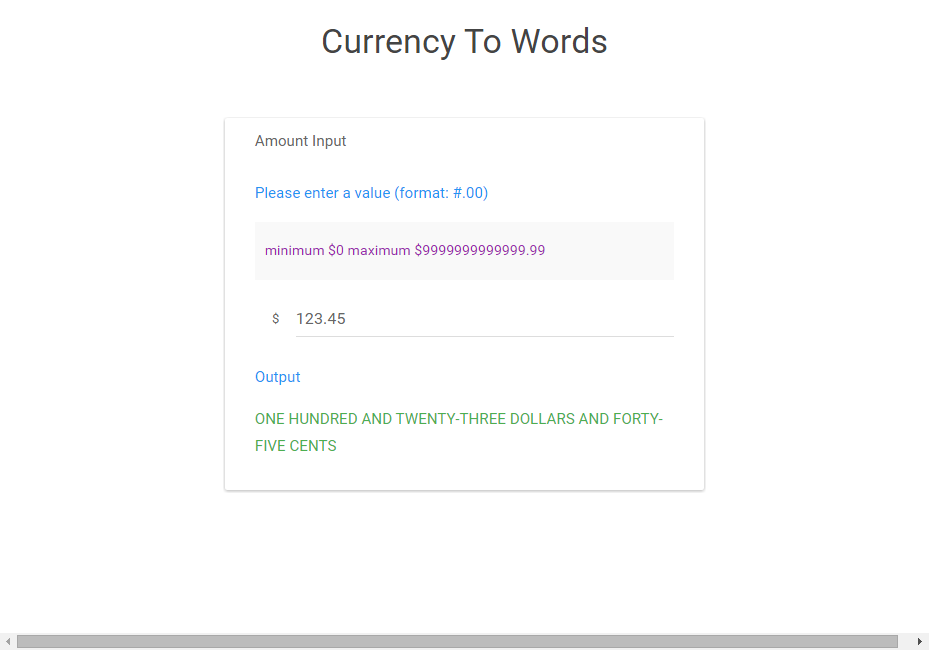
# **Test your skills: Number to Words Web Page**

## **Question**

Input: “123.45”

Output: “ONE HUNDRED AND TWENTY-THREE DOLLARS AND FORTY-FIVE CENTS”

## **Interface Overview**



A textbox for user input. Upon entering a valid amount ($0.00 to $9999999999999.99), it will output words for the entered amount.

## **Discussion**

This is a two-parted project. The first part is the backend of the website that accepts input from the web application and the latter is an independent frontend website that is not generated by the backend. The plural forms of ‘dollar’ and ‘cent’ are used when the amount (for dollar or cent) is greater than 1.

### Algorithm used to generate words for number (amount)

A service gets the value parsed in and split in into an array. For both dollar and cent values, it analyses the magnitude of the number and generate words for it using a dictionary that contains unique number words (e.g. one, two, ... ,nineteen, etc.).

#### Other method 1:

A dictionary that contains all the words for all the amount conceivable. This is not practical.

#### Other method 2:

Recursion method can be used and can be one of the optimal solutions. I didn’t use it because the idea of this solution came from the web (<http://stackoverflow.com/questions/2729752/converting-numbers-in-to-words-c-sharp>), so I came up with my original way of tackling the question.

### Backend

WEBAPI2 is used as the backend of the website. It acts as a rest-ful server the serves the frontend with the service of amount to words conversion.

Reason: The backend for this website only has a sole purpose of converting numeric amount into word strings. WEBAPI2 meets the requirements of the website.

Other possible solution:

MVC5 was a candidate for backend server but since the backend does not have the purpose of serving webpages, WEBAPI2 was chosen for its suitability in this scenario.

### Frontend

The frontend is an independent component that can be hosted separately to the backend. It uses AngularJs for data-binding and http operations and Bootstrap for styling.

Reason: The front-end is separated from the backend in order for a modular frontend and backend application. Failing of any component doesn’t cause the whole application to stop functioning.

Other possible solution:

MVC5 can be adopted for generating frontend view. However, a separated website is implemented so modularization purposes.

## **Test Plan**

### Success Test

|  |  |  |  |
| --- | --- | --- | --- |
| Value | Expected output | Actual output | status |
| 0 | ZERO DOLLAR | ZERO DOLLAR | success |
| 1 | ONE DOLLAR | ONE DOLLAR | success |
| 111 | ONE HUNDRED AND ELEVEN DOLLARS | ONE HUNDRED AND ELEVEN DOLLARS | success |
| .01 | ONE CENT | ONE CENT | success |
| 7.1 | SEVEN DOLLARS AND TEN CENTS | SEVEN DOLLARS AND TEN CENTS | success |
| 7.00 | SEVEN DOLLARS | SEVEN DOLLARS | success |
| .97 | NINETY-SEVEN CENTS | NINETY-SEVEN CENTS | success |
| 12889211.07 | TWELVE MILLION AND EIGHT HUNDRED AND EIGHTY-NINE THOUSAND AND TWO HUNDRED AND ELEVEN DOLLARS AND SEVEN CENTS | TWELVE MILLION AND EIGHT HUNDRED AND EIGHTY-NINE THOUSAND AND TWO HUNDRED AND ELEVEN DOLLARS AND SEVEN CENTS | success |
| 100000001.01 | ONE HUNDRED MILLION AND ONE DOLLARS AND ONE CENT | ONE HUNDRED MILLION AND ONE DOLLARS AND ONE CENT | success |
| 9999999999999.99 | NINE TRILLION AND NINE HUNDRED AND NINETY-NINE BILLION AND NINE HUNDRED AND NINETY-NINE MILLION AND NINE HUNDRED AND NINETY-NINE THOUSAND AND NINE HUNDRED AND NINETY-NINE DOLLARS AND NINETY-NINE CENTS | NINE TRILLION AND NINE HUNDRED AND NINETY-NINE BILLION AND NINE HUNDRED AND NINETY-NINE MILLION AND NINE HUNDRED AND NINETY-NINE THOUSAND AND NINE HUNDRED AND NINETY-NINE DOLLARS AND NINETY-NINE CENTS | success |
| 123.45 | ONE HUNDRED AND TWENTY-THREE DOLLARS AND FORTY-FIVE CENTS | ONE HUNDRED AND TWENTY-THREE DOLLARS AND FORTY-FIVE CENTS | success |
|  |  |  |  |
|  |  |  |  |

### Fail Test

|  |  |  |  |
| --- | --- | --- | --- |
| Value | Expected output | Actual output | status |
| ABC | Not a valid input | Not a valid input | success |
| abc | Not a valid input | Not a valid input | success |
| 1.111 | Not a valid input | Not a valid input | success |
| 0.117 | Not a valid input | Not a valid input | success |
| .198 | Not a valid input | Not a valid input | success |
| 1ac | Not a valid input | Not a valid input | success |
| ac2 | Not a valid input | Not a valid input | success |
| 101.1.1 | Not a valid input | Not a valid input | success |
| 11111222223333.23 | Not a valid input | Not a valid input | success |
|  |  |  |  |
|  |  |  |  |

A unit test project is included to test the service as well.