



Repla is a macOS app to **see your code running live!**

#### Subscribe for Early Access

 

Imagine if **instead of constantly switching back and forth between editing and running your code**, you could **see the results of your code running live** as you write it? Repla is an application that shows a **live preview of your code running right beside your editor**, so you can tell if your program is working instantly.

**Repla is a REPL tool.** Programming with a REPL has a rich history (the shell itself is a form of REPL), but they've never reached their full potential. Traditionally they've required entering code at a plain command prompt, a difficult proposition for programmers today accustomed to "richer" editing environments. Repla changes all that by making the REPL accessible alongside how programmers are already the most comfortable working: Writing code in a text editor and checking it into version control.



#### Features

- Built-in support for **JavaScript, Python, and Ruby**; extensible to support other languages.
- OS-level file-system events to keep the preview up-to-date, so use Repla with **VS Code, Atom, Sublime Text, TextMate, BBEdit, Vim, Emacs**, or any other text editor.
- No configuration or special file formats, so Repla works seamlessly with existing projects, workflows, and version control systems, like **git, svn, and Mercurial**.

[Follow us on Twitter](#)

#### Languages

The below examples show what the live updating Repla window shows while the example source code is being edited in the text editor of your choice.

#### JavaScript

Your Text Editor	Repla
<pre>1 function addNumbers(a, b) { 2   return a + b; 3 } 4 5 var left = 1; 6 7 var right = 2; 8 9 addNumbers(left, right); 10</pre>	<pre>function addNumbers(a, b) {   return a + b; } =&gt; undefined var left = 1; =&gt; undefined var right = 2; =&gt; undefined addNumbers(left, right); =&gt; 3</pre>

#### Python

Your Text Editor	Repla
------------------	-------

**Python**

Your Text Editor	Repla
<pre>1 def addNumbers(a, b): 2     return a + b 3 4 left = 1 5 right = 2 6 addNumbers(1, 2) 7</pre>	<pre>def addNumbers(a, b):     return a + b  left = 1 right = 2 addNumbers(1, 2) =&gt; 3</pre>

**Ruby**

Your Text Editor	Repla
<pre>1 def add_numbers(a, b) 2     return a + b 3 end 4 5 left = 1 6 7 right = 2 8 9 add_numbers(left, right) 10</pre>	<pre>def add_numbers(a, b)     return a + b end =&gt; :add_numbers left = 1 =&gt; 1 right = 2 =&gt; 2 add_numbers(left, right) =&gt; 3</pre>

[Twitter](#)

[Email](#)