

# DxGetTextSize

This function retrieves the theoretical width and height (in pixels) of a certain piece of text, if it were to be drawn using dxDrawText.

**NOTE:** This function already takes the client's screen resolution into account.

## Syntax

```
float, float dxGetTextSize ( string text [, float width = 0, float scaleX = 1.0 [, float scaleY = 1.0 ], mixed font = "default", bool wordBreak = false, bool colorCoded = false] )
```

**OOP Syntax** Help! I don't understand this!

**Note:** *This syntax requires you to ignore the font argument above*

**Method:** *font:getSize(...)*

## Required Arguments

- **text:** A string representing the text for which you wish to retrieve with width for.

## Optional Arguments

*NOTE:* When using optional arguments, you might need to supply all arguments before the one you wish to use. For more information on optional arguments, see optional arguments.

- **width:** The width of the text. Use with *wordBreak = true*.
- **scaleX:** The scale of the text. Scale can also be inputted as a Vector2.
- **scaleY:** The scale of the text.
- **font:** Either a custom DX font element or the name of a built-in dx font:
  - **"default":** Tahoma
  - **"default-bold":** Tahoma Bold
  - **"clear":** Verdana
  - **"arial":** Arial
  - **"sans":** Microsoft Sans Serif
  - **"pricedown":** Pricedown (GTA's theme text)
  - **"bankgothic":** Bank Gothic Medium
  - **"diploma":** Diploma Regular
  - **"beckett":** Beckett Regular
  - **"unifont":** Unifont
- **wordBreak:** If set to *true*, the text will wrap to a new line whenever it reaches the right side of the bounding box. If *false*, the text will always be completely on one line.
- **colorCoded:** Should we exclude color codes from the width? False will include the hex in the length.

## Returns

Returns two floats representing the width and height of the text in pixels.