About
Bacione Javascript Editor is an Integrated Development Environment (IDE) for programming small javascript games on a phone. Several games are included.
License: GPL
Github: github.com/bacionejs
Tested on: Android 13, Chrome 121, tablet and phone

Download
The editor is one file including the games.
Download bacione.html, open your file manager, click bacione.html to run in your Chrome browser, click Run to play the Ski game.

Links
Games.pdf
YouTube
Github
Github discussions
Reddit discussions

Why Choose This IDE?
Discover the ultimate free IDE code editor designed specifically for programming small JavaScript games on your phone offline! This phone editor was created because many people don't have a computer, and other phone editors lack important features.

Bacione means "big KISS," as in "Keep It Satisfyingly Simple." It offers less clicks compared to other phone editors, with common keys accessible in one click and context-aware logic that reduces the need for repetitive actions. Refactoring code is easier, and you have a fully programmable keyboard where you can create your own icons and functions. Designed for phones and tablets, it requires no setup and operates as a single file that's easy to share. Written in plain vanilla JavaScript with no dependencies, it's free (GPL license), contains no ads, and runs directly from your Android file manager in the Chrome browser. The editor itself is only 7 kilobytes (zipped), and is easy to modify and customize right within the editor. Plus, it supports swipe and longpress-repeat for enhanced usability.

Features
* Less clicks compared to other phone editors.
* Common keys are one click.
* Context aware logic reduces clicks.
* Less work to refactor your code.
* Fully programmable keyboard.
* Create your own keyboard icons and functions.
* Designed for phones and tablets.

- No setup required.
- One file.
- Easy to share.
- Offline.
- Plain vanilla JavaScript.
- No dependencies.
- Free (GPL license).
- No ads.
- Run from Android file manager in Chrome browser.
- Easy to modify editor.
- Edit the editor in the editor.
- Editor is only 7 kilobytes (zipped).
- Supports swipe and long press-repeat.

Hello World
At the top of the editor, create this code and click Run:
```
function helloworld(){
  L.canvas(W/2).icon(127759,W/2,W/2);
}
```

Run
Runs the game which contains the cursor.
Also stops the game.

Save/Diff
Saves to a timestamped file.
The file contains your games AND the Editor.
Swipeup shows changes you have made.

Zoom/Fullscreen
Toggles font size.
Swipeup toggles fullscreen.

Complete
Completes a word.
Looks for match in current game and then elsewhere.

Comment
Toggles comments for the current line.
Also puts the line in Paste.

Copy
Selects text or extends selection and puts it in Paste.
Selection looks at the character to the right and then selects:
  newline: line
  word: left,right
  spaces: right
  blank line: next blank line
  blocks: (){}[]""

Paste
Pastes

Backspace
Deletes character or selection.

Arrows
Moves cursor or extends selection.

Enter
Inserts a new line or cancels selection.

Undo/Redo
Restore up to 50 changes.

Rename
Find/Replace

Swipeup
Swipeup on alphanumeric keys is like shift.

Longpress
Repeats key

Errors
Runtime errors display an error message and automatically jumps to the error in code.

Misc/Themes
Color-Picker and Icon-Picker put a code in Paste.
Edit-the-Editor enables changes to the editor source code, such as colors, new buttons, etc.
Swipeup toggles themes: light, dark, gray, retro

Global Variables and Aliases
FPS=(frames per second)
W=innerWidth;
floor=Math.floor;

Overridden
Array sort: compares numbers instead of strings.

Variable tracking
L.debug(string,canvas)

Random number
number=L.rnd(max,min)

Random biased sets
set=L.RandomArray(quantity,range)
set.next()

Sound
sound=L.Sound()
sound.rocket()

Canvas
```
canvas=L.canvas(fontsize,iconstart)
canvas.icon(emoji,x,y)
canvas.clear()
canvas.black()
canvas.white()
```

Color gradient
```
canvas.fillStyle=L.gradient(canvas,bottom,height,[[stop,color],...])
```

Shapes
```
canvas.fill(L.shape([[x,y,...],...],scale))
```

Pixel to Vector
```
canvas.fill(L.shape(L.vertices(pixels)))
```

Game Controls
```
Ski - Click anywhere.
Pairs - Click in squares.
Smash - Click the pumpkin.
Monkeymaze - Click quadrants.
Snake - Click vertical half if snake is horizontal, etc.
Chess - Click piece and destination. Longpress to change piece.
Marslander - Click on right or left side. To disable sound, comment
snd.rocket().
```

Buttons

▷ Run

↓ Save

A Zoom

⌕ Rename

◇ Misc

⬚ Help

⌟ Enter

⍉ Backspace

⤳ Redo

↩ Undo

⧉ Paste

❏ Copy

→ Right

← Left

// Comment

⇨ Complete

⌒ Capslock