



KINGSLAND
UNIVERSITY

Introduction to Programming



J
S



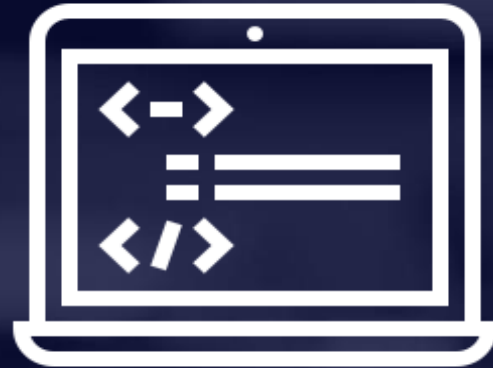
Creating **JavaScript** Programs in Visual Studio Code



Table of Contents

- Welcome to Programming
 - What is Coding?
 - Writing Simple Commands and Programs
- The JavaScript Language
- IDEs and Visual Studio Code
- Creating and Running Console Apps
- Coding Exercises





Welcome to Coding

Writing and Running JavaScript Programs



What is Coding?

- **Coding** means to give **commands** to tell the computer what to do
- Sample command:

```
console.log("I am coding");
```

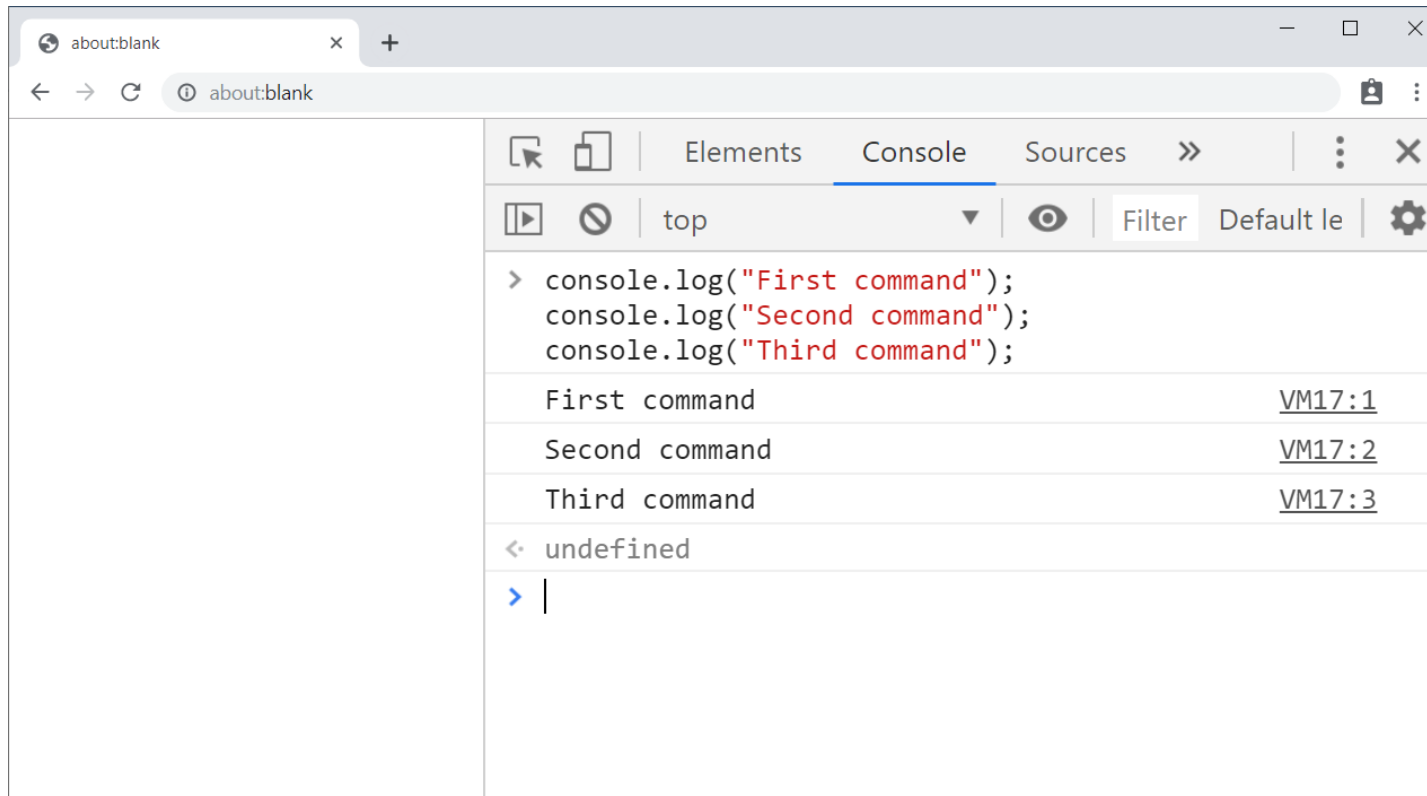
- A **computer program** is a sequence of commands

```
console.log("First command");  
console.log("Second command");  
console.log("Third command");
```



Running JavaScript Commands

- The easiest way to **run JavaScript commands** (JS commands)
 - The **browser console**: press **[F12]** in your Web browser





Commands in JavaScript – Examples

- Calculate an expression and print its value:

```
console.log(5 + 5);
```

- Calculate an area of rectangle:

```
console.log(base * height / 2);
```

- Print the numbers from 1 to 100

```
for (let i = 1; i <= 100; i++)  
    console.log(i);
```





Programming and Algorithms

- **Programming** means writing computer programs (commands)
 - Using certain **programming language**, such as JavaScript or C#
- **Algorithm** == a sequence of commands that achieves certain result
- Programming is done by **programmers** (developers)
- Programmers use IDE or code editors (like Visual Studio Code) to:
 - **Write** the code
 - **Run** and test the code
 - Find a fix **bugs** (debug the code)





Computer Program – Example

- Sample **JavaScript** (sequence of JavaScript commands):

```
let size = 5;  
console.log("Size = " + size);  
console.log("Area = " + size * size);
```

PROBLEMS

DEBUG CONSOLE

...



C:\Program Files\nodejs\node.exe intro.js

Size = 5

Area = 25



Complete Computer Program

- Sample complete **JavaScript** program (JS function + invocation):

```
function calculateArea() {  
    let size = 5;  
    console.log("Size = " + size);  
    console.log("Area = " + size * size);  
}  
  
calculateArea();
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

C:\Program Files (x86)\nodejs\node.exe example.js

Size = 5

Area = 25

Console-Based JS Program – Example

- JavaScript program, which converts from **USD** to **EUR**

```
function convertUsdToEur(input) {  
    let dollars = parseFloat(input);  
    let euro = dollars * 0.883795087;  
    console.log("Euro: " + euro);  
}  
convertUsdToEur("5");
```

The console input typically comes as text

DEBUG CONSOLE



Launch Program



C:\Program Files\nodejs\node.exe intro.js

Euro: 4.418975435



Browser-Based JS Program – Example

- Convert from **USD** to **EUR** in a Web page (HTML code)

```
<html><body>  
  <script src="converter.js"></script>  
  Dollars: <input type="text" id="dollarsBox" />  
  <button onclick="convertUsdToEur()">Convert</button>  
  Euro: <input type="text" id="eurosBox" readonly />  
</body></html>
```

converter.html

← → ↻ ⓘ File | C:/Projects/JS-Basics/converter.html

Dollars:

Euro:

Browser-Based JS Program – Example

- Convert from **USD** to **EUR** in a Web page (JS code)

```
function convertUsdToEur() {  
    let dollars = parseFloat(  
        document.getElementById("dollarsBox").value);  
    let euro = dollars * 0.883795087;  
    document.getElementById("eurosBox").value = euro;  
}
```

converter.js

← → ↻ ⓘ File | C:/Projects/JS-Basics/converter.html

Dollars:

Euro:



The JavaScript Language

Modern, Dynamic Language



Programming Languages, Compilers, Interpreters

- The **programming language** defines a set of rules (syntax)
- Languages are either **compiled** (like C#, C++) or **interpreted** (like JavaScript and Python)
- The **compiler** translates programming code to **machine code**
 - During the compilation, it finds syntax errors
- The **interpreter** executes the code line by line
 - It finds syntax errors at runtime



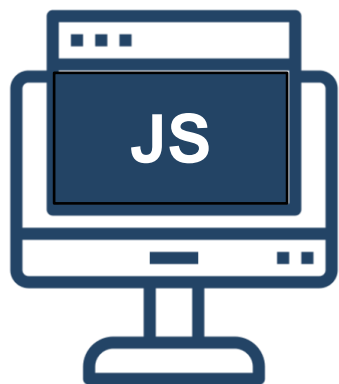
The JavaScript Language

- **JavaScript** is prototype-based, multi-paradigm, dynamic language
 - Supports imperative, functional and object-oriented programming
- Used both on the **client-side** (in the Web browser) and on the **server-side** (in Node.js)
- Good choice for **beginners**
 - Easy syntax, easy to read, **easy to learn**
 - Big developer **community**, many jobs
 - Lots of **resources**: books, lessons, courses

JavaScript is not Java!

JavaScript Execution

- JavaScript is a **high-level, dynamic, interpreted language**
- A program in JS is written in text format (**source code**)
- The **interpreter** executes the code line by line



interpreter



execution engine



IDE - Integrated Development Environments



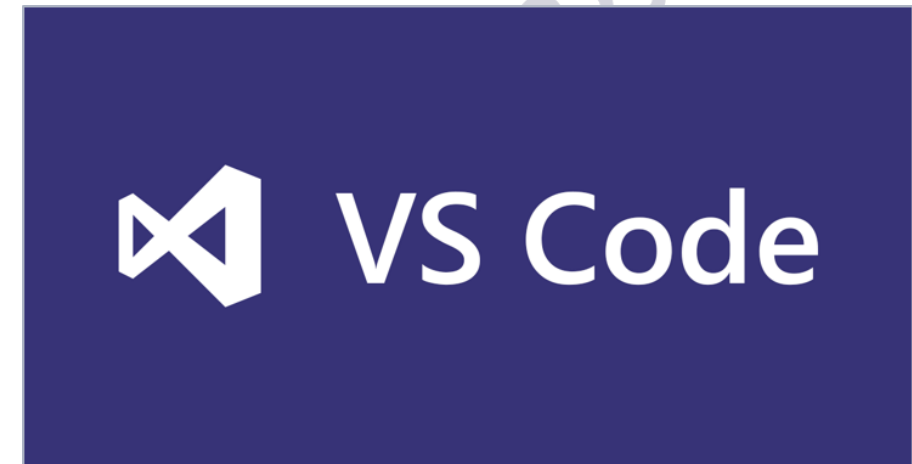
Integrated Development Environments (IDE)

- To write programs, we need an **IDE** (development environment)
 - For **JavaScript**
 - **Visual Studio Code** (free IDE)
 - **WebStorm** (paid IDE)
 - **Atom / Vim / NetBeans / Eclipse / Visual Studio**
 - For **Java**
 - **IntelliJ IDEA / Eclipse / NetBeans**
 - For **Python**
 - **PyCharm / Eclipse / VS Code**
 - For **C#**
 - **Visual Studio / Raider / VS Code**



Integrated Development Environments (IDE)

- Integrated Development Environments (**IDE**) simplify the software development
- **IDEs** save time and help developers
 - Writing, running, testing, and debugging the code + deployment
- For programming in JavaScript, we may use the **Visual Studio Code** IDE
 - Free code editor from Microsoft





Development Environments: Desktop and Online

- Install **Visual Studio Code**: <https://code.visualstudio.com>
- Alternative IDE (online)
 - **JavaScript**: directly use the browser console (press [F12] key)
 - Or use <https://playcode.io>,
<https://repl.it/languages/nodejs>, <https://codesandbox.io>,
<https://jsbin.com>
 - **C#**: <https://dotnetfiddle.net>
 - **Java**: <https://compilejava.net>
 - **Python**: <https://repl.it>



Online Coding Environments (IDEs)

- Open the Web browser console (press **[F12]** on most browsers)



The screenshot shows a web browser's developer console with the 'Console' tab selected. The console displays the following JavaScript code and its execution results:

```
> function calculateArea() {  
    let size = 5;  
    console.log("Size = " + size);  
    console.log("Area = " + size * size);  
}  
  
calculateArea();
```

The output of the function call is shown in two lines:

```
Size = 5  
Area = 25
```

The console also shows the return value of the function call, which is `undefined`.

Size = 5	VM30:3
Area = 25	VM30:4
undefined	



Visual Studio Code

Powerful Code Editor and IDE



What is Node.js?

- **Node.js** == server-side **JavaScript** runtime + toolset for developers
 - Executes JavaScript (JS) code outside of the Web browser

```
Command Prompt - node
C:\Users\nakov>node
Welcome to Node.js v12.1.0.
Type ".help" for more information.
> let size = 5;
undefined
> console.log("Size = " + size);
Size = 5
undefined
> console.log("Area = " + size * size);
Area = 25
undefined
> _
```








Installing Node.js

LTS
Recommended For Most Users

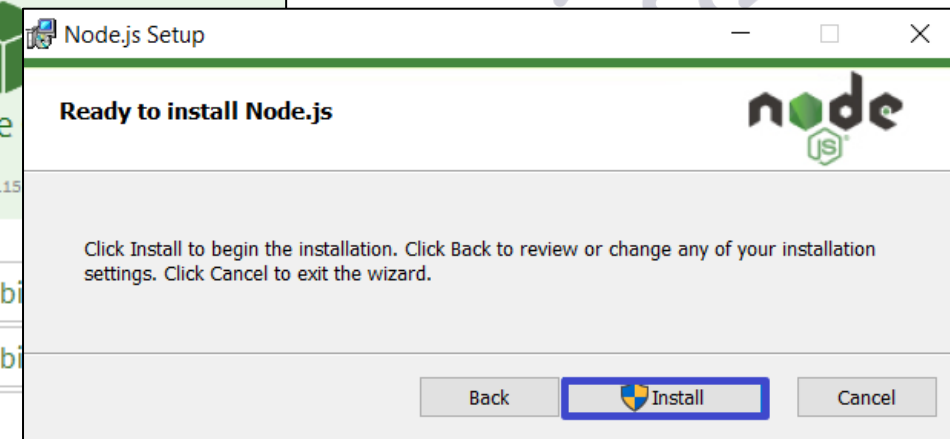
Current
Latest Features


Windows Installer
node-v10.15.3-x84.msi


macOS Installer
node-v10.15.3.pkg


Source
node-v10.15.3.tar.gz

Windows Installer (.msi)	32-bit	64-bit	
Windows Binary (.zip)	32-bit	64-bit	
macOS Installer (.pkg)	64-bit		
macOS Binary (.tar.gz)	64-bit		
Linux Binaries (x64)	64-bit		
Linux Binaries (ARM)	ARMv6	ARMv7	ARMv8
Source Code	node-v10.15.3.tar.gz		

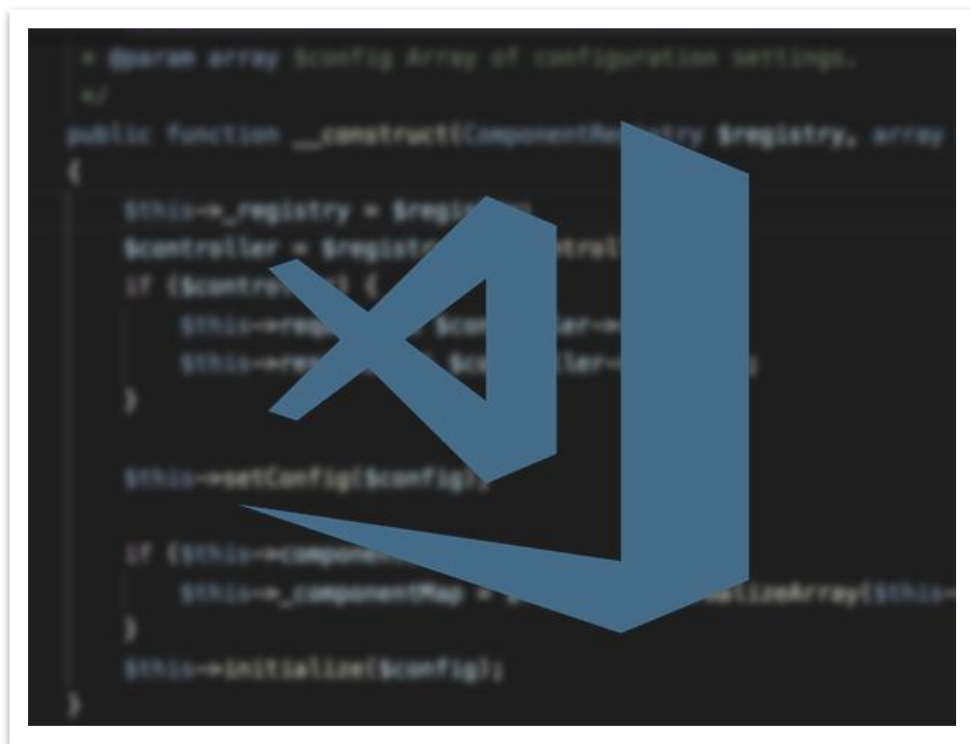


Download from: <https://nodejs.org/en/download/>



What is Visual Studio Code?

- **Visual Studio Code (VS Code)** is powerful code editor and IDE
 - For Windows, Linux and macOS, free and open-source



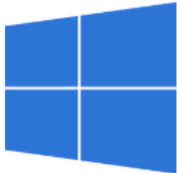
- **Many languages & frameworks:** JavaScript, Node.js, C#, Python, Java, PHP, Go, C++, Ruby, etc.



Installing Visual Studio Code


Download Visual Studio Code

Free and open source. Integrated Git, debugging and extensions.



↓ Windows
Windows 7, 8, 10


User Installer	64 bit	32 bit
System Installer	64 bit	32 bit
.zip	64 bit	32 bit



↓ .deb
Debian, Ubuntu

↓ .rpm
Red Hat, Fedora, SUSE

.deb	64 bit	32 bit
.rpm	64 bit	32 bit
.tar.gz	64 bit	32 bit



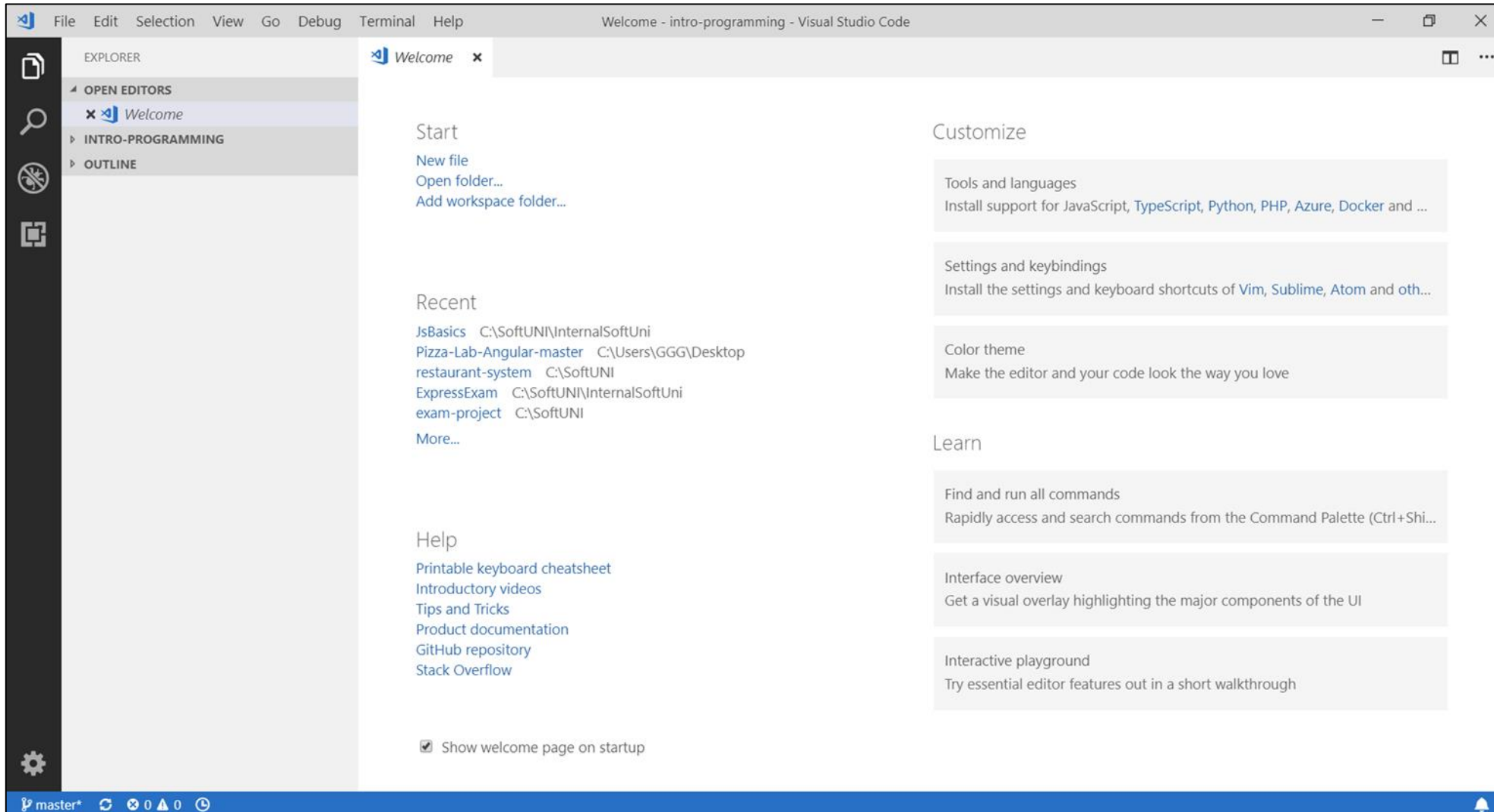
↓ Mac
macOS 10.9+

- Easy to install
- Choose your OS and follow the installer

Download from: <https://code.visualstudio.com/download>

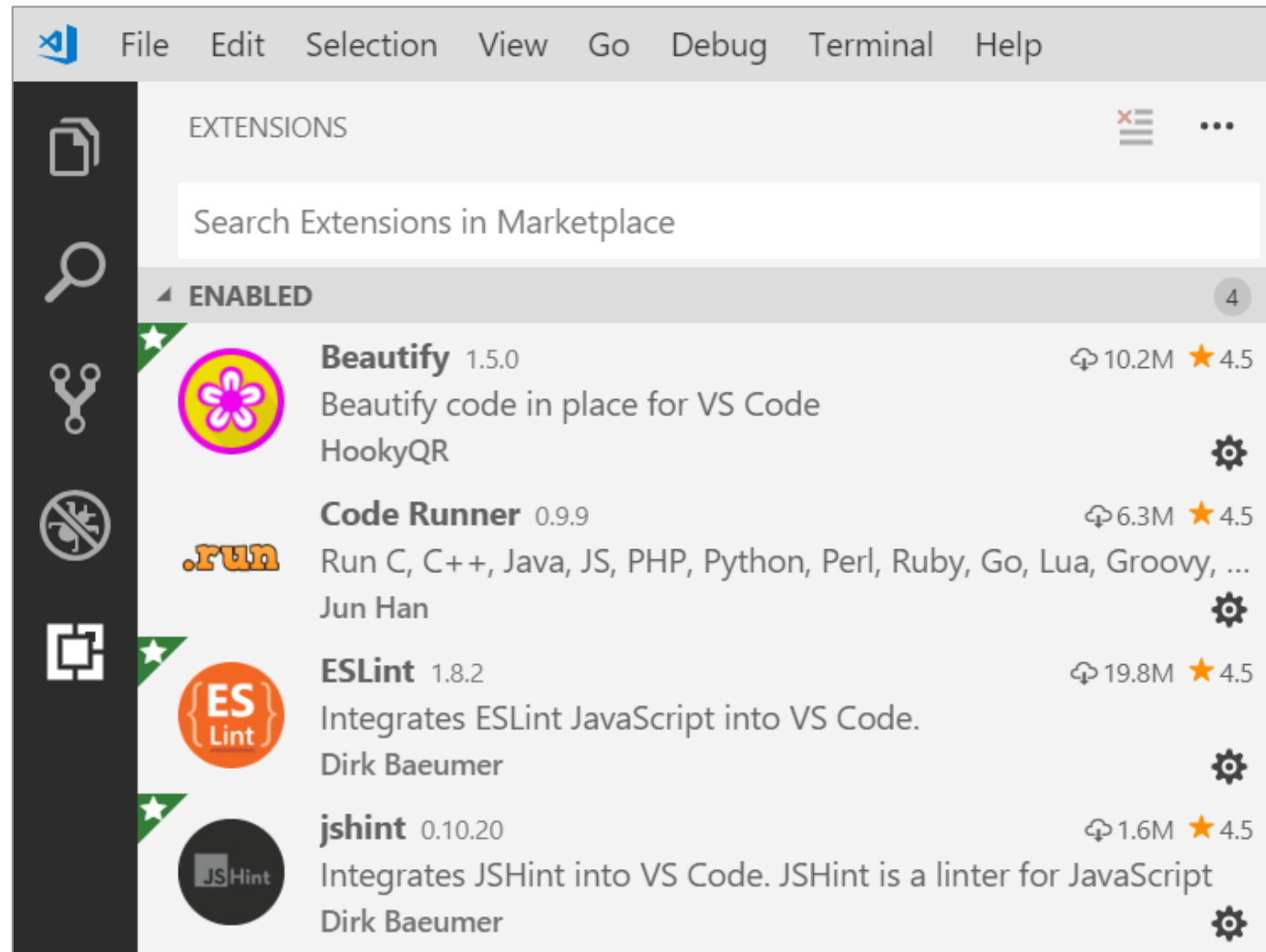


Running Visual Studio Code





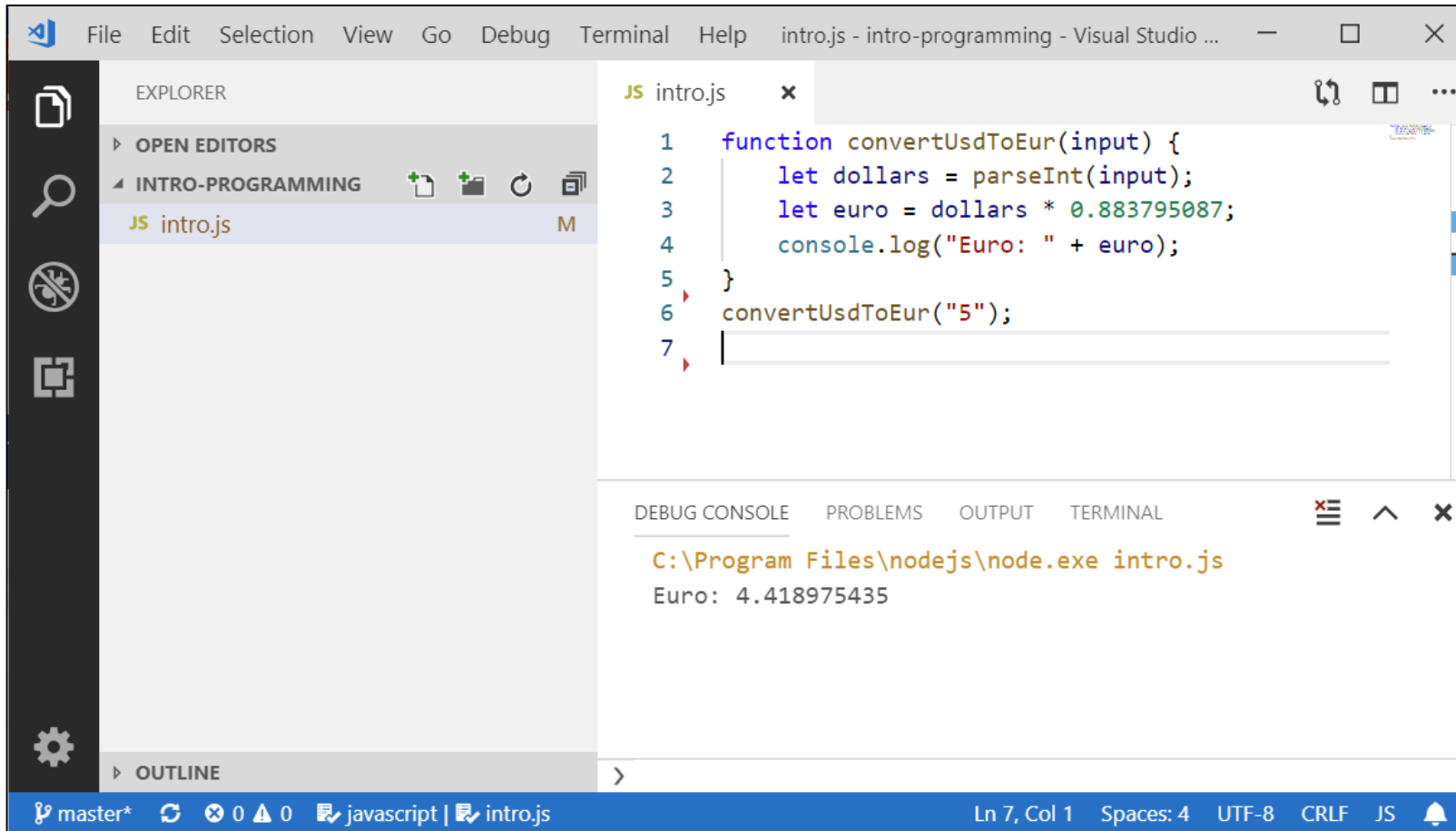
Recommended JS Extensions for VS Code



- Install the following **extensions** in VS Code:
 - **Code Runner, Beautify, JSHint, ESLint**



Projects in VS Code



- VS Code **explorer** holds **projects**
- **Projects** hold source code files: **.js files**



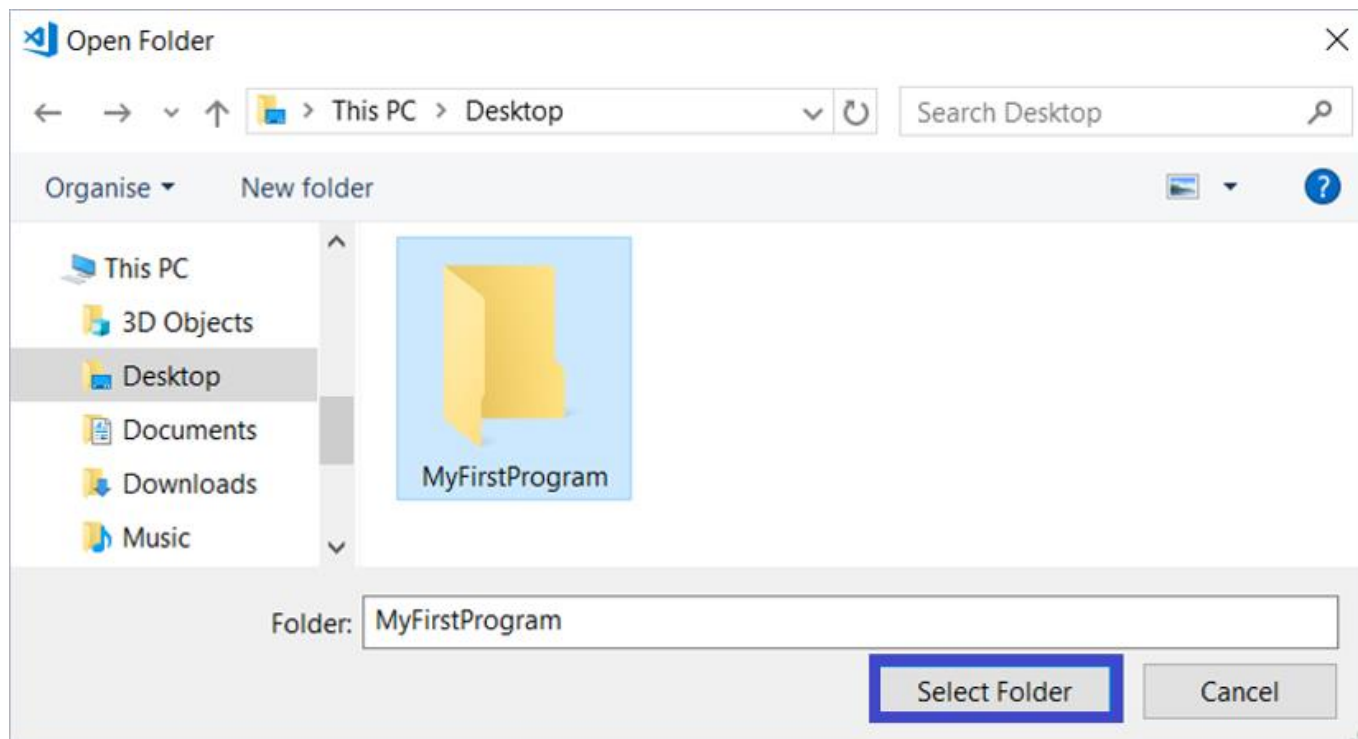
Console App in VS Code

Create and Run a Simple Console App in Visual Studio Code



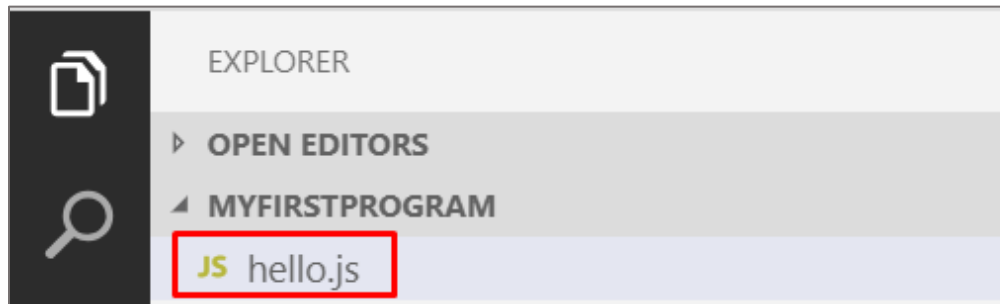
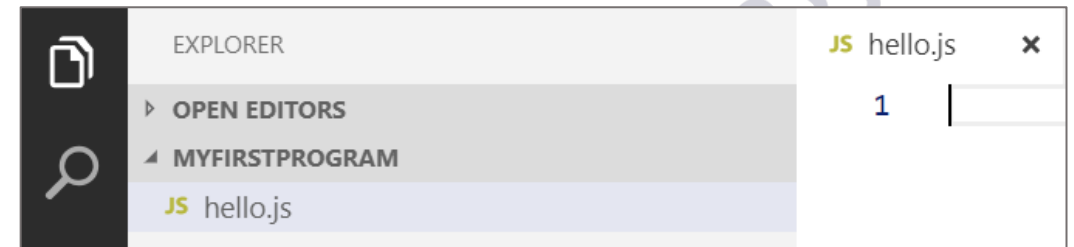
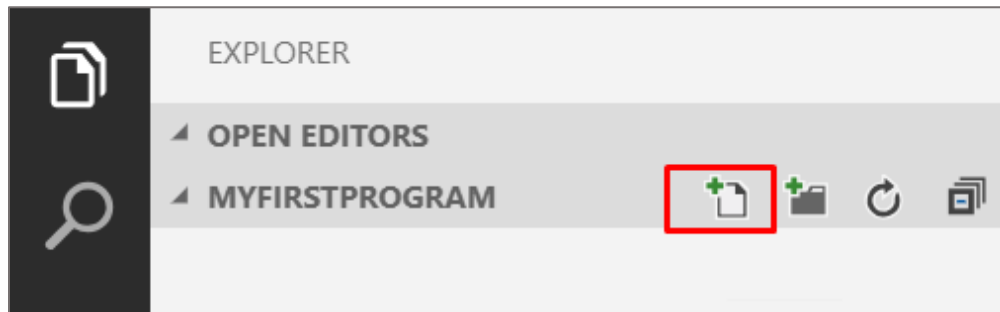
Creating a Console Application in VS Code

- Create a **folder** for your project and open it from VS Code:
 - [File] > [Open Folder]



Writing Programming Code in VS Code

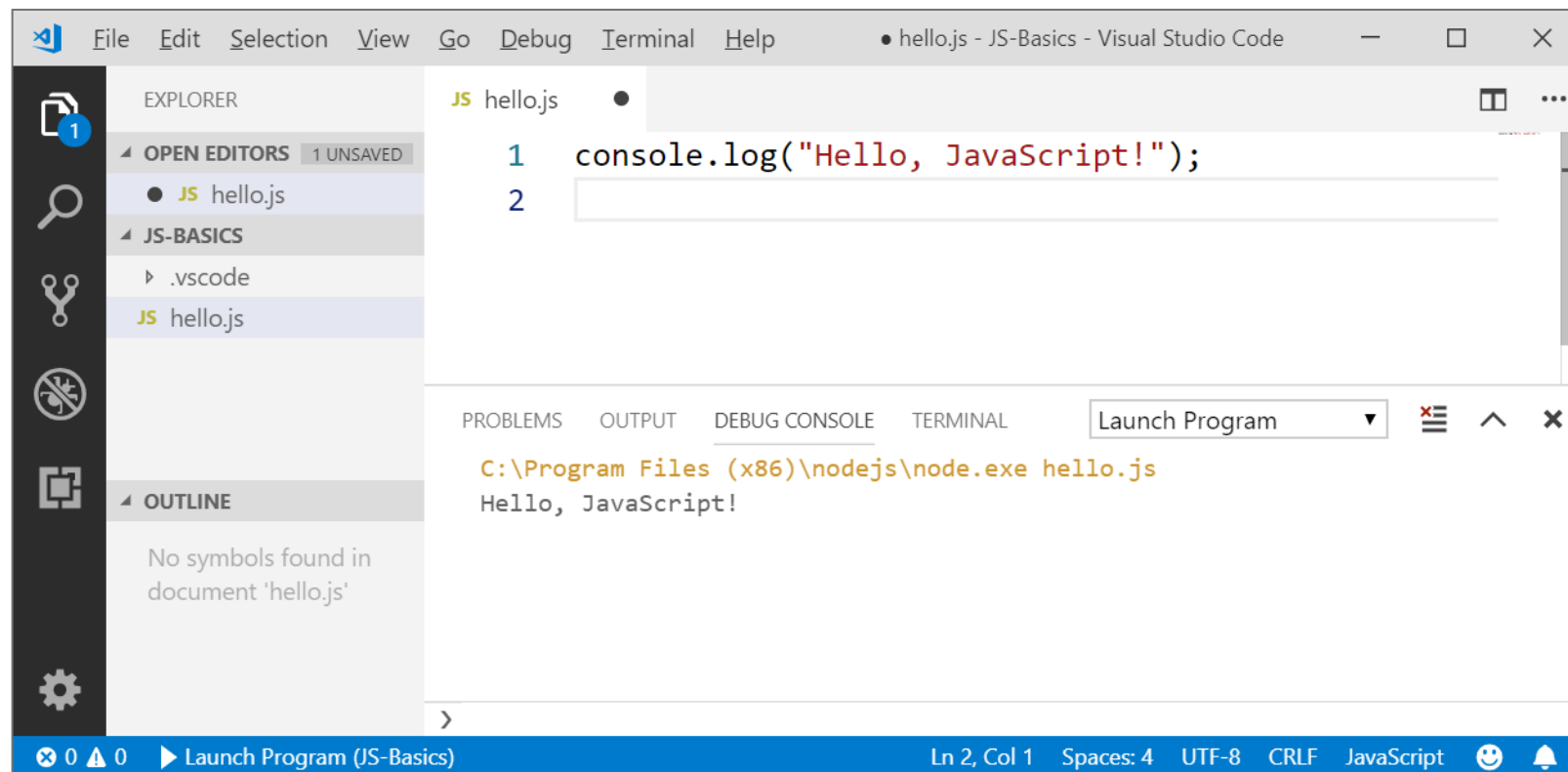
- Create a file **hello.js** to hold your program's source code





Starting the Program in VS Code

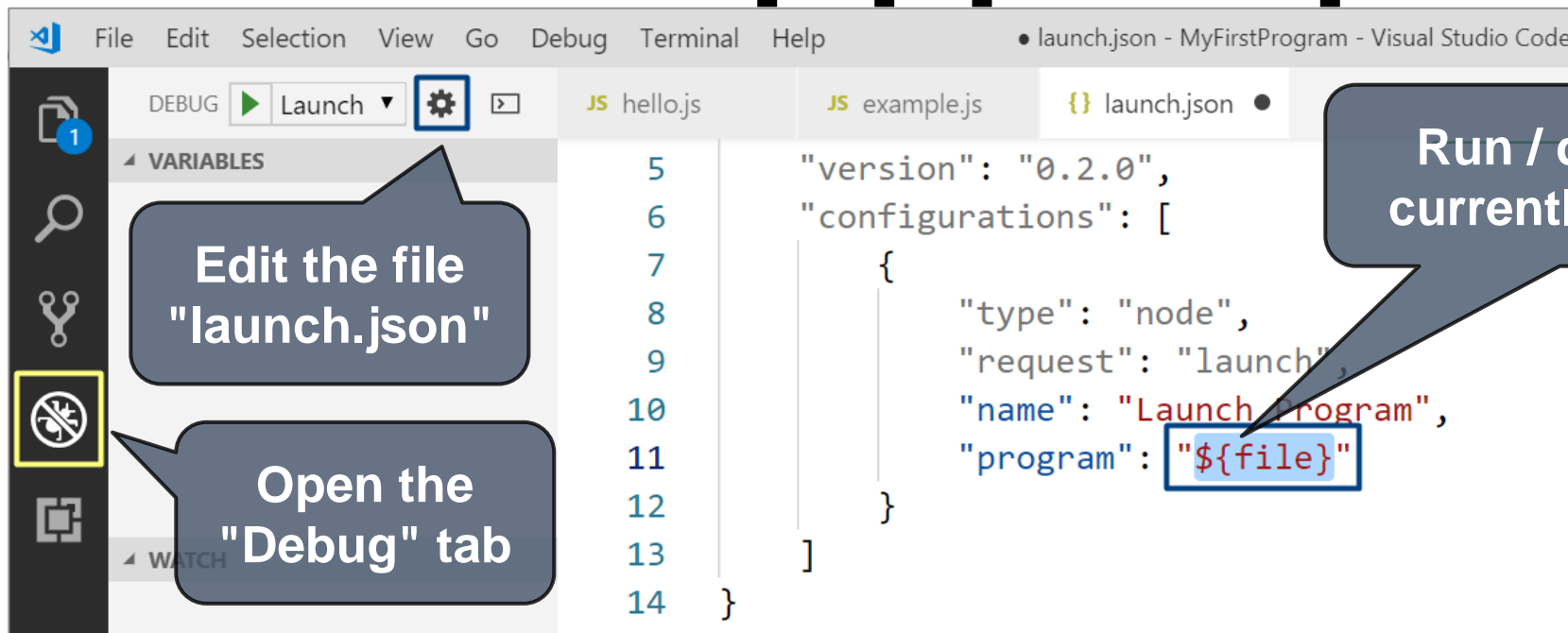
- To start the program, press **[Ctrl + F5]**
- The result will appear in the **[Debug Console]** tab





Start the Current JS File in VS Code

- In VS Code **[F5]** / **[Ctrl+F5]** keys runs your earliest created **.js** file
- If you have multiple **.js** files in VS Code, you may want to **start the current file** with **[F5]** / **[Ctrl + F5]** edit the **launch.json**



Edit the file
"launch.json"

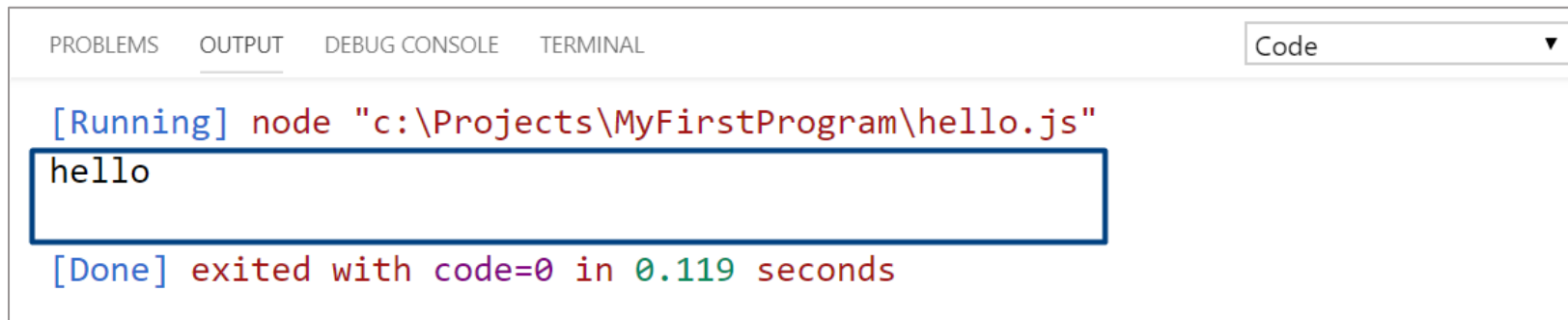
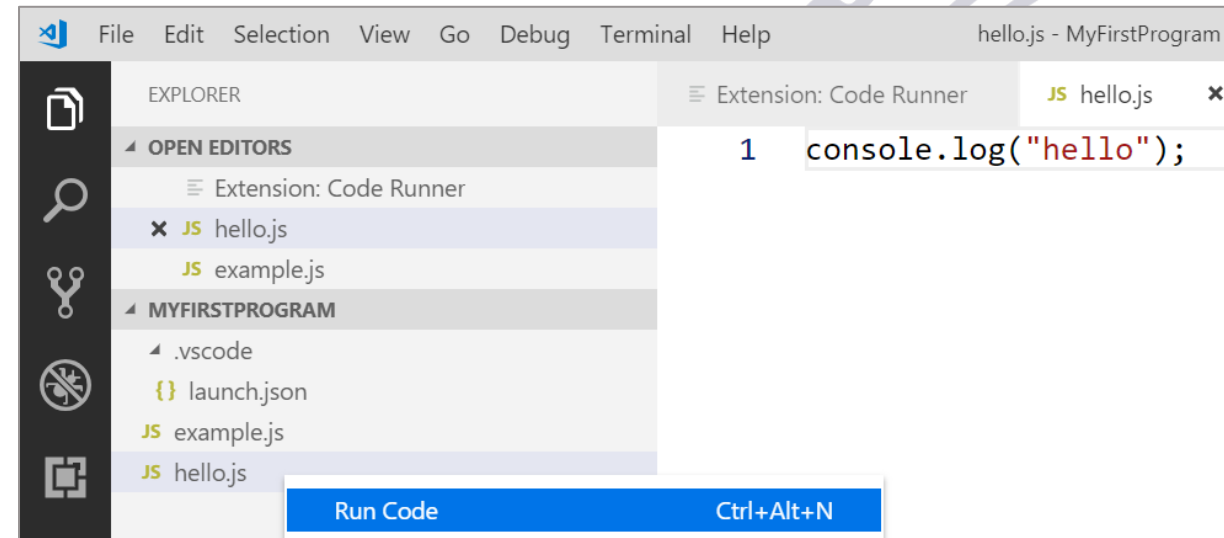
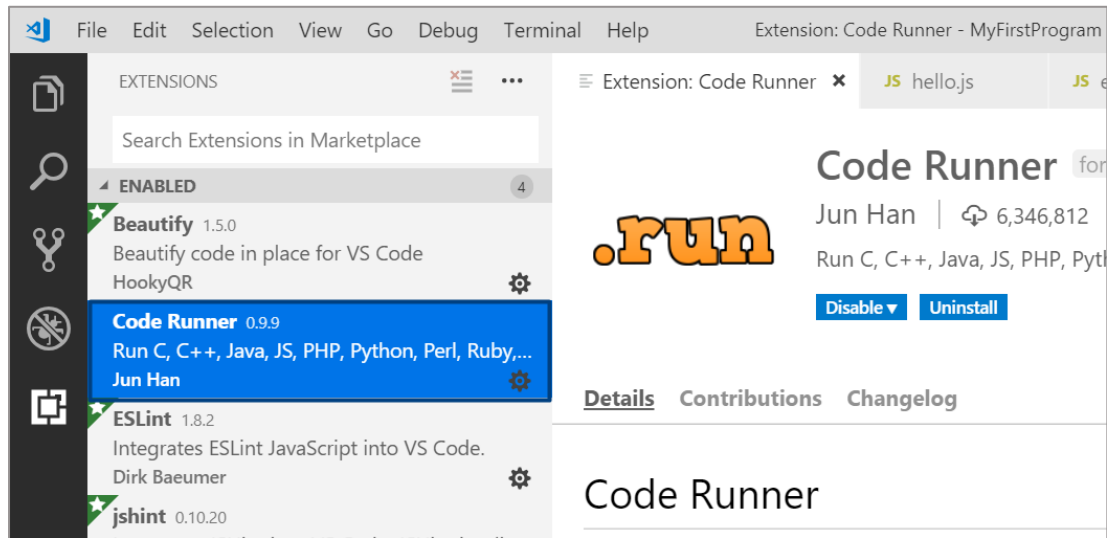
Open the
"Debug" tab

Run / debug the
currently open file



Start the Current JS File in VS Code

- Alternatively, use the **"Code Runner"** extension





Typical Program Errors

Commonly Encountered Errors in the JavaScript Programs



Typical Errors in a JS Program

- Visual Studio Code will complain about **syntax errors**
 - Underlines the wrong code

```
JS intro.js x
1 | console.log('Hello, JavaScript);
   |                               ^
   | Unterminated string literal. ts(1002)
   | Quick Fix... Peek Problem
```



More Typical Errors in a JS Program

- Mixing of **lowercase** and **uppercase** letters:

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL
Debugging with inspector protocol because Node.js v8.9.4 was detected.
node .vscode\Demo.js
c:\Users\User\Documents\VS Code Projects\Loops\.vscode\Demo.js:2
  Console.log("Hello")
  ^
ReferenceError: Console is not defined
    at hello (c:\Users\User\Documents\VS Code Projects\Loops\.vscode\Demo.js:2:5)
    at Object.<anonymous> (c:\Users\User\Documents\VS Code Projects\Loops\.vscode\Demo.js:5:1)
```

- Missing quote " or bracket (or)

```
console.log("Hello JavaScript);
```



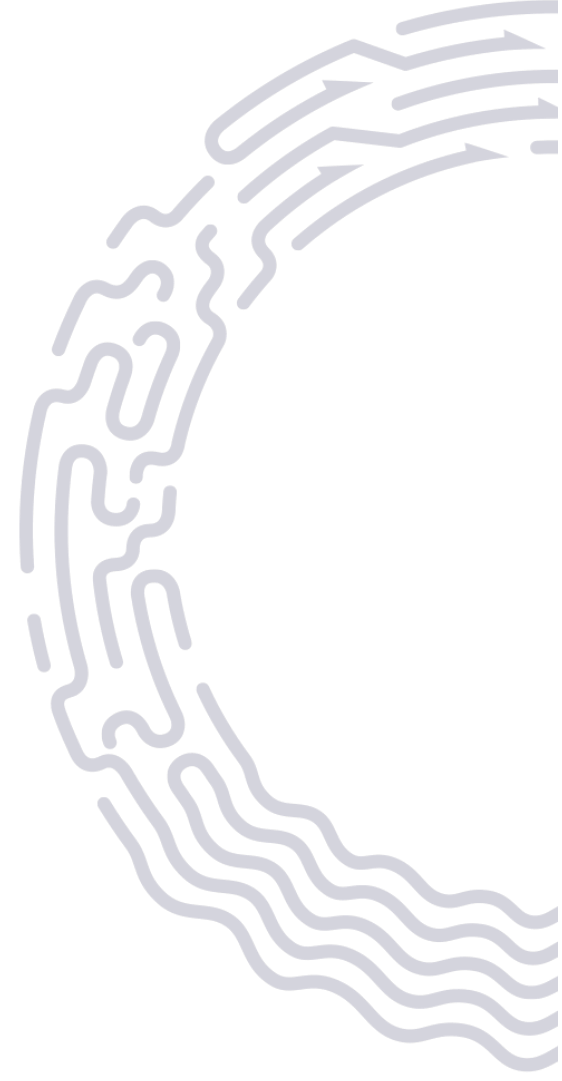
Summary

- What is Coding?
 - Commands and Programs
- The JavaScript Language
- The Visual Studio Code
 - Installing Node.js and VS Code
 - Writing and Running JavaScript Console Apps





QUESTIONS?





License

- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © Kingsland University – <https://kingslanduniversity.com>



THANK YOU