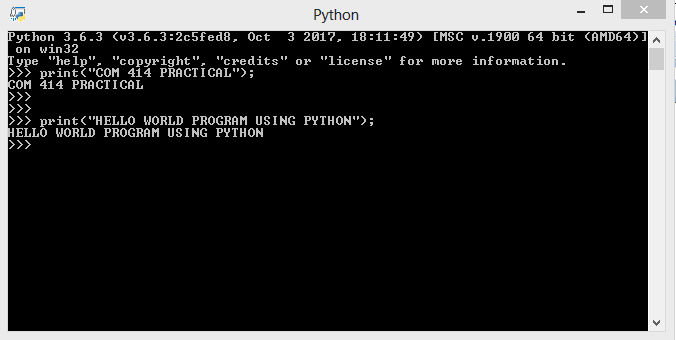
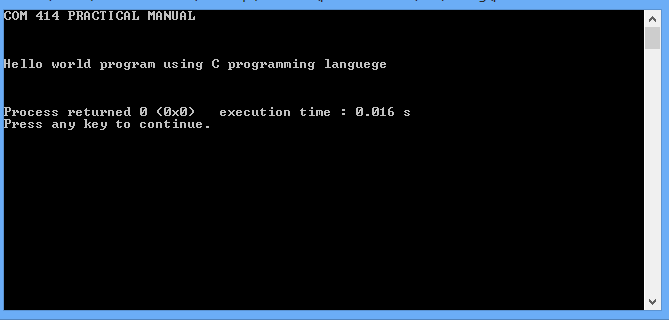
PRACTICAL ONE

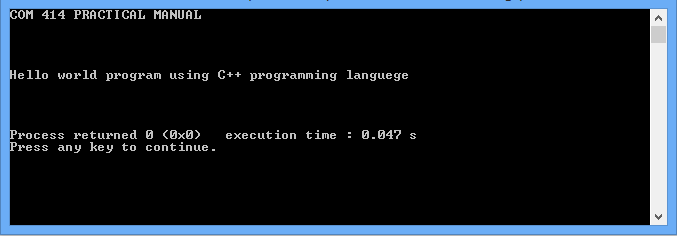
Using Python



Using C programming language

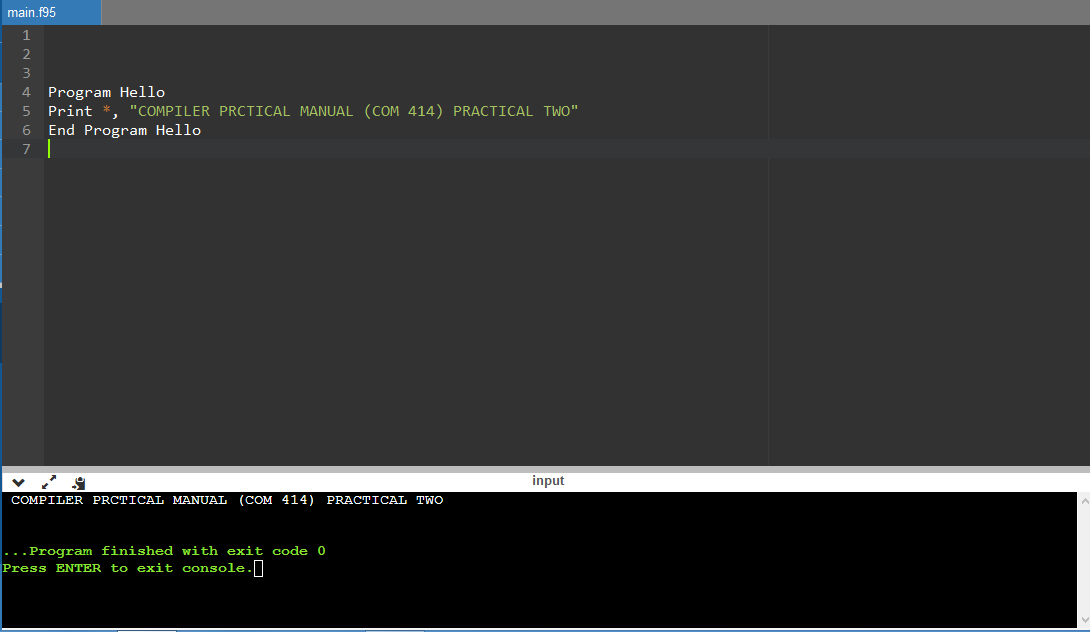


Using C++ programming language

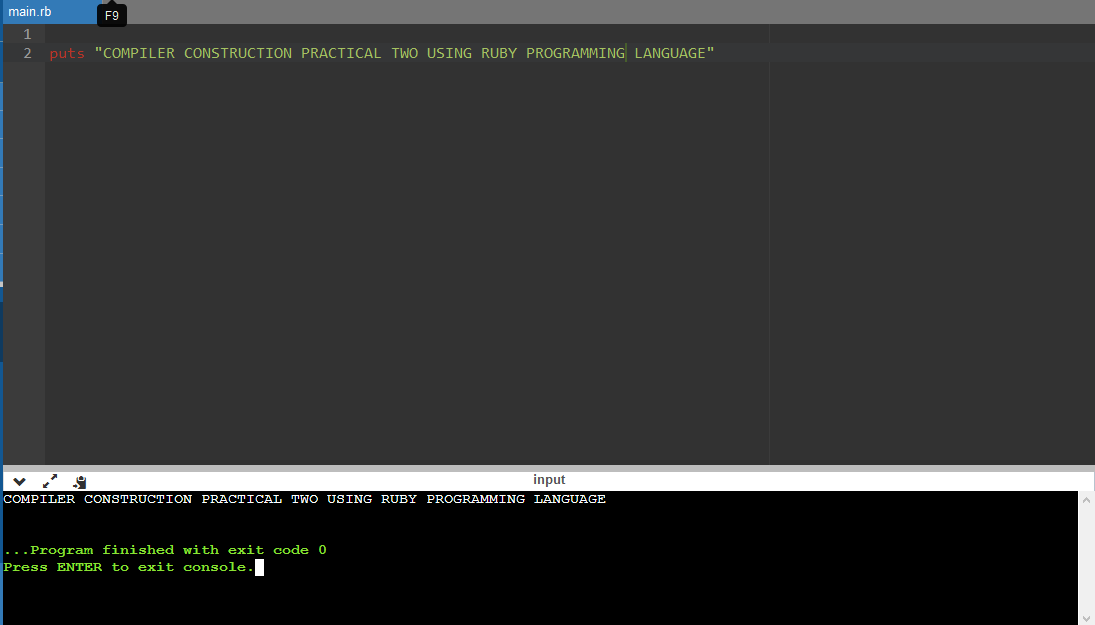


PRACTICAL TWO

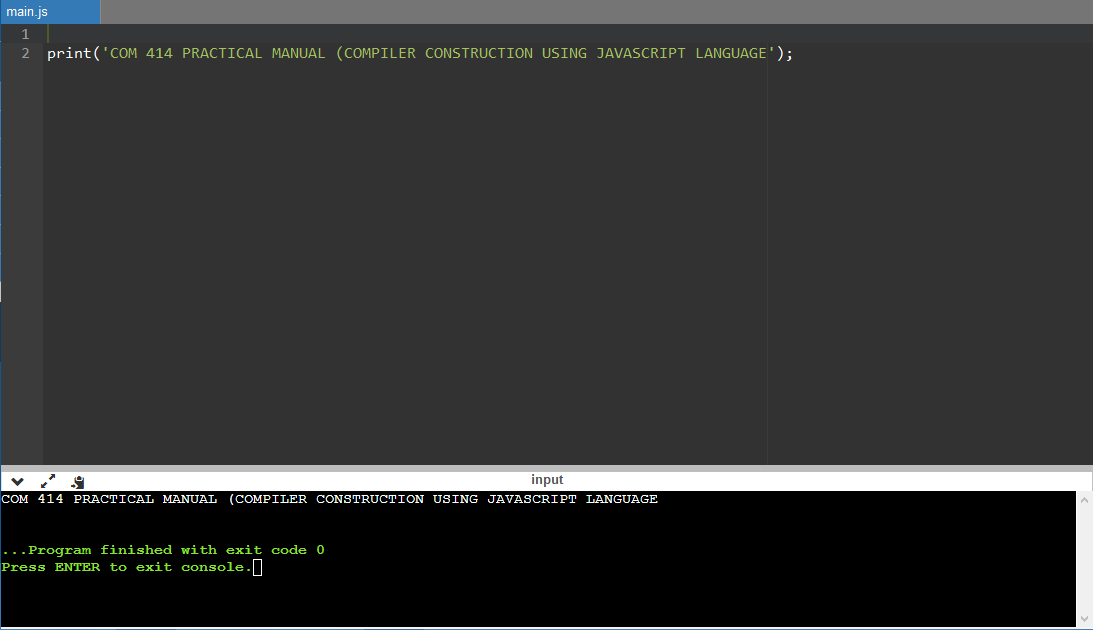
Using Fortran



Using Ruby

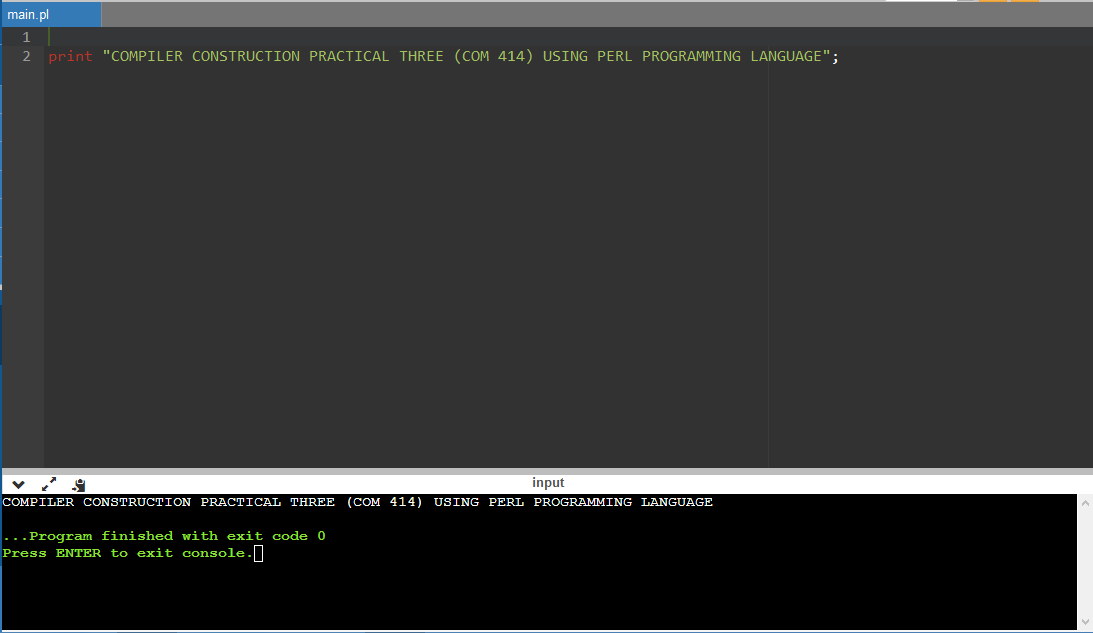


Using JavaScript

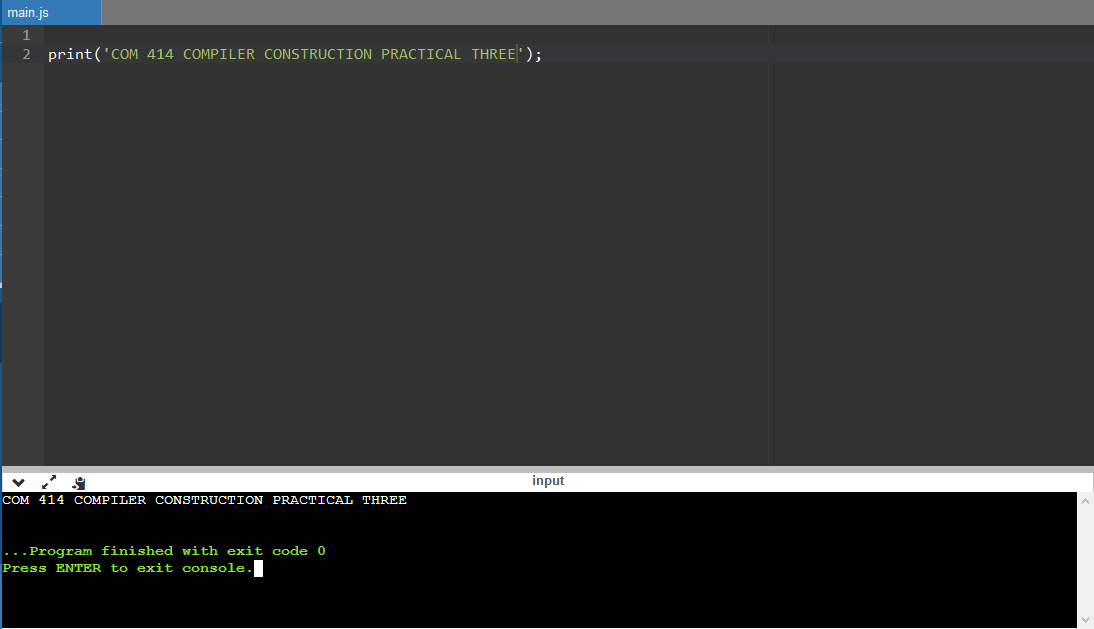


PRACTICAL THREE

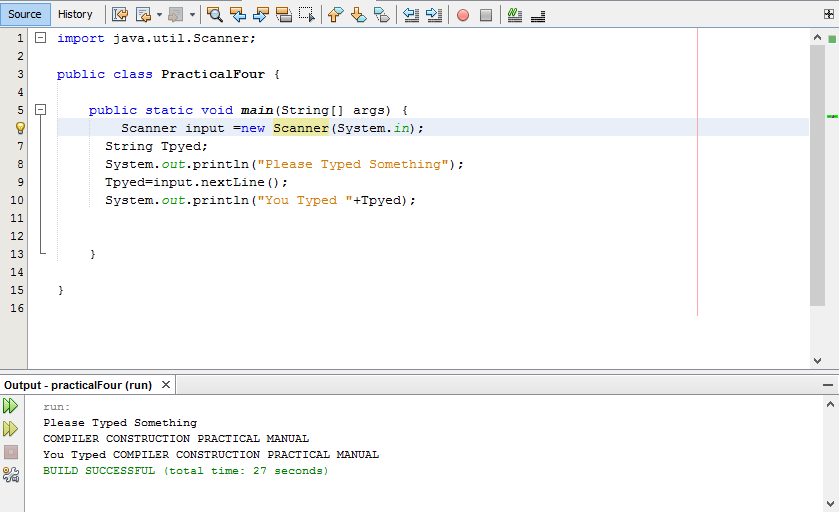
Using Perl



Using JavaScript 3

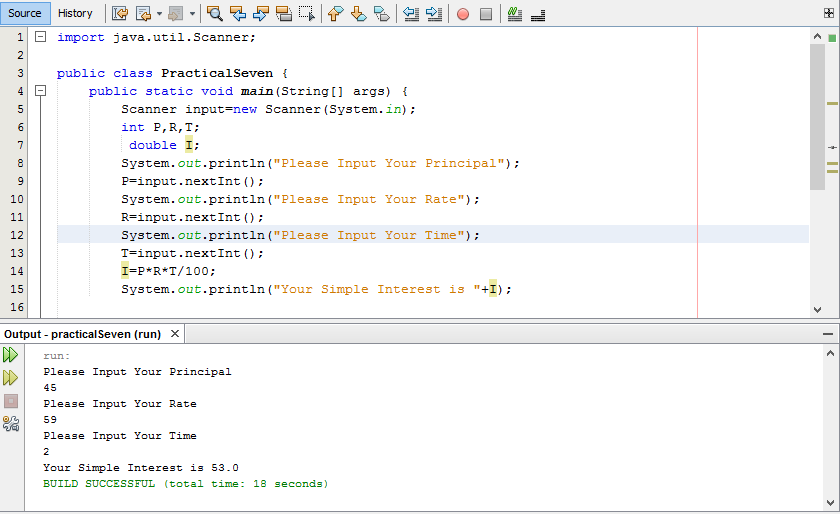


PRACTICAL FOUR



PRACTICAL FIVE

PRACTICAL SEVEN



COMPILER WRITEUP

**PRACTICAL ONE**

uppose, you save a C program with prg1.c – here .c is the extension of C code, prg1.c file contains the program (source code of a C program). Preprocessor reads the file and generates the prg1.i (prg1.ii – for c++ source code) file, this file contains the preprocessed code.  
Compiler reads the prg1.i file and further converts into assembly code and generates prg1.s and then finally generates object code in prg1.o file.  
**PRACTICAL TWO**

An assembler or compiler is just another program that executes on your computer system.

The only thing special about an assembler or compiler is that it translates programs from one form (source code) to another (machine code).

A typical x86 assembler, for example, would read lines of text with x86 instructions, parse each statement, and then write the binary equivalent of each instruction directly to memory or to a file for later execution.

Assemblers have two big advantages over coding in machine code:

**PRACTICAL THREE**

JavaScript is a text-based language that does not need any conversion before being executed. Other languages like [Java and C++ need to be compiled to be executable](http://dev.opera.com/articles/view/38-programming-the-real-basics/#interpreted) but JavaScript is executed instantly by a type of program that interprets the code called a parser (pretty much all web browsers contain a JavaScript parser).

To execute JavaScript in a browser you have two options — either put it inside a **script** element anywhere inside an HTML document, or put it inside an external JavaScript file (with a .js extension) and then reference that file inside the HTML document using an empty **script** element with a **src** attribute. We will look at both of these methods inside this section.

**PRACTICAL SIX**

1. On your computer, open Chrome.
2. At the top right, click More. Settings.
3. At the bottom, click Advanced.
4. Under 'Privacy and security', click Content settings.
5. Click JavaScript.
6. Turn on Allowed (recommended).