

sameDerivative

Checks student submission and supplied problem to see if they have the same derivative

The Problem

There are lots of reasons we would find it much easier to test derivatives, rather than functions themselves. For example, if you want to see if a student supplied a correct antiderivative can be tricky, since there are infinitely many options for an antiderivative. So this validator checks that the derivatives of the student's answer and the author's answer line up.

Potential Pitfalls and Problems

Letters like Arbitrary Constants are treated like Variables

In order to build a validator that is, somehow, indifferent to variable, this validator actually checks all lower case letters as if they are variables to make sure that it matches the provided answer. This can also be nice to make sure that students don't randomly change variables (e.g. default to "x" rather than keep it in terms of the variable given). It also checks the "variable" C , to make the classic $+C$ testable. Note however, that this checks the derivatives as if the C was a variable, so if the student enters something like $+2C$ and the instructor provided $+C$, then the derivatives won't match ($2 \neq 1$) and thus the answer will be rejected. This may be desirable behaviors in some classes, and not in others.

Example

```

1 // NOTE: The below are intended to be used inside an \answer optional argument with the validator key, NOT in
2
3 // sameDerivative checks to see if the derivative with respect to x and c are equal.
4 // Because of how they are loaded, I need to currently manually check each variable letter, so we do a full c
5 // Currently not checking e, f, g, h, or i as variables because they are special reserve letters in math.
6 sameDerivative = function(a,b) {
7     return (
8         a.derivative('a').equals( b.derivative('a') ) &&
9         a.derivative('b').equals( b.derivative('b') ) &&
10        a.derivative('c').equals( b.derivative('c') ) &&
11        a.derivative('d').equals( b.derivative('d') ) &&
12        a.derivative('j').equals( b.derivative('j') ) &&
13        a.derivative('k').equals( b.derivative('k') ) &&
14        a.derivative('l').equals( b.derivative('l') ) &&
15        a.derivative('m').equals( b.derivative('m') ) &&
16        a.derivative('n').equals( b.derivative('n') ) &&
17        a.derivative('o').equals( b.derivative('o') ) &&
18        a.derivative('p').equals( b.derivative('p') ) &&
19        a.derivative('q').equals( b.derivative('q') ) &&
20        a.derivative('r').equals( b.derivative('r') ) &&
21        a.derivative('s').equals( b.derivative('s') ) &&
22        a.derivative('t').equals( b.derivative('t') ) &&
23        a.derivative('u').equals( b.derivative('u') ) &&
24        a.derivative('v').equals( b.derivative('v') ) &&
25        a.derivative('w').equals( b.derivative('w') ) &&

```

Learning outcomes:

```

26     a.derivative('x').equals( b.derivative('x') ) &&
27     a.derivative('y').equals( b.derivative('y') ) &&
28     a.derivative('z').equals( b.derivative('z') ) &&
29     a.derivative('C').equals( b.derivative('C') ) ) ;
30 };
31
32
33

```

Problem 1 This problem shows the 'same derivative' validator. It's intended to be used to test the result of an indefinite integral, so it requires the "+C" at the end, and (should) mark any answer wrong that doesn't have it. Note that the C is case-sensitive.

Enter in any answer whose derivative is $x^2 + \sin(x) - 3$ (and don't forget the +C, but notice you can also add random constants to it too).

$$\int x^2 + \sin(x) - 3dx = \boxed{\frac{1}{3}x^3 - \cos(x) - 3x + C}$$

Sidenote: You can avoid the problem of the +C if you don't put it in the expected answer box, which will then (correctly) mark an answer wrong that does include a +C. So in a sense you can force the student to include it, or not, with this validator. Although only checking the derivatives to match has obvious issues for non indefinite integrals.

The Code

```

\begin{javascript}
// NOTE: The below are intended to be used inside an \answer optional argument with the validator key, NOT in
// sameDerivative checks to see if the derivative with respect to x and c are equal.
// Because of how they are loaded, I need to currently manually check each variable letter, so we do a full c
// Currently not checking e, f, g, h, or i as variables because they are special reserve letters in math.
sameDerivative = function(a,b) {
    return (
        a.derivative('a').equals( b.derivative('a') ) &&
        a.derivative('b').equals( b.derivative('b') ) &&
        a.derivative('c').equals( b.derivative('c') ) &&
        a.derivative('d').equals( b.derivative('d') ) &&
        a.derivative('j').equals( b.derivative('j') ) &&
        a.derivative('k').equals( b.derivative('k') ) &&
        a.derivative('l').equals( b.derivative('l') ) &&
        a.derivative('m').equals( b.derivative('m') ) &&
        a.derivative('n').equals( b.derivative('n') ) &&
        a.derivative('o').equals( b.derivative('o') ) &&
        a.derivative('p').equals( b.derivative('p') ) &&
        a.derivative('q').equals( b.derivative('q') ) &&
        a.derivative('r').equals( b.derivative('r') ) &&
        a.derivative('s').equals( b.derivative('s') ) &&
        a.derivative('t').equals( b.derivative('t') ) &&
        a.derivative('u').equals( b.derivative('u') ) &&
        a.derivative('v').equals( b.derivative('v') ) &&
        a.derivative('w').equals( b.derivative('w') ) &&
        a.derivative('x').equals( b.derivative('x') ) &&
        a.derivative('y').equals( b.derivative('y') ) &&
        a.derivative('z').equals( b.derivative('z') ) &&
        a.derivative('C').equals( b.derivative('C') ) ) ;
};

```

```
\end{javascript}
```

```
%%%
```

```
\begin{problem}
```

This problem shows the 'same derivative' validator. It's intended to be used to test the result of an ind

Enter in any answer whose derivative is $x^2 + \sin(x) - 3$ (and don't forget the $+C$, but notice you ca

```
\[
```

```
\int x^2 + \sin(x) - 3 dx = \answer[validator=sameDerivative]{\frac{1}{3}x^3 - \cos(x) - 3x + C}
```

```
\]
```

Sidenote: You can avoid the problem of the $+C$ if you don't put it in the expected answer box, which wil

```
\end{problem}
```