]/\(3.3/guide-api/Minion\\_CLI\)](#)
  3. [\[Minion\\_Task\]/\(3.3/guide-api/Minion\\_Task\)](#)
  4. [\[Unittest\\_Helpers\]/\(3.3/guide-api/Unittest\\_Helpers\)](#)
  5. [\[Unittest\\_TestCase\]/\(3.3/guide-api/Unittest\\_TestCase\)](#)
  6. [\[Unittest\\_TestSuite\]/\(3.3/guide-api/Unittest\\_TestSuite\)](#)

## Form

**extends** [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#)

Form helper class. Unless otherwise noted, all generated HTML will be made safe using the [\[HTML::chars\]/\(3.3/guide-api/HTML#chars\)](#) method. This prevents against simple XSS attacks that could otherwise be triggered by inserting HTML characters into form fields.

```
package
  Kohana
category
  Helpers
author
```

Kohana Team

copyright

© 2007-2012 Kohana Team

license

[\[http://kohanaframework.org/license\]](http://kohanaframework.org/license)(<http://kohanaframework.org/license>)

**Note:** Class declared in `SYSPATH/classes/Form.php` on line 3.

## Constants

- *None*

## Properties

- *None*

## Methods

- [\[button\(\)\]\(#button\)](#)
- [\[checkbox\(\)\]\(#checkbox\)](#)
- [\[close\(\)\]\(#close\)](#)
- [\[file\(\)\]\(#file\)](#)
- [\[hidden\(\)\]\(#hidden\)](#)
- [\[image\(\)\]\(#image\)](#)
- [\[input\(\)\]\(#input\)](#)
- [\[label\(\)\]\(#label\)](#)
- [\[open\(\)\]\(#open\)](#)
- [\[password\(\)\]\(#password\)](#)
- [\[radio\(\)\]\(#radio\)](#)
- [\[select\(\)\]\(#select\)](#)
- [\[submit\(\)\]\(#submit\)](#)
- [\[textarea\(\)\]\(#textarea\)](#)

# Methods

**public static button( string \$name , string \$body [, array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]\(/3.3/guide-api/Kohana\\_Form\)](#))

Creates a button form input. Note that the body of a button is NOT escaped, to allow images and other HTML to be used.

```
echo Form::button('save', 'Save Profile', array('type' => 'submit'));
```

## Parameters

- string **\$name** required - Input name
- string **\$body** required - Input value
- array **\$attributes** = NULL - Html attributes

## Tags

- Uses - [\[HTML::attributes\]\(/3.3/guide-api/HTML#attributes\)](#)

## Return Values

- string

## Source Code

```
public static function button($name, $body, array $attributes = NULL)
{
```

```
// Set the input name
$attributes['name'] = $name;

return '<button'.HTML::attributes($attributes).'>'.$body.'</button>';
}
```

**public static checkbox( string \$name [, string \$value = NULL , boolean \$checked = bool FALSE , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#))

Creates a checkbox form input.

```
echo Form::checkbox('remember_me', 1, (bool) $remember);
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** = NULL - Input value
- boolean **\$checked** = bool FALSE - Checked status
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[Form::input\]/\(3.3/guide-api/Form#input\)](#)

#### Return Values

- string

#### Source Code

```
public static function checkbox($name, $value = NULL, $checked = FALSE, array $attributes = NULL)
{
    $attributes['type'] = 'checkbox';

    if ($checked === TRUE)
    {
        // Make the checkbox active
        $attributes[] = 'checked';
    }

    return Form::input($name, $value, $attributes);
}
```

**public static close( )** (defined in [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#))

Creates the closing form tag.

```
echo Form::close();
```

#### Return Values

- string

#### Source Code

```
public static function close()
{
    return '</form>';
}
```

```
}
```

**public static file( string \$name [, array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#))

Creates a file upload form input. No input value can be specified.

```
echo Form::file('image');
```

#### Parameters

- string **\$name** required - Input name
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[Form::input\]/\(3.3/guide-api/Form#input\)](#)

#### Return Values

- string

#### Source Code

```
public static function file($name, array $attributes = NULL)
{
    $attributes['type'] = 'file';

    return Form::input($name, NULL, $attributes);
}
```

**public static hidden( string \$name [, string \$value = NULL , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#))

Creates a hidden form input.

```
echo Form::hidden('csrf', $token);
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** = NULL - Input value
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[Form::input\]/\(3.3/guide-api/Form#input\)](#)

#### Return Values

- string

#### Source Code

```
public static function hidden($name, $value = NULL, array $attributes = NULL)
{
    $attributes['type'] = 'hidden';

    return Form::input($name, $value, $attributes);
}
```

```
}

```

**public static image( string \$name , string \$value [, array \$attributes = NULL , boolean \$index = bool FALSE ] )** (defined in [\[Kohana\\_Form\]](#) [\(/3.3/guide-api/Kohana\\_Form\)](#))

Creates a image form input.

```
echo Form::image(NULL, NULL, array('src' => 'media/img/login.png'));
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** required - Input value
- array **\$attributes** = NULL - Html attributes
- boolean **\$index** = bool FALSE - Add index file to URL?

#### Tags

- Uses - [\[Form::input\]\(/3.3/guide-api/Form#input\)](#)

#### Return Values

- string

#### Source Code

```
public static function image($name, $value, array $attributes = NULL, $index = FALSE)
{
    if ( ! empty($attributes['src']))
    {
        if (strpos($attributes['src'], '://') === FALSE)
        {
            // Add the base URL
            $attributes['src'] = URL::base($index).$attributes['src'];
        }
    }

    $attributes['type'] = 'image';

    return Form::input($name, $value, $attributes);
}
```

**public static input( string \$name [, string \$value = NULL , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]](#) [\(/3.3/guide-api/Kohana\\_Form\)](#))

Creates a form input. If no type is specified, a "text" type input will be returned.

```
echo Form::input('username', $username);
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** = NULL - Input value
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[HTML::attributes\]\(/3.3/guide-api/HTML#attributes\)](#)

#### Return Values



- string

### Source Code

```
public static function input($name, $value = NULL, array $attributes = NULL)
{
    // Set the input name
    $attributes['name'] = $name;

    // Set the input value
    $attributes['value'] = $value;

    if ( ! isset($attributes['type']))
    {
        // Default type is text
        $attributes['type'] = 'text';
    }

    return '<input'.HTML::attributes($attributes).' />';
}
```

**public static label( string \$input [, string \$text = NULL , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#))

Creates a form label. Label text is not automatically translated.

```
echo Form::label('username', 'Username');
```

### Parameters

- string **\$input** required - Target input
- string **\$text** = NULL - Label text
- array **\$attributes** = NULL - Html attributes

### Tags

- Uses - [\[HTML::attributes\]/\(3.3/guide-api/HTML#attributes\)](#)

### Return Values

- string

### Source Code

```
public static function label($input, $text = NULL, array $attributes = NULL)
{
    if ($text === NULL)
    {
        // Use the input name as the text
        $text = ucwords(preg_replace('/[\W_]+/', ' ', $input));
    }

    // Set the label target
    $attributes['for'] = $input;

    return '<label'.HTML::attributes($attributes).'>'.$text.'</label>';
}
```

**public static open( [ mixed \$action = NULL , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/\(3.3/guide-api/Kohana\\_Form\)](#))

Generates an opening HTML form tag.

```
// Form will submit back to the current page using POST
echo Form::open();

// Form will submit to 'search' using GET
echo Form::open('search', array('method' => 'get'));

// When "file" inputs are present, you must include the "enctype"
echo Form::open(NULL, array('enctype' => 'multipart/form-data'));
```

### Parameters

- mixed **\$action** = NULL - Form action, defaults to the current request URI, or [Request] class to use
- array **\$attributes** = NULL - Html attributes

### Tags

- Uses - [\[Request\]/\(3.3/guide-api/Request\)](#), [\[URL::site\]/\(3.3/guide-api/URL#site\)](#), [\[HTML::attributes\]/\(3.3/guide-api/HTML#attributes\)](#)

### Return Values

- string

### Source Code

```
public static function open($action = NULL, array $attributes = NULL)
{
    if ($action instanceof Request)
    {
        // Use the current URI
        $action = $action->uri();
    }

    if ( ! $action)
    {
        // Allow empty form actions (submits back to the current url).
        $action = '';
    }
    elseif (strpos($action, '://') === FALSE)
    {
        // Make the URI absolute
        $action = URL::site($action);
    }

    // Add the form action to the attributes
    $attributes['action'] = $action;

    // Only accept the default character set
    $attributes['accept-charset'] = Kohana::$charset;

    if ( ! isset($attributes['method']))
    {
        // Use POST method
        $attributes['method'] = 'post';
    }

    return '<form'.HTML::attributes($attributes). '>';
}
```

**public static password( string \$name [, string \$value = NULL , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/3.3/guide-api/Kohana\\_Form](#))

Creates a password form input.

```
echo Form::password('password');
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** = NULL - Input value
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[Form::input\]/3.3/guide-api/Form#input](#)

#### Return Values

- string

#### Source Code

```
public static function password($name, $value = NULL, array $attributes = NULL)
{
    $attributes['type'] = 'password';

    return Form::input($name, $value, $attributes);
}
```

**public static radio( string \$name [, string \$value = NULL , boolean \$checked = bool FALSE , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/3.3/guide-api/Kohana\\_Form](#))

Creates a radio form input.

```
echo Form::radio('like_cats', 1, $cats);
echo Form::radio('like_cats', 0, ! $cats);
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** = NULL - Input value
- boolean **\$checked** = bool FALSE - Checked status
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[Form::input\]/3.3/guide-api/Form#input](#)

#### Return Values

- string

#### Source Code

```
public static function radio($name, $value = NULL, $checked = FALSE, array $attributes = NULL)
{
    $attributes['type'] = 'radio';
```

```
if ($checked === TRUE)
{
    // Make the radio active
    $attributes[] = 'checked';
}

return Form::input($name, $value, $attributes);
}
```

**public static select( string \$name [, array \$options = NULL , mixed \$selected = NULL , array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]](#) ([/3.3/guide-api/Kohana\\_Form](#)))

Creates a select form input.

```
echo Form::select('country', $countries, $country);
```

**Note:** Support for multiple selected options was added in v3.0.7.

### Parameters

- string **\$name** required - Input name
- array **\$options** = NULL - Available options
- mixed **\$selected** = NULL - Selected option string, or an array of selected options
- array **\$attributes** = NULL - Html attributes

### Tags

- Uses - [\[HTML::attributes\]/3.3/guide-api/HTML#attributes](#)

### Return Values

- string

### Source Code

```
public static function select($name, array $options = NULL, $selected = NULL, array $attributes = NULL)
{
    // Set the input name
    $attributes['name'] = $name;

    if (is_array($selected))
    {
        // This is a multi-select, god save us!
        $attributes[] = 'multiple';
    }

    if ( ! is_array($selected))
    {
        if ($selected === NULL)
        {
            // Use an empty array
            $selected = array();
        }
        else
        {
            // Convert the selected options to an array
            $selected = array( (string) $selected);
        }
    }
}
```

```
if (empty($options))
{
    // There are no options
    $options = '';
}
else
{
    foreach ($options as $value => $name)
    {
        if (is_array($name))
        {
            // Create a new optgroup
            $group = array('label' => $value);

            // Create a new list of options
            $_options = array();

            foreach ($name as $_value => $_name)
            {
                // Force value to be string
                $_value = (string) $_value;

                // Create a new attribute set for this option
                $option = array('value' => $_value);

                if (in_array($_value, $selected))
                {
                    // This option is selected
                    $option[] = 'selected';
                }

                // Change the option to the HTML string
                $_options[] = '<option'.HTML::attributes($option).'>'.HTML::chars($_name, FALSE).'</option>';
            }

            // Compile the options into a string
            $_options = "\n".implode("\n", $_options)."\n";

            $options[$value] = '<optgroup'.HTML::attributes($group).'>'. $_options.'</optgroup>';
        }
        else
        {
            // Force value to be string
            $value = (string) $value;

            // Create a new attribute set for this option
            $option = array('value' => $value);

            if (in_array($value, $selected))
            {
                // This option is selected
                $option[] = 'selected';
            }

            // Change the option to the HTML string
            $options[$value] = '<option'.HTML::attributes($option).'>'.HTML::chars($name, FALSE).'</option>';
        }
    }

    // Compile the options into a single string
    $options = "\n".implode("\n", $options)."\n";
}
```

```

    }

    return '<select'.HTML::attributes($attributes).'>'.$options.'</select>';
}

```

**public static submit( string \$name , string \$value [, array \$attributes = NULL ] )** (defined in [\[Kohana\\_Form\]/3.3/guide-api/Kohana\\_Form](#))

Creates a submit form input.

```
echo Form::submit(NULL, 'Login');
```

#### Parameters

- string **\$name** required - Input name
- string **\$value** required - Input value
- array **\$attributes** = NULL - Html attributes

#### Tags

- Uses - [\[Form::input\]/3.3/guide-api/Form#input](#)

#### Return Values

- string

#### Source Code

```

public static function submit($name, $value, array $attributes = NULL)
{
    $attributes['type'] = 'submit';

    return Form::input($name, $value, $attributes);
}

```

**public static textarea( string \$name [, string \$body = string(0) "" , array \$attributes = NULL , boolean \$double\_encode = bool TRUE ] )**  
(defined in [\[Kohana\\_Form\]/3.3/guide-api/Kohana\\_Form](#))

Creates a textarea form input.

```
echo Form::textarea('about', $about);
```

#### Parameters

- string **\$name** required - Textarea name
- string **\$body** = string(0) "" - Textarea body
- array **\$attributes** = NULL - Html attributes
- boolean **\$double\_encode** = bool TRUE - Encode existing HTML characters

#### Tags

- Uses - [\[HTML::attributes\]/3.3/guide-api/HTML#attributes](#), [\[HTML::chars\]/3.3/guide-api/HTML#chars](#)

#### Return Values

- string

#### Source Code

```
public static function textarea($name, $body = '', array $attributes = NULL, $double_encode = TRUE)
{
    // Set the input name
    $attributes['name'] = $name;

    // Add default rows and cols attributes (required)
    $attributes += array('rows' => 10, 'cols' => 50);

    return '<textarea'.HTML::attributes($attributes).'>'.HTML::chars($body, $double_encode).'</textarea>';
}
```

Powered by [\[Kohana\]](http://kohanaframework.org/) v3.3.4