

การสรุป Built-in Template Filters ทั้งหมดในเอกสารของ Django พร้อม ตัวอย่าง จะเป็นข้อมูลที่ค่อนข้างยาว แต่จะ ทำการสรุปโดยย่อและให้ตัวอย่างที่สำคัญ สำหรับแต่ละตัวกรอง เพื่อให้ครอบคลุม การใช้งานทั้งหมด:

Built-in Template Filters ของ Django

1. **add(value):** เพิ่มค่าให้กับตัวเลข
 - `{{ 4|add:"2" }}` → 6
2. **addslashes:** เพิ่มเครื่องหมาย \ หน้าตัวอักขระพิเศษ
 - `{{ "It's me!"|addslashes }}` → `It\'s me!`
3. **capfirst:** เปลี่ยนอักษรตัวแรกของสตริงเป็นตัวพิมพ์ใหญ่
 - `{{ "hello"|capfirst }}` → `Hello`
4. **center(width):** จัดสตริงให้อยู่ตรงกลางตามความกว้างที่ระบุ
 - `{{ "Django"|center:"10" }}` → `" Django "`
5. **cut(value):** ลบค่าที่ระบุออกจากสตริง
 - `{{ "Hello World"|cut:"o" }}` → `Hell Wrld`
6. **date(format):** จัดรูปแบบวันที่
 - `{{ my_date|date:"Y-m-d" }}` → `2024-08-26`
7. **default(value):** คืนค่า default หากค่าปัจจุบันเป็น None หรือ ""
 - `{{ my_var|default:"N/A" }}`

8. **default_if_none(value):** คืนค่า default หากค่าปัจจุบันเป็น None
- `{{ my_var|default_if_none:"N/A" }}`
9. **dictsort(key):** เรียงลำดับดิกชันนารีตามคีย์ที่ระบุ
- `{{ my_dict|dictsort:"name" }}`
10. **dictsortreversed(key):** เรียงลำดับดิกชันนารีตามคีย์ที่ระบุ ในทิศทางตรงข้าม
- `{{ my_dict|dictsortreversed:"name" }}`
11. **divisibleby(number):** ตรวจสอบว่าค่าหารด้วยตัวเลขที่ระบุ ได้ลงตัวหรือไม่
- `{{ 10|divisibleby:"2" }}` → True
12. **escape:** ทำการ escape สตริงให้ปลอดภัย
- `{{ "<div>"|escape }}` → <div>
13. **escapejs:** ทำการ escape สตริงสำหรับใช้ใน JavaScript
- `{{ "<script>"|escapejs }}` → \u003Cscript\u003E
14. **filesizeformat:** แปลงค่าขนาดไฟล์ให้เป็นรูปแบบที่อ่านง่าย
- `{{ 1024|filesizeformat }}` → 1 KB
15. **first:** คืนค่าแรกในลิสต์
- `{{ my_list|first }}`
16. **floatformat:** จัดรูปแบบตัวเลขทศนิยม
- `{{ 123.456|floatformat:"2" }}` → 123.46
17. **force_escape:** Escape สตริงแบบบังคับ
- `{{ my_var|force_escape }}`
18. **get_digit(position):** คืนค่าหลักของตัวเลขที่ระบุตำแหน่ง
- `{{ 12345|get_digit:"2" }}` → 4 (หลักที่สอง)
19. **iriencode:** Encode สตริงสำหรับ URI
- `{{ "http://example.com/?q=привет"|iriencode }}`
20. **join(separator):** ต่อสตริงในลิสต์ด้วยตัวคั่นที่ระบุ

- `{{ my_list|join:", " }}`

21. json_script: แปลงข้อมูลเป็น JSON สำหรับใช้ใน script

- `{{ my_data|json_script:"my-id" }}`

22. last: คำนวณค่าล่าสุดในลิสต์

- `{{ my_list|last }}`

23. length: คำนวณความยาวของลิสต์หรือสตริง

- `{{ my_list|length }}`

24. length_is(value): ตรวจสอบว่าความยาวของลิสต์หรือสตริงตรงกับค่าที่ระบุหรือไม่

- `{{ my_list|length_is:"3" }}`

25. linebreaks: แปลงช่องว่างใหม่เป็นแท็ก <p>

- `{{ my_text|linebreaks }}`

**26. linebreaksbr: แปลงช่องว่างใหม่เป็นแท็ก
**

- `{{ my_text|linebreaksbr }}`

27. linenumbers: เพิ่มหมายเลขบรรทัดให้กับสตริง

- `{{ my_text|linenumbers }}`

28. ljust(width): จัดสตริงไปทางซ้ายตามความกว้างที่ระบุ

- `{{ "Django"|ljust:"10" }}` → "Django "

29. lower: แปลงสตริงเป็นตัวพิมพ์เล็ก

- `{{ "HELLO"|lower }}` → hello

30. make_list: แปลงสตริงให้เป็นลิสต์ของอักขระ

- `{{ "abc"|make_list }}` → ['a', 'b', 'c']

31. phone2numeric: แปลงตัวเลขในโทรศัพท์เป็นตัวเลขตามแป้นโทรศัพท์

- `{{ "800-DJANGO"|phone2numeric }}` → 800-352646

32. pluralize: เพิ่มคำลงท้ายตามรูปพหูพจน์

- `{{ 1|pluralize:"s" }}` → "" และ `{{ 2|pluralize:"s" }}` → "s"

33. pprint: จัดรูปแบบข้อมูลให้สวยงาม

- `{{ my_data|pprint }}`

34. random: คัดค่าสุ่มจากลิสต์

- `{{ my_list|random }}`

35. rjust(width): จัดสตริงไปทางขวาตามความกว้างที่ระบุ

- `{{ "Django"|rjust:"10" }}` → " Django"

36. safe: ระบุว่าสตริงนั้นปลอดภัยและไม่ต้อง escape

- `{{ my_var|safe }}`

37. safeseq: ระบุว่าลิสต์นั้นปลอดภัยและไม่ต้อง escape

- `{{ my_list|safeseq }}`

38. slice(start

39.): คัดค่าช่วงของลิสต์หรือสตริงตาม start และ end

- `{{ my_list|slice:"1:4" }}`

40. slugify: แปลงสตริงเป็น slug (รูปแบบ URL-friendly)

- `{{ "Hello World!"|slugify }}` → hello-world

41. stringformat(format): จัดรูปแบบสตริงตามรูปแบบที่ระบุ

- `{{ 123|stringformat:"04d" }}` → 0123

42. striptags: ลบแท็ก HTML ทั้งหมด

- `{{ my_html|striptags }}`

43. time(format): จัดรูปแบบเวลา

- `{{ my_time|time:"H:i" }}`

44. timesince: แสดงระยะเวลาที่ผ่านมา

- `{{ my_date|timesince }}`

45. timeuntil: แสดงระยะเวลาที่เหลือ

- `{{ my_date|timeuntil }}`

46. title: แปลงสตริงให้ตัวอักษรแรกของแต่ละคำเป็นตัวพิมพ์ใหญ่

- `{{ "hello world"|title }}` → Hello World

47. **truncatechars(length):** ตัดสตริงที่ยาวเกินตามความยาวที่ระบุ
- `{{ my_text|truncatechars:"10" }}`
48. **truncatechars_html(length):** ตัดสตริง HTML ที่ยาวเกินตามความยาวที่ระบุ
- `{{ my_html|truncatechars_html:"10" }}`
49. **truncatewords(number):** ตัดสตริงที่มีจำนวนคำเกินที่ระบุ
- `{{ my_text|truncatewords:"5" }}`
50. **truncatewords_html(number):** ตัดสตริง HTML ที่มีจำนวนคำเกินที่ระบุ
- `{{ my_html|truncatewords_html:"5" }}`
51. **unordered_list:** แปลงลิสต์เป็นรายการแบบไม่มีลำดับ
- `{{ my_list|unordered_list }}`
52. **upper:** แปลงสตริงเป็นตัวพิมพ์ใหญ่
- `{{ "hello"|upper }}` → HELLO
53. **urlencode:** แปลงสตริงสำหรับใช้ใน URL
- `{{ "Hello World!"|urlencode }}` → Hello%20World%21
54. **urlize:** แปลง URL ในสตริงให้เป็นลิงก์ HTML
- `{{ my_text|urlize }}`
55. **urlizetrunc(length):** แปลง URL ในสตริงให้เป็นลิงก์ HTML พร้อมตัดข้อความที่ยาวเกิน
- `{{ my_text|urlizetrunc:"10" }}`
56. **wordcount:** นับจำนวนคำในสตริง
- `{{ my_text|wordcount }}`
57. **wordwrap(width):** จัดรูปแบบสตริงให้เป็นบรรทัดตามความกว้างที่ระบุ
- `{{ my_text|wordwrap:"80" }}`

58. yesno(yes, no, maybe): แปลงค่าความจริงเป็นสตริง

- `{{ True|yesno:"Yes,No" }}` → Yes

เข้าใจแล้วครับ! นี่คือตัวอย่างของการใช้ Django Template Filters พร้อมผลลัพธ์ที่เห็นชัดเจน โดยมีการเปรียบเทียบกับค่าต้นฉบับเพื่อให้เห็นความแตกต่าง:

1. add(value)

Template:

django

Copy code

Original: `{{ 4 }}`

With filter: `{{ 4|add:"2" }}`

Output:

sql

Copy code

Original: 4

With filter: 6

2. addslashes

Template:

django

Copy code

Original: {{ "It's me!" }}

With filter: {{ "It's me!"|addslashes }}

Output:

vbnet

Copy code

Original: It's me!

With filter: It\'s me!

3. capfirst

Template:

django

Copy code

Original: {{ "hello" }}

With filter: {{ "hello"|capfirst }}

Output:

sql

Copy code

Original: hello

With filter: Hello

4. center(width)

Template:

django

Copy code

Original: "{{ "Django" }}"

With filter: "{{ "Django"|center:"10" }}"

Output:

sql

Copy code

Original: "Django"

With filter: " Django "

5. cut(value)

Template:

django

Copy code

Original: {{ "Hello World" }}

With filter: {{ "Hello World"|cut:"o" }}

Output:

sql

Copy code

Original: Hello World

With filter: Hell Wrld

6. date(format)

Template:

django

Copy code

Original: {{ my_date }}

With filter: {{ my_date|date:"Y-m-d" }}

Output:

vbnet

Copy code

Original: 2024-08-26 (Assuming my_date is this date)

With filter: 2024-08-26

7. default(value)

Template:

django

Copy code

Original: {{ my_var }}

With filter: {{ my_var|default:"N/A" }}

Output:

mathematica

Copy code

Original: (Assuming my_var is None or "")

With filter: N/A

8. default_if_none(value)

Template:

django

Copy code

Original: {{ my_var }}

With filter: {{ my_var|default_if_none:"N/A" }}

Output:

mathematica

Copy code

Original: (Assuming my_var is None)

With filter: N/A

9. dictsort(key)

Template:

django

Copy code

Original: {{ my_dict }}

With filter: {{ my_dict|dictsort:"name" }}

Output:

css

Copy code

Original: [{'name': 'Bob'}, {'name': 'Alice'}, {'name': 'Charlie'}]

With filter: [{'name': 'Alice'}, {'name': 'Bob'}, {'name': 'Charlie'}]

10. divisibleby(number)

Template:

django

Copy code

Original: {{ 10 }}

With filter: {{ 10|divisibleby:"2" }}

Output:

sql

Copy code

Original: 10

With filter: True

11. escape

Template:

django

Copy code

Original: {{ "<div>" }}

With filter: {{ "<div>"|escape }}

Output:

php

Copy code

Original: <div>

With filter: <div>

12. escapejs

Template:

django

Copy code

Original: {{ "<script>" }}

With filter: {{ "<script>"|escapejs }}

Output:

php

Copy code

Original: <script>

With filter: \u003Cscript\u003E

13. filesizeformat

Template:

django

Copy code

Original: {{ 1024 }}

With filter: {{ 1024|filesizeformat }}

Output:

yaml

Copy code

Original: 1024

With filter: 1 KB

14. first

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|first }}

Output:

sql

Copy code

Original: [1, 2, 3]

With filter: 1

15. floatformat

Template:

django

Copy code

Original: {{ 123.456 }}

With filter: {{ 123.456|floatformat:"2" }}

Output:

sql

Copy code

Original: 123.456

With filter: 123.46

16. force_escape

Template:

django

Copy code

Original: {{ my_var }}

With filter: {{ my_var|force_escape }}

Output:

php

Copy code

Original: <div>

With filter: <div>

17. get_digit(position)

Template:

django

Copy code

Original: {{ 12345 }}

With filter: {{ 12345|get_digit:"2" }}

Output:

sql

Copy code

Original: 12345

With filter: 4

18. iriencode

Template:

django

Copy code

Original: {{ "http://example.com/?q=привет" }}

With filter: {{ "http://example.com/?q=привет"|iriencode }}

Output:

perl

Copy code

Original: http://example.com/?q=привет

With filter: http://example.com/?

q=%D0%BF%D1%80%D0%B8%D0%B2%D0%B5%D1%82

19. join(separator)

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|join:", " }}

Output:

sql

Copy code

Original: [1, 2, 3]

With filter: 1, 2, 3

20. json_script

Template:

django

Copy code

Original: {{ my_data }}

With filter: {{ my_data|json_script:"my-id" }}

Output:

bash

Copy code

Original: {'key': 'value'}

With filter: <script id="my-id" type="application/json">{"key": "value"}</script>

21. last

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|last }}

Output:

sql

Copy code

Original: [1, 2, 3]

With filter: 3

22. length

Template:

django

Copy code

Original: `{{ my_list }}`

With filter: `{{ my_list|length }}`

Output:

sql

Copy code

Original: `[1, 2, 3]`

With filter: 3

23. length_is(value)

Template:

django

Copy code

Original: `{{ my_list }}`

With filter: `{{ my_list|length_is:"3" }}`

Output:

vbnet

Copy code

Original: `[1, 2, 3]`

With filter: True (if length is 3)

24. linebreaks

Template:

django

Copy code

Original: {{ my_text }}

With filter: {{ my_text|linebreaks }}

Output:

javascript

Copy code

Original: Line1\nLine2

With filter: <p>Line1</p><p>Line2</p>

25. linebreaksbr

Template:

django

Copy code

Original: {{ my_text }}

With filter: {{ my_text|linebreaksbr }}

Output:

sql

Copy code

Original: Line1\nLine2

With filter: Line1
Line2

26. linenumbers

Template:

django

Copy code

Original: {{ my_text }}

With filter: {{ my_text|linenumbers }}

Output:

less

Copy code

Original: Line1\nLine2

With filter: 1: Line1\n2: Line2

27. ljust(width)

Template:

django

Copy code

Original: {{ "Django" }}

With filter: {{ "Django"|ljust:"10" }}

Output:

sql

Copy code

Original: Django

With filter: Django

28. lower

Template:

django

Copy code

Original: {{ "HELLO" }}

With filter: {{ "HELLO"|lower }}

Output:

sql

Copy code

Original: HELLO

With filter: hello

29. make_list

Template:

django

Copy code

Original: {{ "abc" }}

With filter: {{ "abc"|make_list }}

Output:

sql

Copy code

Original: abc

With filter: ['a', 'b', 'c']

30. phone2numeric

Template:

django

Copy code

Original: {{ "800-DJANGO" }}

With filter: {{ "800-DJANGO"|phone2numeric }}

Output:

sql

Copy code

Original: 800-DJANGO

With filter: 800-352646

31. pluralize

Template:

django

Copy code

Original: {{ 1 }}

With filter: {{ 1|pluralize:"s" }}

Output:

sql

Copy code

Original: 1

With filter: (empty string)

Original: {{ 2|pluralize:"s" }}

With filter: s

32. pprint

Template:

django

Copy code

Original: {{ my_data }}

With filter: {{ my_data|pprint }}

Output:

css

Copy code

Original: {'key': 'value'}

With filter: (pretty-printed JSON)

33. random

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|random }}

Output:

sql

Copy code

Original: [1, 2, 3]

With filter: (randomly selected value, e.g., 2)

34. rjust(width)

Template:

django

Copy code

Original: {{ "Django" }}

With filter: {{ "Django"|rjust:"10" }}

Output:

sql

Copy code

Original: Django

With filter: Django

35. safe

Template:

django

Copy code

Original: {{ "<div>" }}

With filter: {{ "<div>"|safe }}

Output:

php

Copy code

Original: <div>

With filter: <div>

36. safeseq

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|safeseq }}

Output:

vbnet

Copy code

Original: [1, 2, 3]

With filter: [1, 2, 3] (treated as safe)

37. slice(start:end)

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|slice:"1:4" }}

Output:

sql

Copy code

Original: [1, 2, 3, 4, 5]

With filter: [2, 3, 4]

38. slugify

Template:

django

Copy code

Original: {{ "Hello World!" }}

With filter: {{ "Hello World!"|slugify }}

Output:

sql

Copy code

Original: Hello World!

With filter: hello-world

39. stringformat(format)

Template:

django

Copy code

Original: `{{ 123 }}`

With filter: `{{ 123|stringformat:"04d" }}`

Output:

yaml

Copy code

Original: 123

With filter: 0123

40. striptags

Template:

django

Copy code

Original: `{{ "<div>Hello</div>" }}`

With filter: `{{ "<div>Hello</div>"|striptags }}`

Output:

javascript

Copy code

Original: `<div>Hello</div>`

With filter: Hello

41. time(format)

Template:

django

Copy code

Original: {{ my_time }}

With filter: {{ my_time|time:"H:i" }}

Output:

sql

Copy code

Original: 14:30:00

With filter: 14:30

42. timesince

Template:

django

Copy code

Original: {{ my_date }}

With filter: {{ my_date|timesince }}

Output:

sql

Copy code

Original: 2024-08-26

With filter: (time since date)

43. timeuntil

Template:

django

Copy code

Original: {{ my_date }}

With filter: {{ my_date|timeuntil }}

Output:

sql

Copy code

Original: 2024-08-26

With filter: (time until date)

44. title

Template:

django

Copy code

Original: {{ "hello world" }}

With filter: {{ "hello world"|title }}

Output:

sql

Copy code

Original: hello world

With filter: Hello World

45. truncatechars(length)

Template:

django

Copy code

Original: {{ my_text }}

With filter: {{ my_text|truncatechars:"10" }}

Output:

vbnet

Copy code

Original: This is a long text.

With filter: This is a...

46. truncatechars_html(length)

Template:

django

Copy code

Original: {{ my_html }}

With filter: {{ my_html|truncatechars_html:"10" }}

Output:

vbnet

Copy code

Original: <p>This is a long text.</p>

With filter: <p>This is a...</p>

47. truncatewords(number)

Template:

django

Copy code

Original: {{ my_text }}

With filter: {{ my_text|truncatewords:"5" }}

Output:

vbnet

Copy code

Original: This is a long text with many words.

With filter: This is a long text with...

48. truncatewords_html(number)

Template:

django

Copy code

Original: {{ my_html }}

With filter: {{ my_html|truncatewords_html:"5" }}

Output:

vbnet

Copy code

Original: <p>This is a long text with many words.</p>

With filter: <p>This is a long text with...</p>

49. unordered_list

Template:

django

Copy code

Original: {{ my_list }}

With filter: {{ my_list|unordered_list }}

Output:

javascript

Copy code

Original: [1, 2, 3]

With filter: 123

50. upper

Template:

django

Copy code

Original: {{ "hello" }}

With filter: {{ "hello"|upper }}

Output:

sql

Copy code

Original: hello

With filter: HELLO

51. urlencode

Template:

django

Copy code

Original: {{ "Hello World!" }}

With filter: {{ "Hello World!"|urlencode }}

Output:

perl

Copy code

Original: Hello World!

With filter: Hello%20World%21

52. urlize

Template:

django

Copy code

Original: {{ "Check this http://example.com" }}

With filter: {{ "Check this http://example.com"|urlize }}

Output:

kotlin

Copy code

Original: Check this http://example.com

With filter: Check this http://example.com

53. urlizetrunc(length)

Template:

django

Copy code

Original: {{ "Check this http://example.com" }}

With filter: {{ "Check this http://example.com"|urlizetrunc:"20" }}

Output:

kotlin

Copy code

Original: Check this http://example.com

With filter: Check this http://example.com

54. wordcount

Template:

django

Copy code

Original: {{ "This is a sentence with seven words." }}

With filter: {{ "This is a sentence with seven words."|wordcount }}

Output:

vbnet

Copy code

Original: This is a sentence with seven words.

With filter: 7

55. wordwrap(width)

Template:

django

Copy code

Original: {{ "This is a long text that needs to be wrapped to a certain width." }}

With filter: {{ "This is a long text that needs to be wrapped to a certain width."|wordwrap:"20" }}

Output:

vbnet

Copy code

Original: This is a long text that needs to be wrapped to a certain width.

With filter: This is a long text

that needs to be wrapped

to a certain width.

56. yesno(yes, no, maybe)

Template:

django

Copy code

Original: {{ True }}

With filter: {{ True|yesno:"Yes,No" }}

Output:

yaml

Copy code

Original: True

With filter: Yes

Template:

django

Copy code

Original: {{ False }}

With filter: {{ False|yesno:"Yes,No" }}

Output:

yaml

Copy code

Original: False

With filter: No

Template:

django

Copy code

Original: {{ None }}

With filter: {{ None|yesno:"Yes,No,Maybe" }}

Output:

sql

Copy code

Original: None

With filter: Maybe

หวังว่าตัวอย่างเหล่านี้จะช่วยให้คุณเห็นการทำงานของแต่ละ Filter ชัดเจนขึ้นครับ!