A close-up photograph of a green printed circuit board (PCB). On the board, there is a large black integrated circuit (likely a microcontroller) with a grid of pins. Several other electronic components are visible, including a small yellow capacitor and a green component with the number '6' on it. The board features a grid of circular pads and green printed tracks. The background is dark, making the green board stand out.

Variáveis

É possível declarar variáveis de diferentes formas, utilizando **var**, **let** e **const**

Ciclo de vida de uma variável

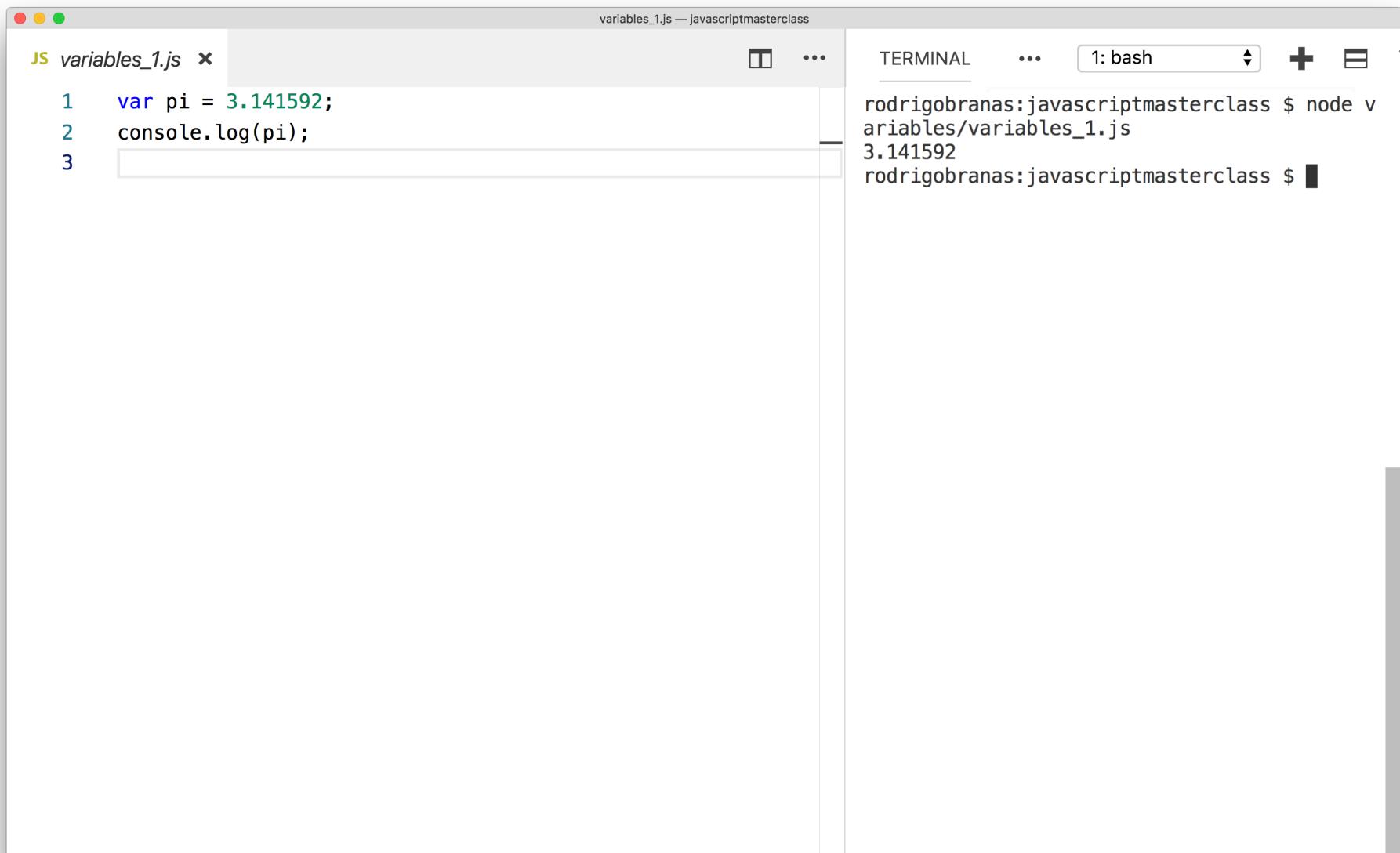
Declaração

Inicialização

Atribuição

- **Declaração:** O nome da variável é registrado no contexto de execução, também conhecido como escopo, da função
- **Inicialização:** A variável é inicializada com o valor `undefined`
- **Atribuição:** Um valor é atribuído para a variável

Ao utilizar var, a variável é declarada e inicializada no escopo da função, não respeitando bloco e permitindo a redeclaração e reatribuição



A screenshot of a Mac OS X desktop environment showing a terminal window. The window title is "variables_1.js — javascriptmasterclass". The terminal pane displays the output of running the JavaScript file "variables_1.js" with Node.js, showing the value "3.141592" printed to the console.

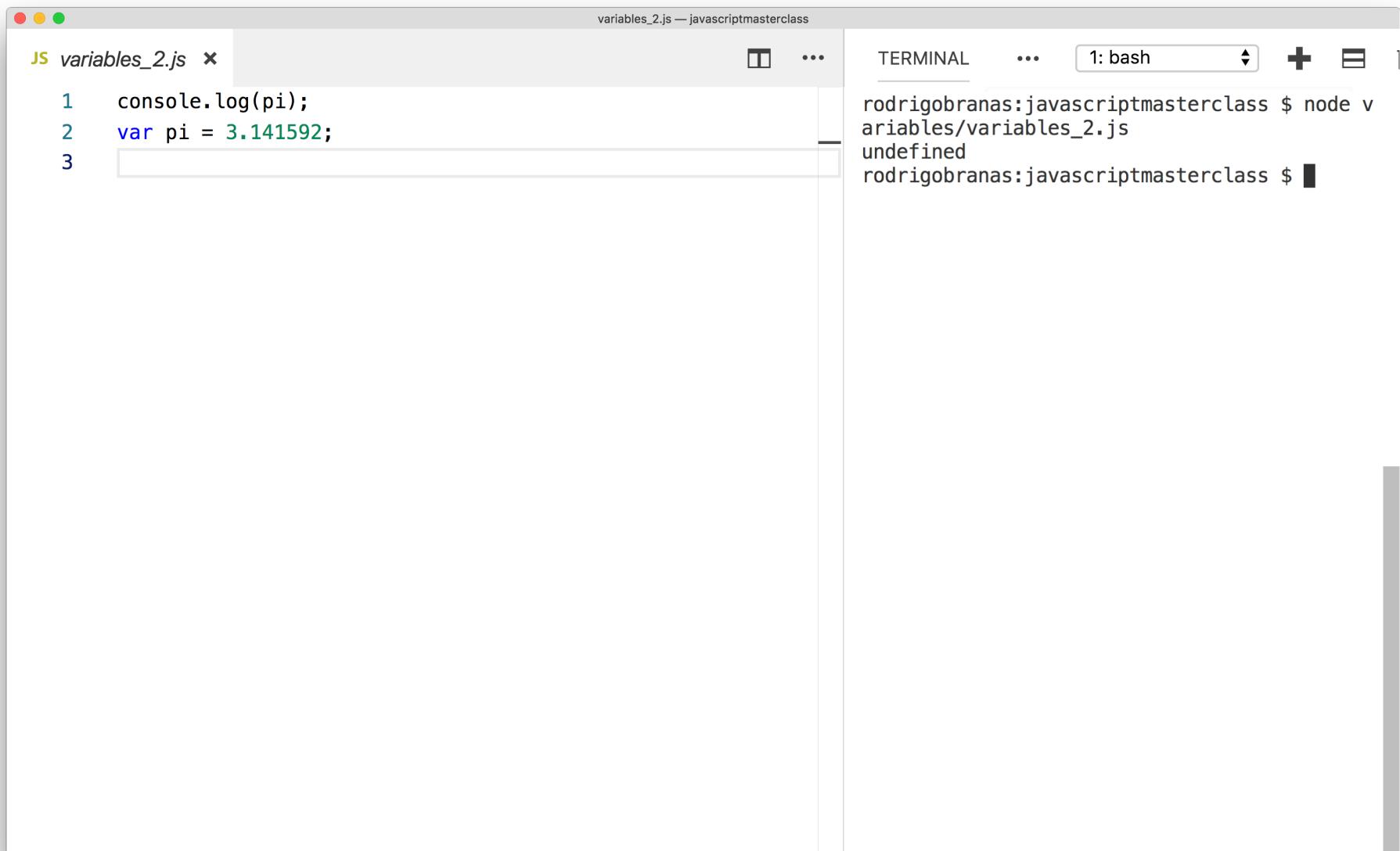
variables_1.js — javascriptmasterclass

JS variables_1.js x

```
1 var pi = 3.141592;
2 console.log(pi);
3
```

TERMINAL ... 1: bash

```
rodrigobranas:javascriptmasterclass $ node variables/variables_1.js
3.141592
rodrigobranas:javascriptmasterclass $
```



A screenshot of a macOS desktop environment showing a terminal window. The window title is "variables_2.js — javascriptmasterclass". The terminal tab is active, showing the command "node variables/variables_2.js" and its output: "undefined". The terminal interface includes standard macOS window controls (red, yellow, green buttons) and a dock with various icons.

variables_2.js — javascriptmasterclass

JS variables_2.js x

TERMINAL ... 1: bash + =

```
1 console.log(pi);
2 var pi = 3.141592;
3
```

```
rodrigobranas:javascriptmasterclass $ node variables/variables_2.js
undefined
rodrigobranas:javascriptmasterclass $
```

variables_3.js — javascriptmasterclass

JS variables_3.js x

```
1  var pi = 3.141592;
2  console.log(pi);
3  pi = 3;
4  console.log(pi);
5
```

TERMINAL ... 1: bash

```
rodrigobranas:javascriptmasterclass $ node variables/variables_3.js
3.141592
3
rodrigobranas:javascriptmasterclass $
```

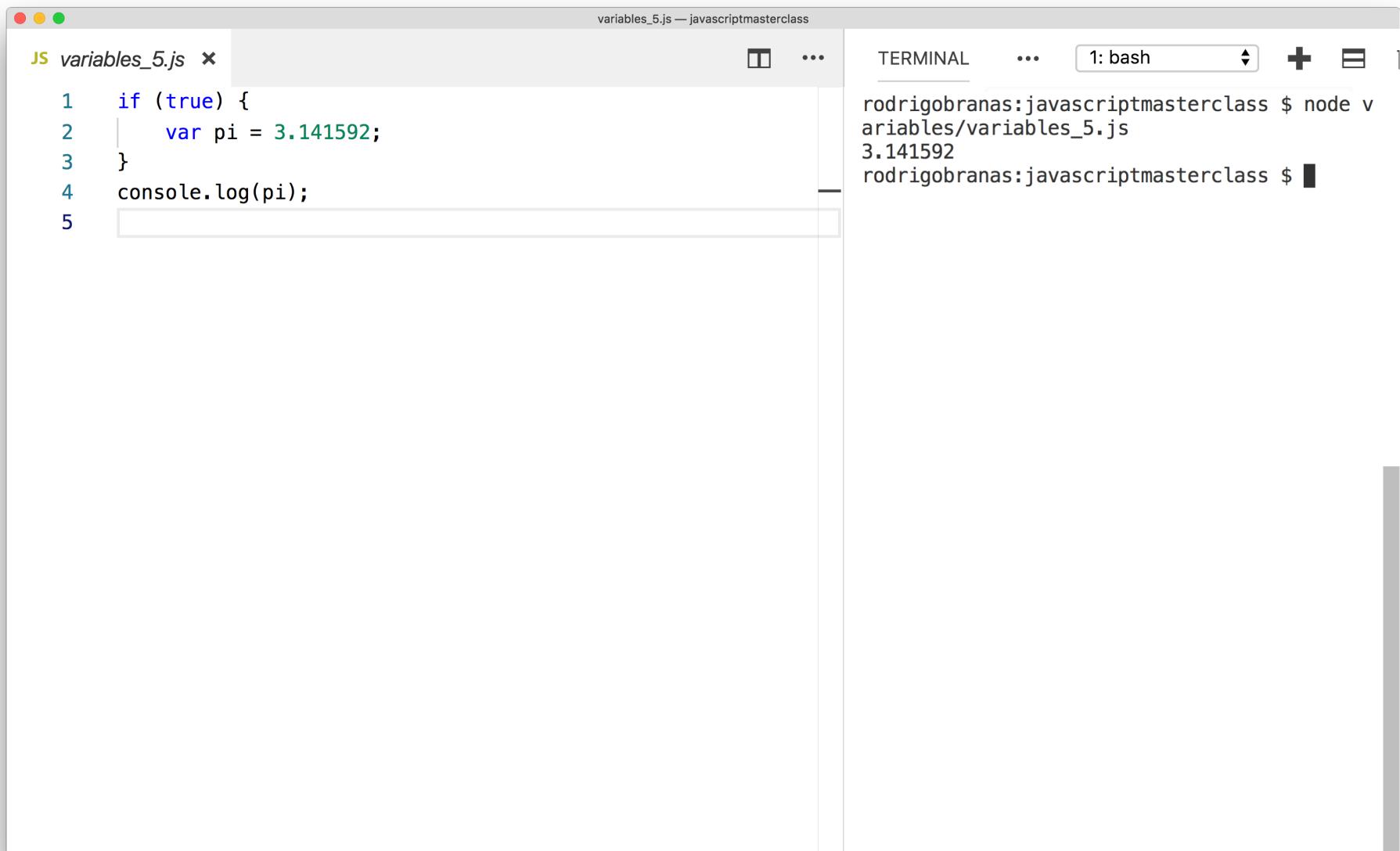
variables_4.js — javascriptmasterclass

JS variables_4.js x

```
1  var pi = 3.141592;
2  console.log(pi);
3  var pi = 3;
4  console.log(pi);
5
```

TERMINAL ... 1: bash

```
rodrigobranas:javascriptmasterclass $ node variables/variables_4.js
3.141592
3
rodrigobranas:javascriptmasterclass $
```



A screenshot of a macOS terminal window. The window title is "variables_5.js — javascriptmasterclass". The main pane on the left contains the following JavaScript code:

```
JS variables_5.js x
1  if (true) {
2    var pi = 3.141592;
3  }
4  console.log(pi);
5
```

The right pane, labeled "TERMINAL", shows the output of running the script with "node".

```
rodrigobranas:javascriptmasterclass $ node variables/variables_5.js
3.141592
rodrigobranas:javascriptmasterclass $
```

Ao utilizar let, a variável é declarada no escopo da função mas só é inicializada posteriormente, respeitando bloco e permitindo reatribuição mas não a redeclaração

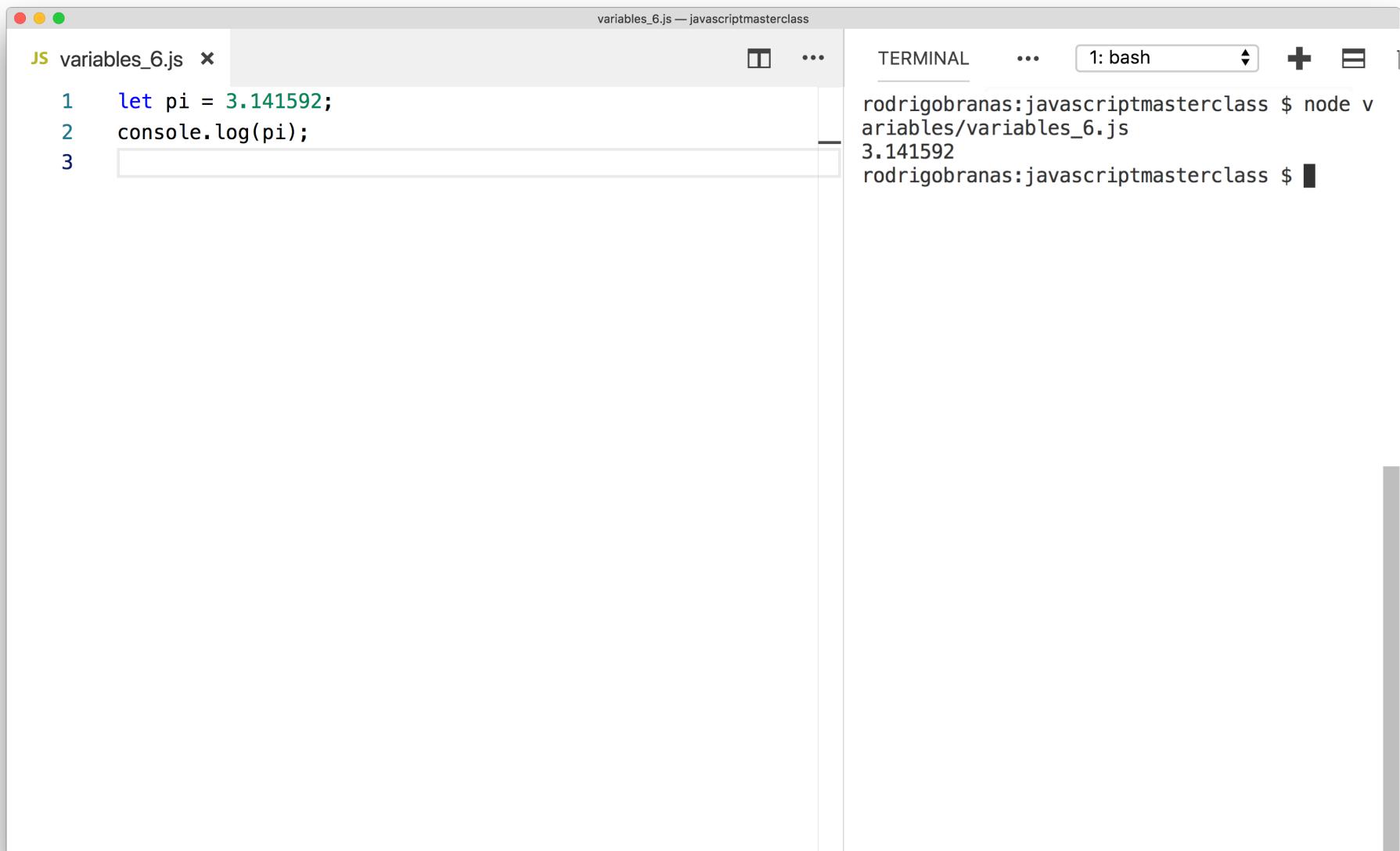
variables_6.js — javascriptmasterclass

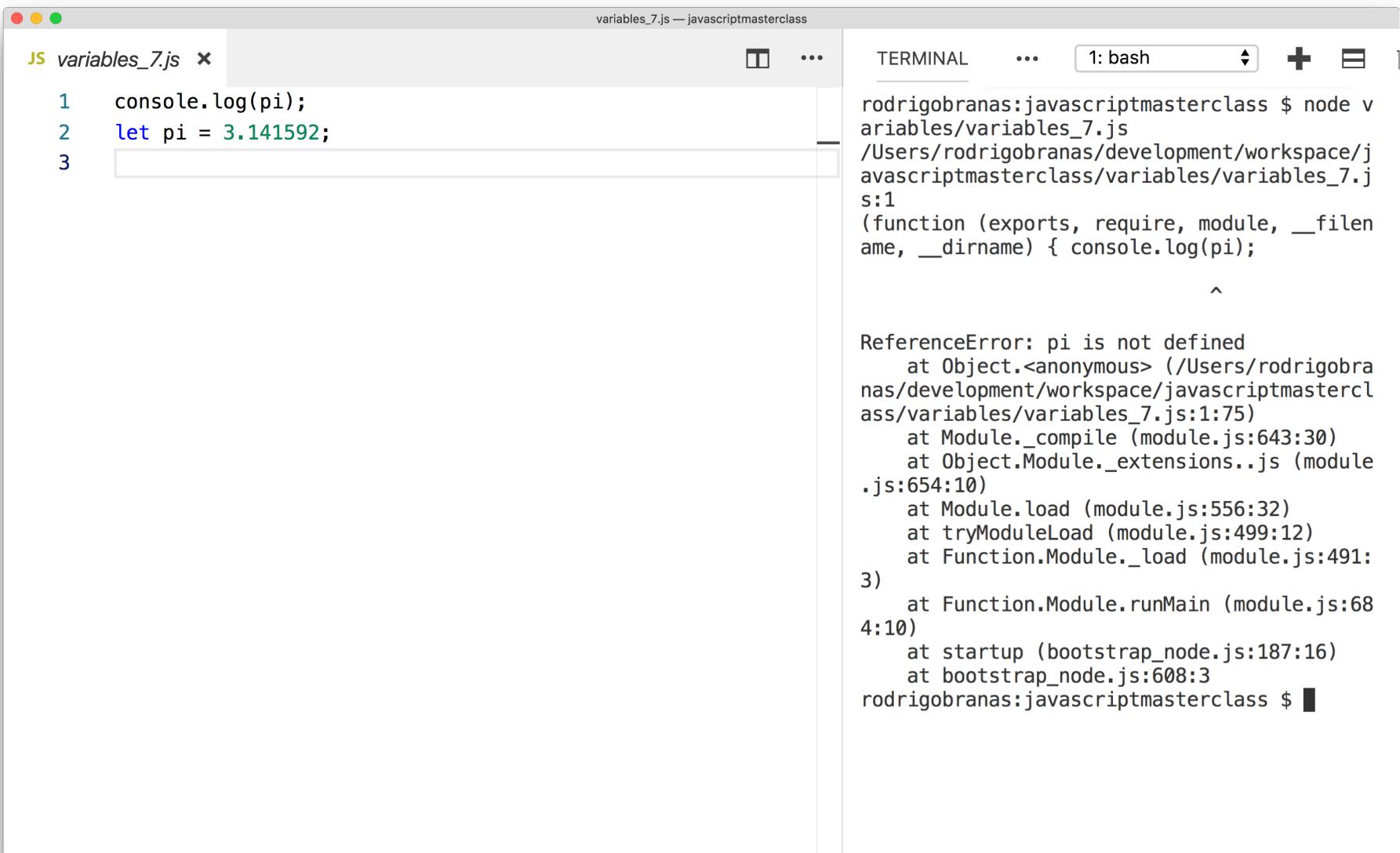
JS variables_6.js x

```
1 let pi = 3.141592;
2 console.log(pi);
3
```

TERMINAL ... 1: bash

```
rodrigobranas:javascriptmasterclass $ node variables/variables_6.js
3.141592
rodrigobranas:javascriptmasterclass $
```





A screenshot of a Mac OS X terminal window. The window title is "variables_7.js — javascriptmasterclass". The terminal tab is selected, showing the command "node variables/variables_7.js" and its output. The output shows a ReferenceError: pi is not defined, with a stack trace from module.js:643:30 to module.js:654:10, and ends with "rodrigobranas:javascriptmasterclass \$".

```
variables_7.js — javascriptmasterclass
TERMINAL 1: bash
rodrigobranas:javascriptmasterclass $ node variables/variables_7.js
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_7.js:1
(function (exports, require, module, __filename, __dirname) { console.log(pi);
^
ReferenceError: pi is not defined
    at Object.<anonymous> (/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_7.js:1:75)
    at Module._compile (module.js:643:30)
    at Object.Module._extensions..js (module.js:654:10)
    at Module.load (module.js:556:32)
    at tryModuleLoad (module.js:499:12)
    at Function.Module._load (module.js:491:3)
    at Function.Module.runMain (module.js:684:10)
    at startup (bootstrap_node.js:187:16)
    at bootstrap_node.js:608:3
rodrigobranas:javascriptmasterclass $
```

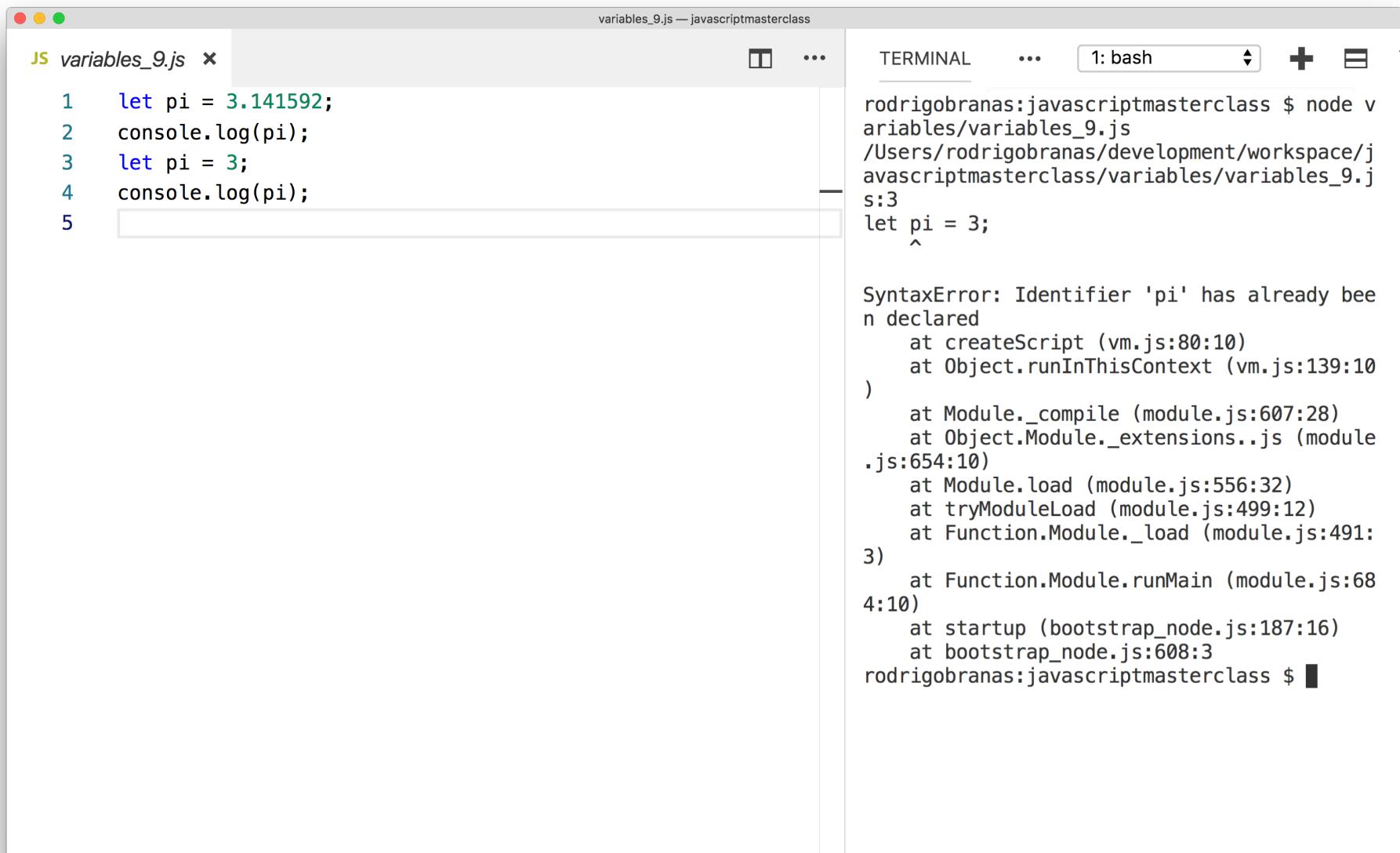
variables_8.js — javascriptmasterclass

JS variables_8.js x

```
1  let pi = 3.141592;
2  console.log(pi);
3  pi = 3;
4  console.log(pi);
5
```

TERMINAL ... 1: bash

```
rodrigobranas:javascriptmasterclass $ node variables/variables_8.js
3.141592
3
rodrigobranas:javascriptmasterclass $
```



A screenshot of a Mac OS X terminal window titled "variables_9.js — javascriptmasterclass". The window is split into two panes: a code editor on the left and a terminal on the right.

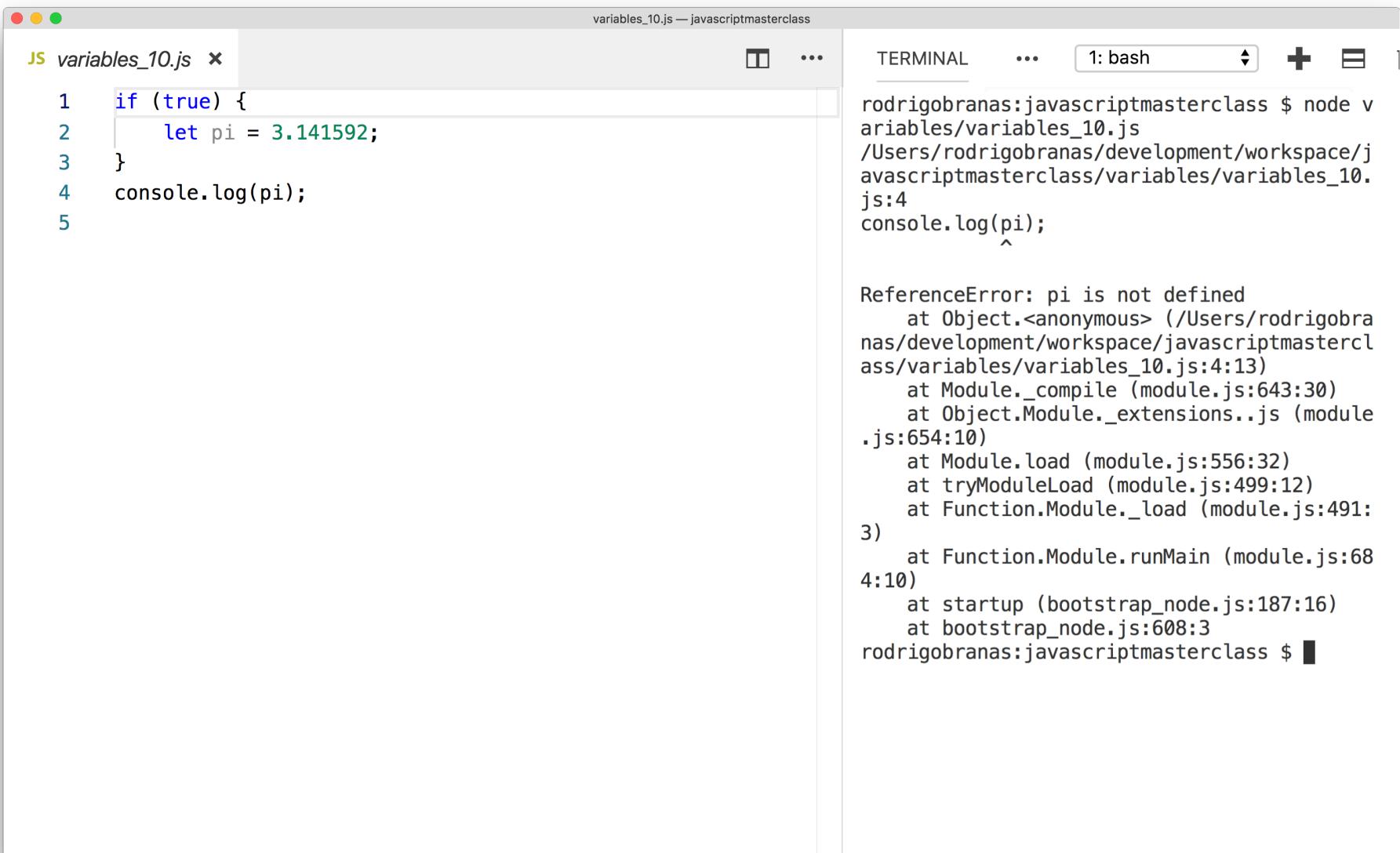
Code Editor (Left):

```
JS variables_9.js x
variables_9.js — javascriptmasterclass
1  let pi = 3.141592;
2  console.log(pi);
3  let pi = 3;
4  console.log(pi);
5
```

Terminal (Right):

```
rodrigobranas:javascriptmasterclass $ node variables/variables_9.js
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_9.js:3
let pi = 3;
^

SyntaxError: Identifier 'pi' has already been declared
  at createScript (vm.js:80:10)
  at Object.runInThisContext (vm.js:139:10)
)
  at Module._compile (module.js:607:28)
  at Object.Module._extensions..js (module.js:654:10)
  at Module.load (module.js:556:32)
  at tryModuleLoad (module.js:499:12)
  at Function.Module._load (module.js:491:3)
  at Function.Module.runMain (module.js:684:10)
  at startup (bootstrap_node.js:187:16)
  at bootstrap_node.js:608:3
rodrigobranas:javascriptmasterclass $
```



A screenshot of a Mac OS X terminal window. The window title is "variables_10.js — javascriptmasterclass". The left pane shows the code for "variables_10.js":

```
JS variables_10.js x
variables_10.js — javascriptmasterclass

1  if (true) {
2      let pi = 3.141592;
3  }
4  console.log(pi);
5
```

The right pane shows the terminal output:

```
TERMINAL ... 1: bash + = 1

rodrigobranas:javascriptmasterclass $ node variables/variables_10.js
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_10.js:4
console.log(pi);
^

ReferenceError: pi is not defined
    at Object.<anonymous> (/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_10.js:4:13)
        at Module._compile (module.js:643:30)
        at Object.Module._extensions..js (module.js:654:10)
        at Module.load (module.js:556:32)
        at tryModuleLoad (module.js:499:12)
        at Function.Module._load (module.js:491:3)
        at Function.Module.runMain (module.js:684:10)
        at startup (bootstrap_node.js:187:16)
        at bootstrap_node.js:608:3
rodrigobranas:javascriptmasterclass $
```

Ao utilizar `const`, a variável é declarada no escopo da função mas só é inicializada posteriormente, respeitando bloco e não permitindo reatribuição nem redeclaração

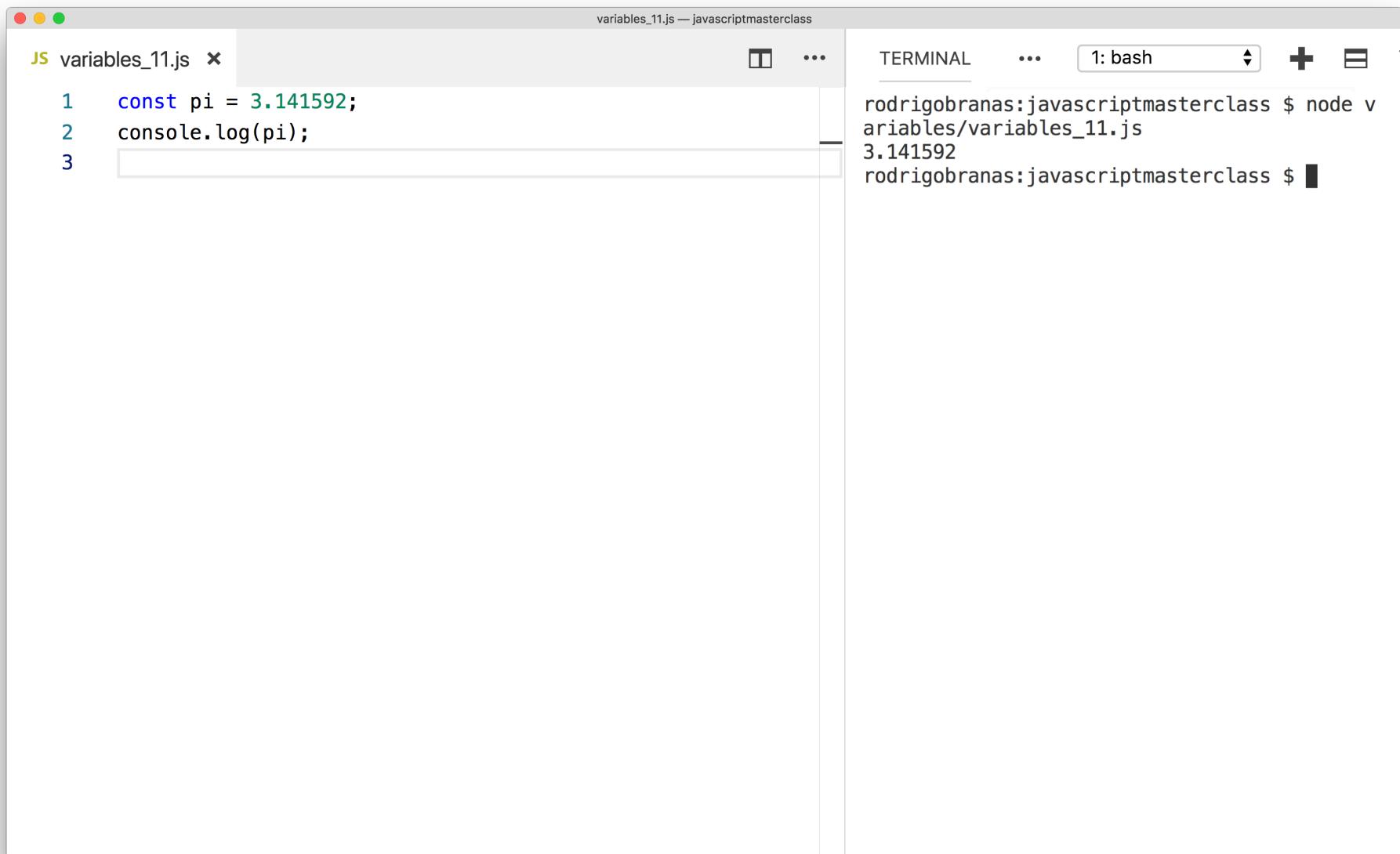
variables_11.js — javascriptmasterclass

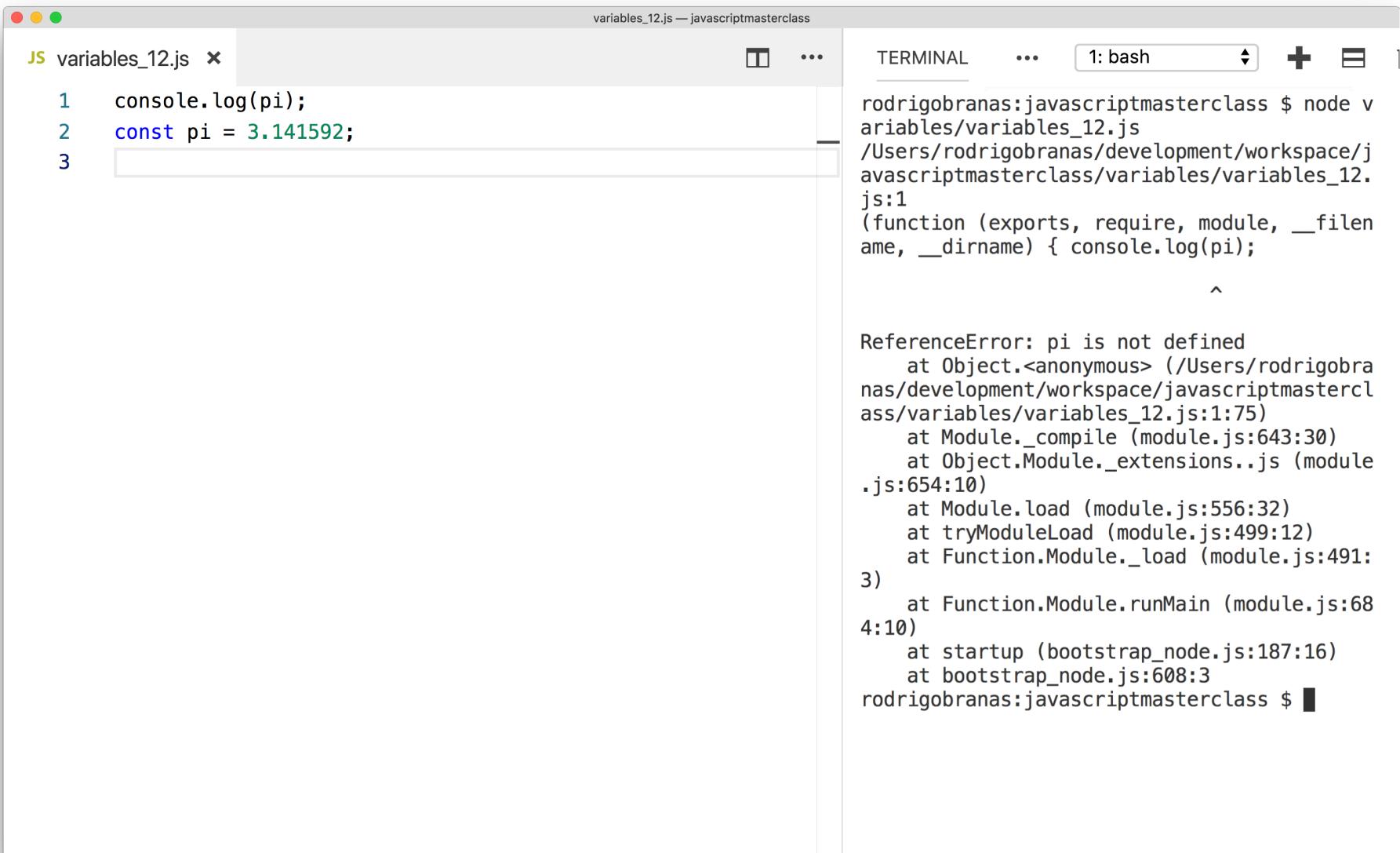
JS variables_11.js x

```
1 const pi = 3.141592;
2 console.log(pi);
3
```

TERMINAL ... 1: bash

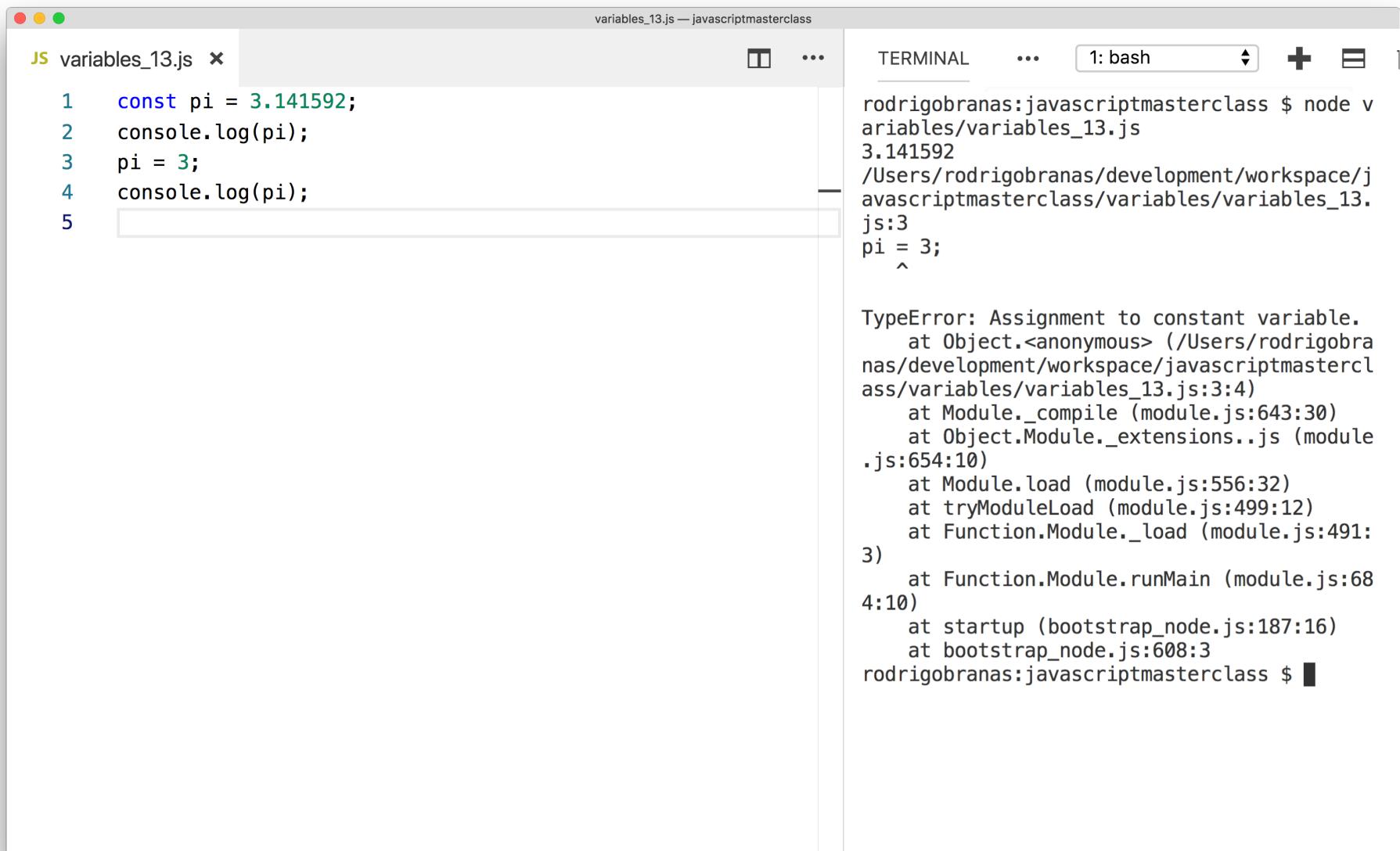
```
rodrigobranas:javascriptmasterclass $ node variables/variables_11.js
3.141592
rodrigobranas:javascriptmasterclass $
```





A screenshot of a Mac OS X terminal window. The window title is "variables_12.js — javascriptmasterclass". The terminal tab is selected, showing the command "node variables/variables_12.js" and the resulting error output. The error message is "ReferenceError: pi is not defined" with a stack trace. The terminal window has a dark theme with light text and a light background.

```
variables_12.js — javascriptmasterclass
TERMINAL 1: bash
rodrigobranas:javascriptmasterclass $ node variables/variables_12.js
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_12.js:1
(function (exports, require, module, __filename, __dirname) { console.log(pi);
^
ReferenceError: pi is not defined
    at Object.<anonymous> (/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_12.js:1:75)
    at Module._compile (module.js:643:30)
    at Object.Module._extensions..js (module.js:654:10)
    at Module.load (module.js:556:32)
    at tryModuleLoad (module.js:499:12)
    at Function.Module._load (module.js:491:3)
    at Function.Module.runMain (module.js:684:10)
    at startup (bootstrap_node.js:187:16)
    at bootstrap_node.js:608:3
rodrigobranas:javascriptmasterclass $
```



A screenshot of a Mac OS X terminal window titled "variables_13.js — javascriptmasterclass". The window is split into two panes: a code editor on the left and a terminal on the right.

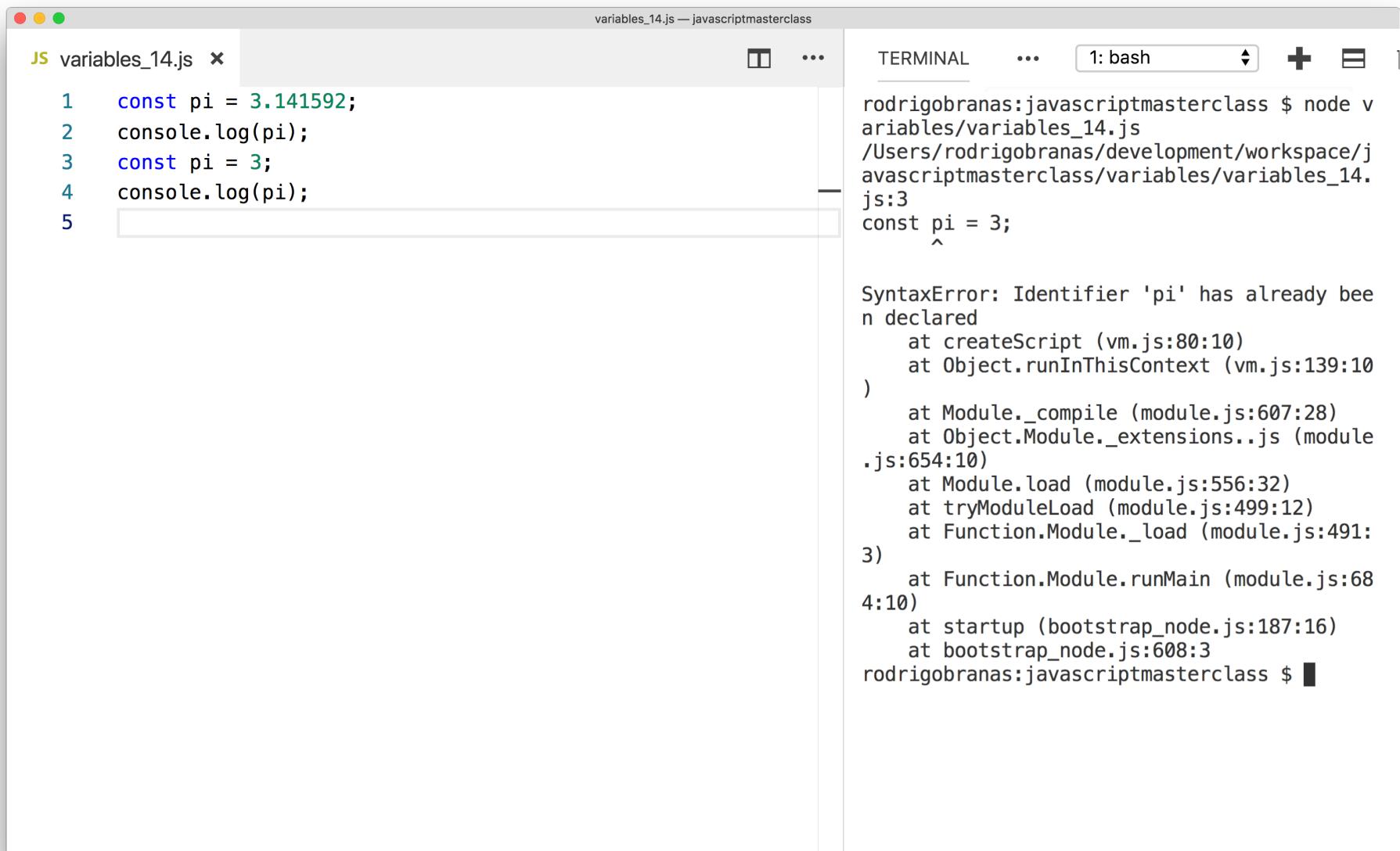
Code Editor (Left):

```
JS variables_13.js ×
1  const pi = 3.141592;
2  console.log(pi);
3  pi = 3;
4  console.log(pi);
5
```

Terminal (Right):

```
rodrigobranas:javascriptmasterclass $ node variables/variables_13.js
3.141592
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_13.js:3
pi = 3;
^

TypeError: Assignment to constant variable.
    at Object.<anonymous> (/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_13.js:3:4)
    at Module._compile (module.js:643:30)
    at Object.Module._extensions..js (module.js:654:10)
    at Module.load (module.js:556:32)
    at tryModuleLoad (module.js:499:12)
    at Function.Module._load (module.js:491:3)
    at Function.Module.runMain (module.js:684:10)
    at startup (bootstrap_node.js:187:16)
    at bootstrap_node.js:608:3
rodrigobranas:javascriptmasterclass $
```



A screenshot of a Mac OS X terminal window titled "variables_14.js — javascriptmasterclass". The window is split into two panes: a code editor on the left and a terminal on the right.

Code Editor (Left):

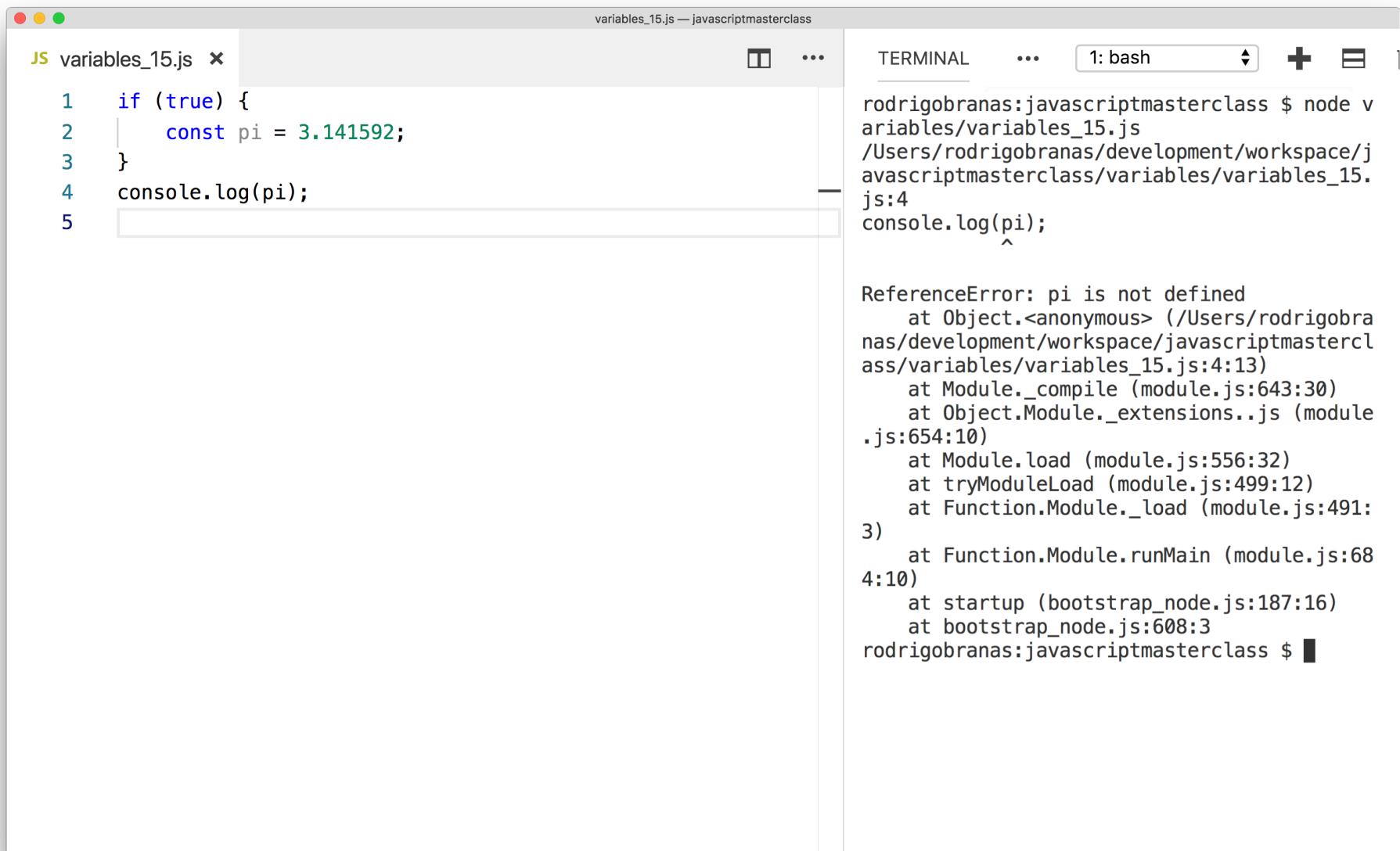
```
JS variables_14.js x
variables_14.js — javascriptmasterclass

1  const pi = 3.141592;
2  console.log(pi);
3  const pi = 3;
4  console.log(pi);
5
```

Terminal (Right):

```
rodrigobranas:javascriptmasterclass $ node variables/variables_14.js
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_14.js:3
const pi = 3;
^

SyntaxError: Identifier 'pi' has already been declared
  at createScript (vm.js:80:10)
  at Object.runInThisContext (vm.js:139:10)
)
  at Module._compile (module.js:607:28)
  at Object.Module._extensions..js (module.js:654:10)
  at Module.load (module.js:556:32)
  at tryModuleLoad (module.js:499:12)
  at Function.Module._load (module.js:491:3)
  at Function.Module.runMain (module.js:684:10)
  at startup (bootstrap_node.js:187:16)
  at bootstrap_node.js:608:3
rodrigobranas:javascriptmasterclass $
```



A screenshot of a Mac OS X terminal window. The window title is "variables_15.js — javascriptmasterclass". The left pane shows the code for "variables_15.js":

```
1  if (true) {  
2      const pi = 3.141592;  
3  }  
4  console.log(pi);  
5
```

The right pane shows the terminal output:

```
rodrigobranas:javascriptmasterclass $ node variables/variables_15.js  
/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_15.js:4  
console.log(pi);  
^  
ReferenceError: pi is not defined  
    at Object.<anonymous> (/Users/rodrigobranas/development/workspace/javascriptmasterclass/variables/variables_15.js:4:13)  
    at Module._compile (module.js:643:30)  
    at Object.Module._extensions..js (module.js:654:10)  
    at Module.load (module.js:556:32)  
    at tryModuleLoad (module.js:499:12)  
    at Function.Module._load (module.js:491:3)  
    at Function.Module.runMain (module.js:684:10)  
    at startup (bootstrap_node.js:187:16)  
    at bootstrap_node.js:608:3  
rodrigobranas:javascriptmasterclass $
```

Ao declarar uma variável sem var, let ou
const ela é criada no escopo global



CAUTION

Nunca declare variáveis sem var, let e
const e evite a utilização de var

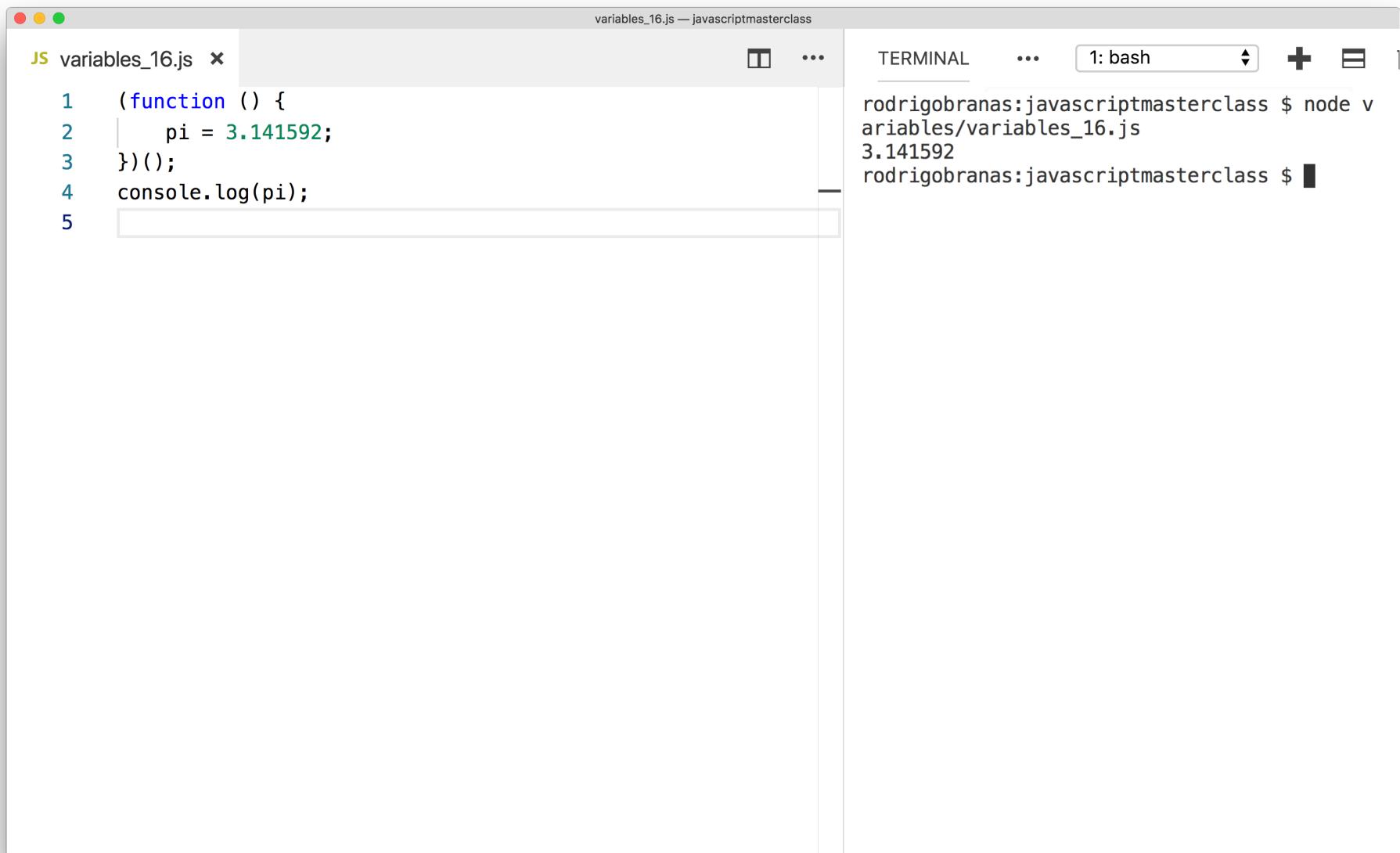
variables_16.js — javascriptmasterclass

JS variables_16.js x

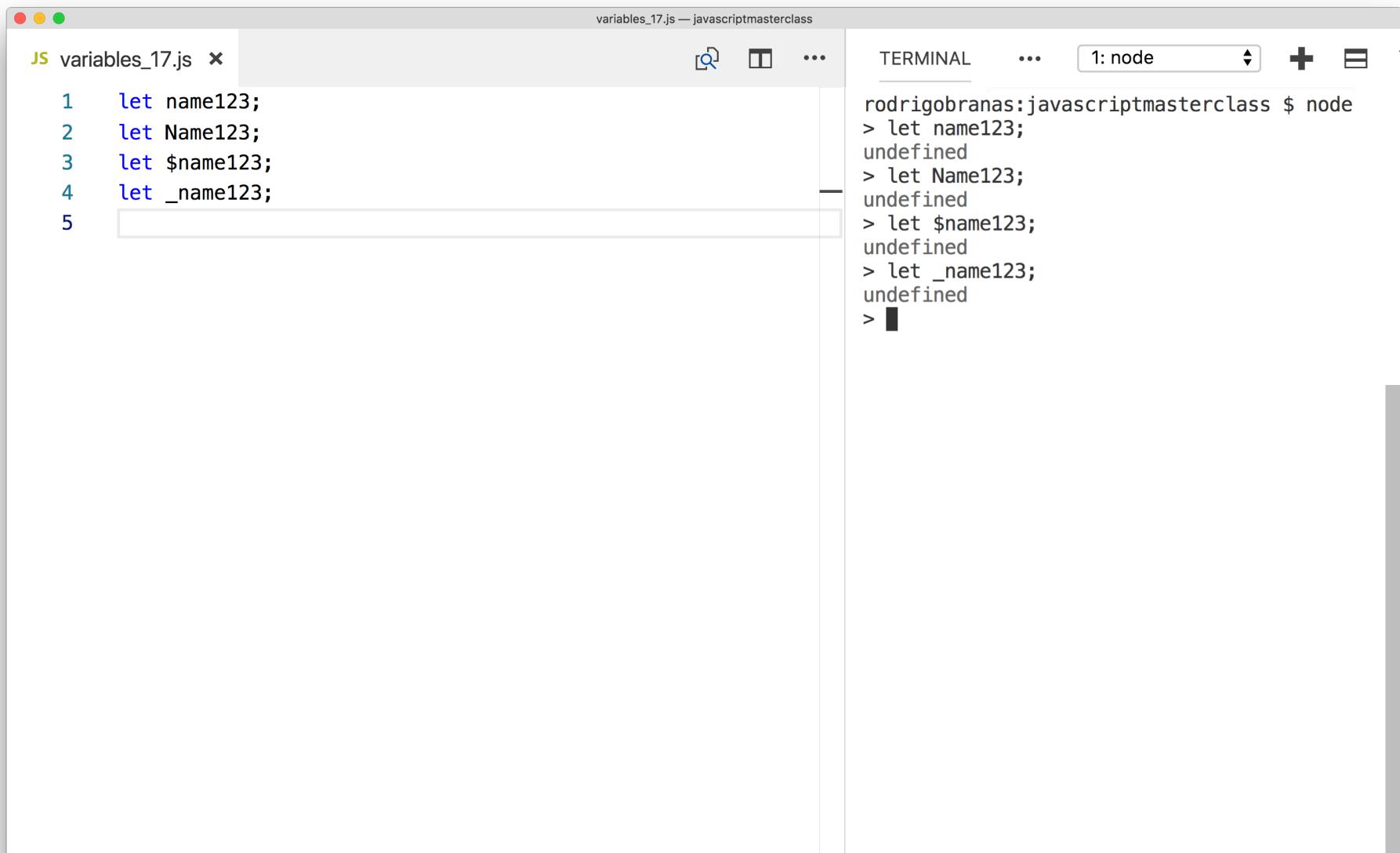
```
1 (function () {  
2   pi = 3.141592;  
3 })();  
4 console.log(pi);  
5
```

TERMINAL ... 1: bash

```
rodrigobranas:javascriptmasterclass $ node variables/variables_16.js  
3.141592  
rodrigobranas:javascriptmasterclass $
```



Um identificador válido deve começar com
[a-zA-Z_] seguido por [a-zA-Z0-9_]



A screenshot of a Mac OS X terminal window. The window title is "variables_17.js — javascriptmasterclass". The main pane contains the following code:

```
1 let name123;
2 let Name123;
3 let $name123;
4 let _name123;
5
```

The right pane, labeled "TERMINAL", shows the output of running the script with "node". The output is:

```
rodrigobranas:javascriptmasterclass $ node
> let name123;
undefined
> let Name123;
undefined
> let $name123;
undefined
> let _name123;
undefined
> █
```

variables_18.js — javascriptmasterclass

JS variables_18.js x

1 let 123name;
2 let @name;
3 let "name;
4 let 'name;
5

TERMINAL ... 1: node

rodrigobranas:javascriptmasterclass \$ node
> let 123name;
let 123name;
 ^

SyntaxError: Invalid or unexpected token

> let @name;
let @name;
 ^

SyntaxError: Invalid or unexpected token

> let "name;
let "name;
 ^^^^^

SyntaxError: Invalid or unexpected token

> let 'name;
let 'name;
 ^^^^^

SyntaxError: Invalid or unexpected token

> █