

# ユーザー増加中のIntelliJ IDEAを中心にIDEを 探る

- 講師：前田和昭
- 担当：中部大学オープンソース研究会
- [kmaeda@gmail.com](mailto:kmaeda@gmail.com)
- <https://github.com/kmaeda/OscNagoya2018>
- オープンソースカンファレンス 2018 Nagoya (2018年5月19日)

多くのスライドでは、Wikipediaのコンテンツを使っています。

この文書は、クリエイティブ・コモンズ 表示-継承ライセンスの下で  
利用可能です。



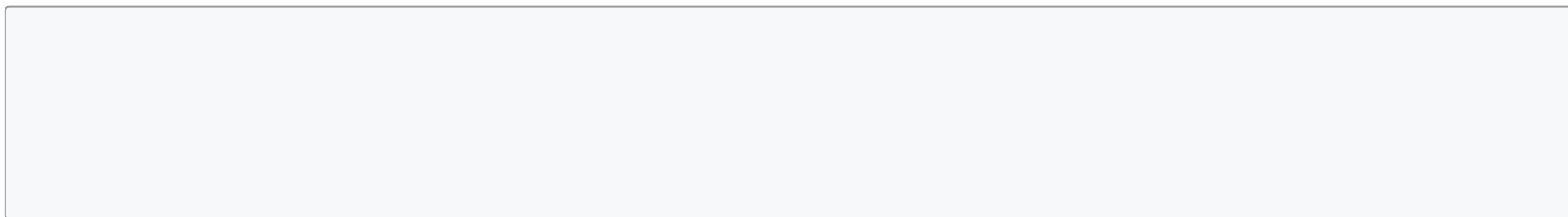
# IntelliJ IDEAを中心にIDEを探る

IntelliJ IDEA Community Editionがオープンソースとして公開されてから約8年たちました.

以前はIDE（統合開発環境）と言えばEclipseが主流だったはずなのに、いつの間にかIntelliJ IDEAの利用者が大幅に増えてきました.

このセミナーでは、IDEの歴史から始まり、無料で公開されているIntelliJ IDEA Community Editionを中心にIDEを解説します.

のはずが...



# IntelliJ IDEAを中心にIDEを探る

IntelliJ IDEA Community Editionがオープンソースとして公開されてから約8年たちました。

以前はIDE（統合開発環境）と言えばEclipseが主流だったはずなのに、いつの間にかIntelliJ IDEAの利用者が大幅に増えてきました。

このセミナーでは、IDEの歴史から始まり、無料で公開されているIntelliJ IDEA Community Editionを中心にIDEを解説します。

のはずが...

調子に乗りすぎて、IntelliJ IDEAの話に入る前に、  
スライドは既に30枚以上... (^\_^;;

ど～しよ～

# 前田の紹介

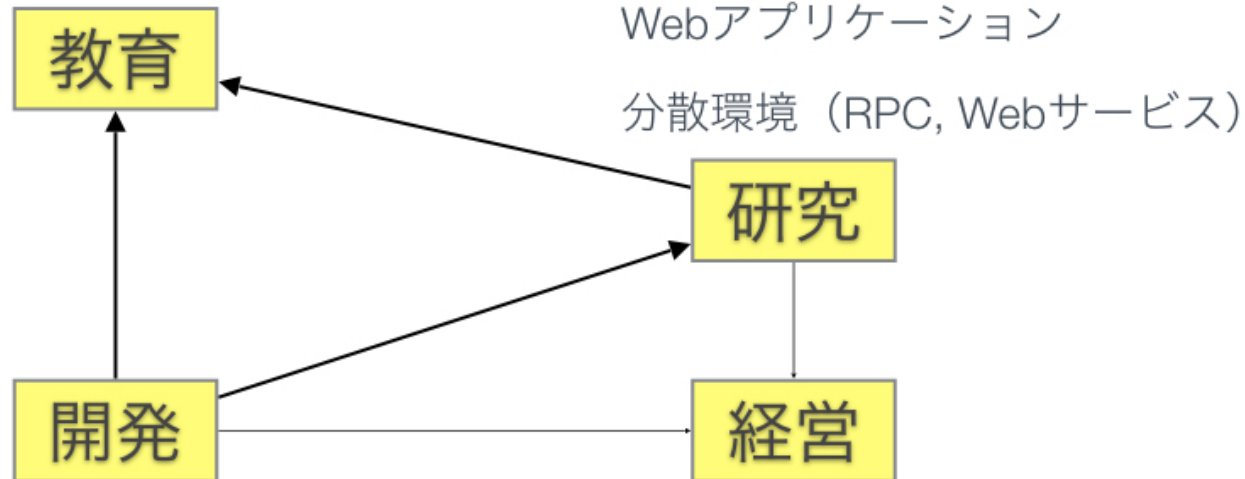
ソフトウェア開発  
プログラミング

プログラミング言語処理系

XML (パーザ, 応用)

Webアプリケーション

分散環境 (RPC, Webサービス)



プログラミング言語処理系  
(C#, VB, Java, C++, SQL,  
VHDL)

起業家精神・企業家精神

ベンチャー企業

オープンソースソフトウェア (OSS)

# Hello World! Program

- A "Hello, World!" program is a computer program that outputs or displays "Hello, World!" to a user. Being a very simple program in most programming languages, it is often used to illustrate the basic syntax of a programming language for a working program.[1] It is often the very first program people write when they are new to a language.

*C Programming Language*.<sup>[2]</sup> The example program from that book prints "hello, world" (without capital letters or exclamation mark), and was inherited<sup>[citation needed]</sup> from a 1974 [Bell Laboratories](#) internal memorandum by [Brian Kernighan](#), *Programming in C: A Tutorial*.<sup>[3]</sup>

```
#include <stdio.h>

main( )
{
    printf("hello, world\n");
}
```

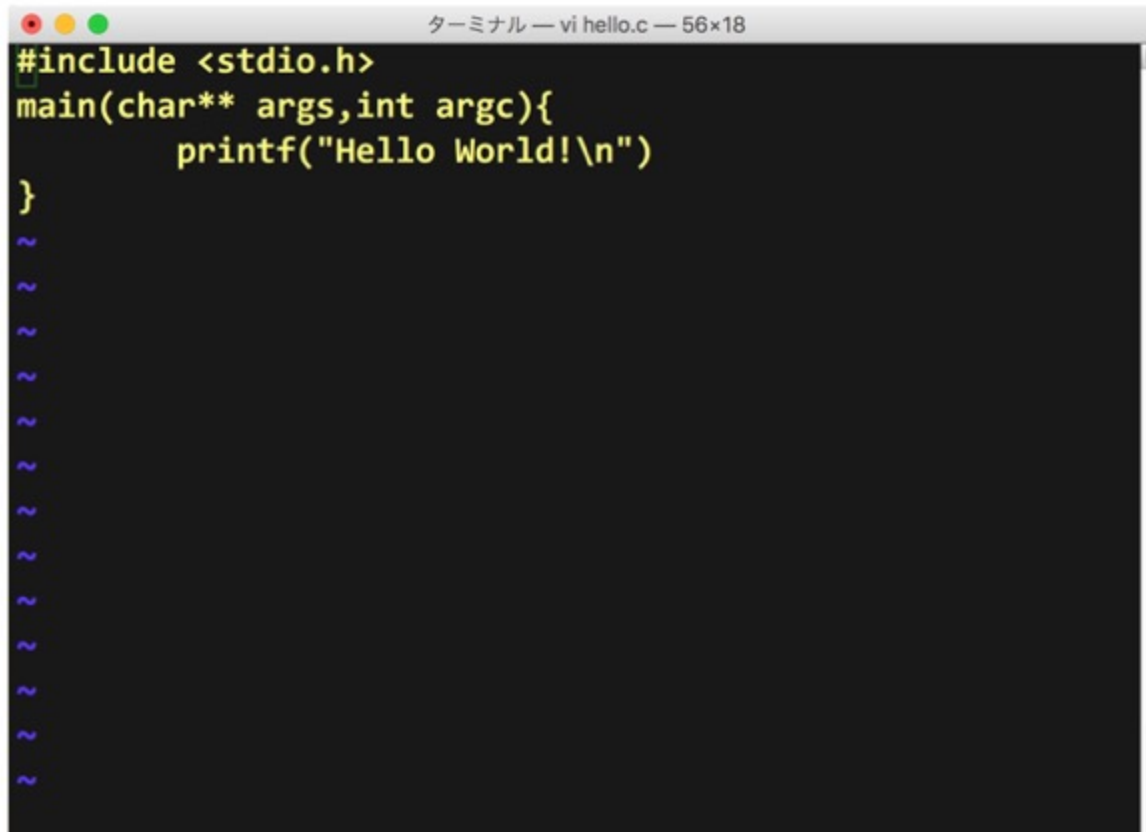
## VT100:Old Computer Terminal



<https://en.wikipedia.org/wiki/VT100>

CC BY 2.0

# Hello World! Program



A screenshot of a terminal window with a dark background. The title bar at the top reads "ターミナル — vi hello.c — 56x18". The code is written in yellow text. It starts with an include statement for `<stdio.h>`, followed by a `main` function that takes `char** args` and `int argc` as arguments. Inside the function, there is a `printf` statement that prints "Hello World!\n". The code ends with a closing brace. Below the code, there are several lines of tilde characters (~) in purple, indicating that the code is wrapped or that the terminal is in a state where it shows the continuation of the code.

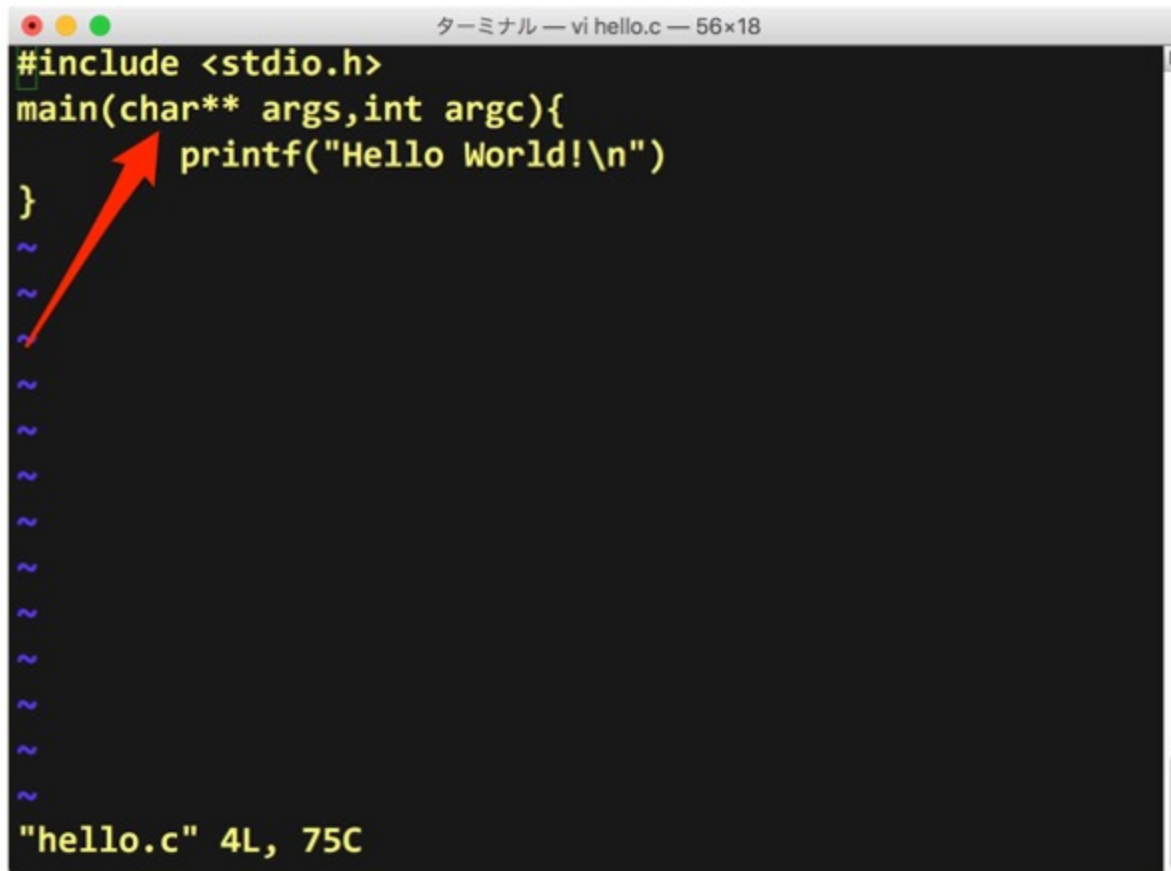
```
#include <stdio.h>
main(char** args,int argc){
    printf("Hello World!\n")
}
~
~
~
~
~
~
~
~
~
~
~
~
```

# Hello World! Program

```
ターミナル — zsh — 56x18
[kaz]% cc hello.c
hello.c:2:1: warning: type specifier missing, defaults
      to 'int' [-Wimplicit-int]
main(char** args,int argc){
^
hello.c:2:1: error: first parameter of 'main'
      (argument count) must be of type 'int'
hello.c:2:1: error: second parameter of 'main'
      (argument array) must be of type 'char **'
hello.c:3:26: error: expected ';' after expression
      printf("Hello World!\n")
                          ^
                          ;
1 warning and 3 errors generated.
[kaz]%
```



# Hello World! Program



A screenshot of a terminal window titled "ターミナル — vi hello.c — 56x18". The window displays the source code for a C program. The code is as follows:

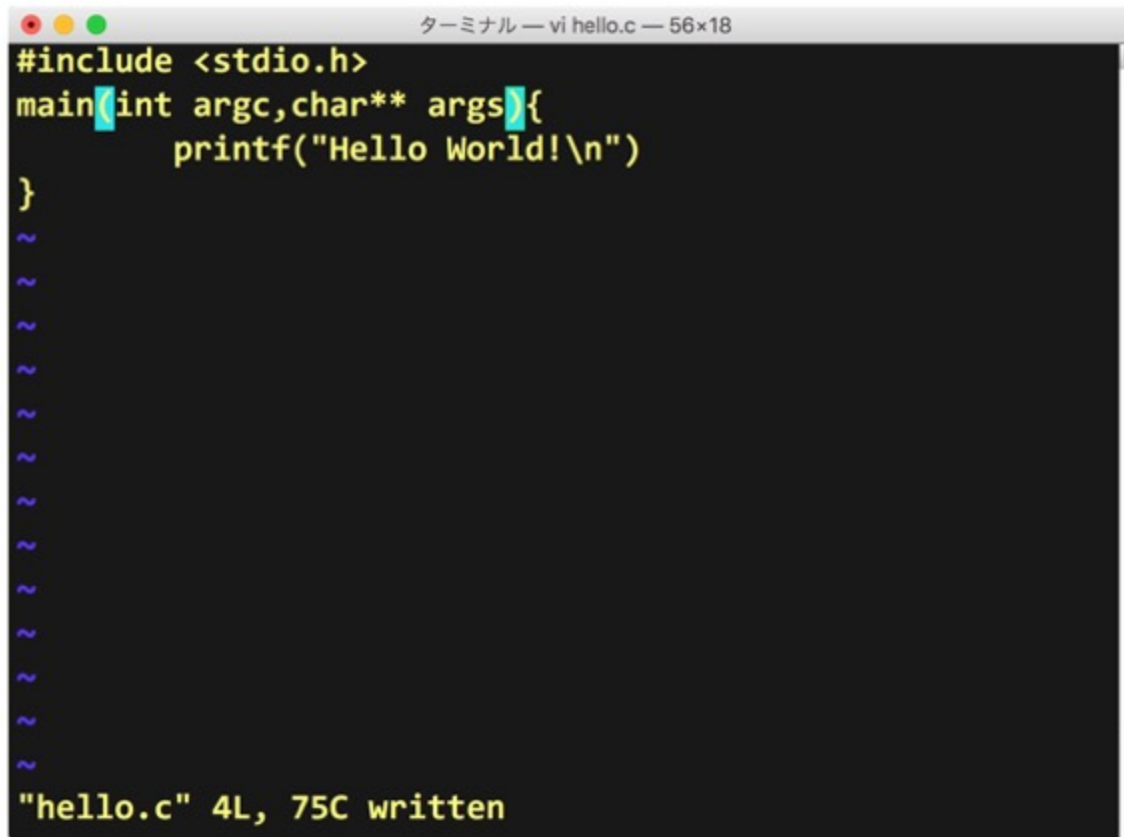
```
#include <stdio.h>
main(char** args,int argc){
    printf("Hello World!\n")
}
```

The code is written in yellow text on a black background. A red arrow points from the closing curly brace of the `main` function to the `printf` statement. The terminal shows several tilde (~) characters on the left side, indicating the current line in the file. At the bottom of the terminal, the status bar shows `"hello.c" 4L, 75C`.

# PDP11/23でUNIX V7の頃 (メモリ64KB)

```
ターミナル — -sh — 56x19
$ ed hello.c
75
1,$n
1      #include <stdio.h>
2      main(char** args,int argc){
3          printf("Hello World!\n")
4      }
2c
main(int argc,char** argv){
.
1,$n
1      #include <stdio.h>
2      main(int argc,char** argv){
3          printf("Hello World!\n")
4      }
w
75
q
$
```

# Hello World! Program

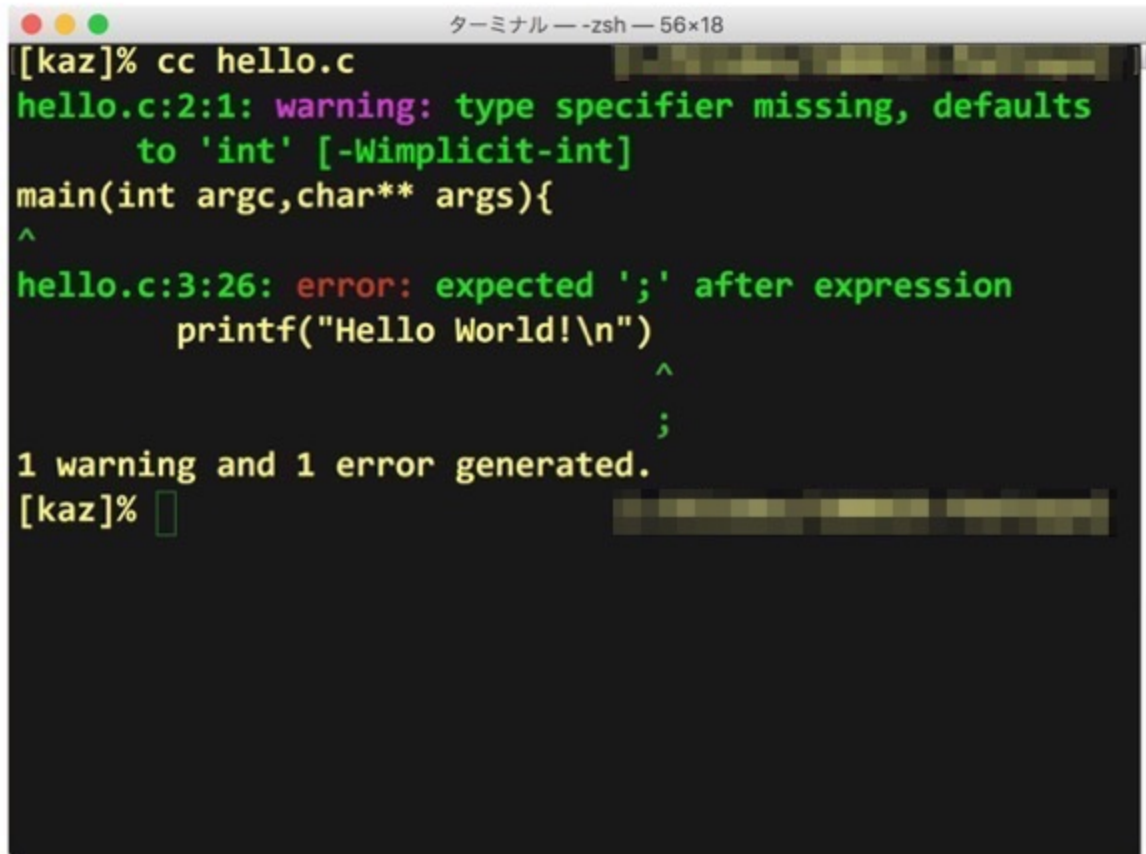


A screenshot of a terminal window with a dark background. The window title bar at the top shows three colored window control buttons (red, yellow, green) on the left and the text "ターミナル — vi hello.c — 56x18" on the right. The terminal displays the following C code:

```
#include <stdio.h>
main(int argc, char** args){
    printf("Hello World!\n")
}
```

Below the code, there are several lines of tilde (~) characters, indicating that the output of the program is being scrolled through. At the bottom of the terminal, the message `"hello.c" 4L, 75C written` is displayed.

# Hello World! Program



```
ターミナル — zsh — 56x18
[kaz]% cc hello.c
hello.c:2:1: warning: type specifier missing, defaults
      to 'int' [-Wimplicit-int]
main(int argc, char** args){
^
hello.c:3:26: error: expected ';' after expression
      printf("Hello World!\n")
                        ^
                        ;
1 warning and 1 error generated.
[kaz]% 
```

# Hello World! Program


[illegible]

# Hello World! Program

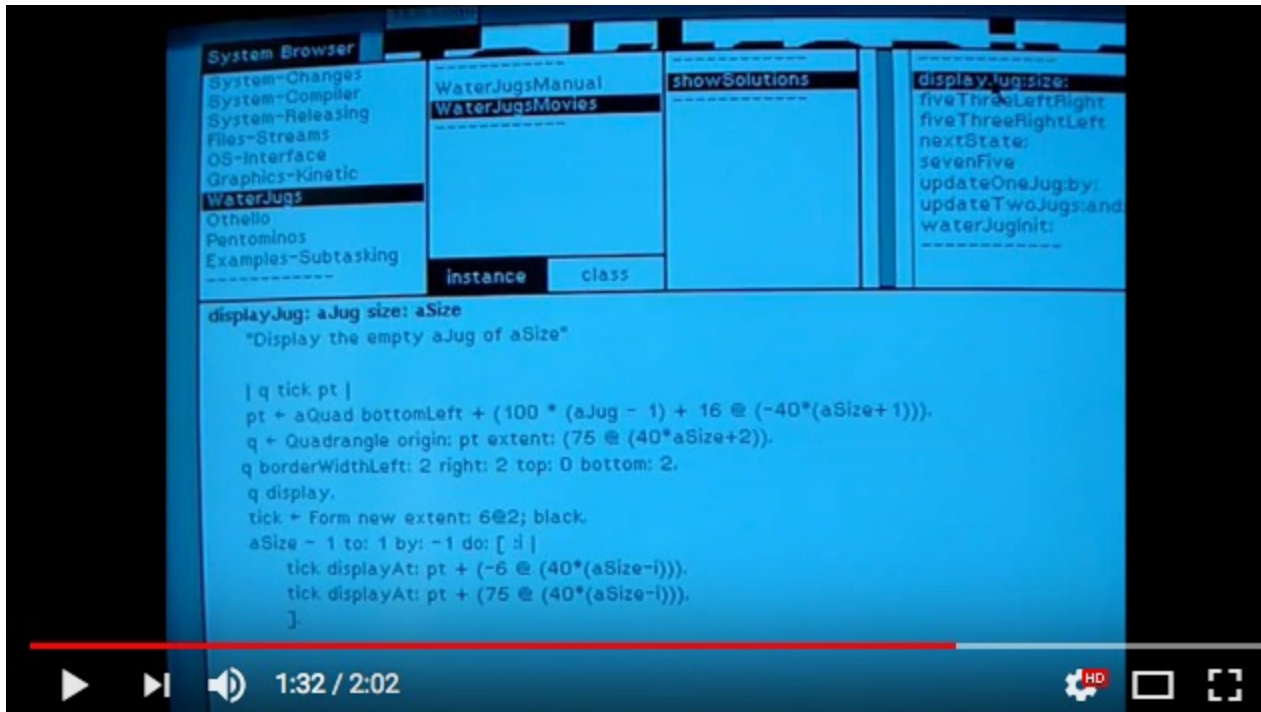
[illegible]

# Hello World! Program

```
ターミナル — zsh — 56x18
[kaz]% vi hello.c
[kaz]% cc hello.c
hello.c:2:1: warning: type specifier missing, defaults
      to 'int' [-Wimplicit-int]
main(int argc, char** args){
^
1 warning generated.
[kaz]% vi hello.c
[kaz]% cc hello.c
[kaz]% a.out
zsh: command not found: a.out
[kaz]% ./a.out
Hello World!
[kaz]%
```



# GUI (Graphical User Interface) の到来



Tektronix 4404 Smalltalk Demo

<https://www.youtube.com/watch?v=8yxCJfayW-8>

クリエイティブ・コモンズ表示ライセンス

CC BY



# Tektronix 4404

- Motorola 68010 32bit microprocessor
- 1 Mbyte high-speed dynamic RAM memory
- Virtual memory management for 8 Mbyte address space
- 640x480 monochrome, bit-mapped display
- 1024x1024 display bit-map with smooth panning
- 40 Mbyte, 5.25" hard disk and 320 Kbyte, 5.25" floppy disk
- (option) Franz LISP programming language
- (option) Prolog programming language
- (option) EMACS Editor

<http://www.wirfs-brock.com/allen/files/tek/4404-Flyer.pdf> より抜粋

Commercial, sold for \$14,950

<http://www.merlintec.com/swiki/hardware/26.html>

# 統合開発環境

IDE (Integrated Development Environment)

ソフトウェアの開発環境

従来、コンパイラ、テキストエディタ、デバッガなどがばらばらで利用していたものをひとつの対話型操作環境（多くはGUI）から利用できるようにしたもの。

<https://ja.wikipedia.org/wiki/統合開発環境>

# Integrated development environment

An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development.

An IDE normally consists of a source code editor, build automation tools, and a debugger.

Most modern IDEs have **intelligent code completion**.

[https://en.wikipedia.org/wiki/Integrated\\_development\\_environment](https://en.wikipedia.org/wiki/Integrated_development_environment)

より

# 統合開発環境の特徴

- プロジェクト管理
- バージョン管理
- GUIの作成
- チーム開発
- 作成補助
- ビルド、デバッグ補助

<https://ja.wikipedia.org/wiki/統合開発環境より>

## 代表的なIDE（統合開発環境）

- Visual Studio Code
- Eclipse
- NetBeans
- IntelliJ IDEA

# Visual Studio Code

Visual Studio Codeはソースコードエディタである。マイクロソフトにより開発され、Windows、Linux、macOS上で動作する。デバッグ、Gitクライアントの統合、シンタックスハイライト、インテリセンス、スニペット、リファクタリングなどの機能を持つ。カスタマイズもでき、利用者はエディタのテーマやキーボードショートカット等を変更できる。

## 歴史

Visual Studio Codeは、マイクロソフトの開発者会議Build 2015（2015年4月29日）にて公開され、プレビュー版がリリースされた。

2015年11月18日、Visual Studio Codeはリリースされ、その一部ソースコードがGitHubにMIT Licenseのもとで公開された。その際、拡張機能のサポートも追加された。

[https://ja.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://ja.wikipedia.org/wiki/Visual_Studio_Code) より

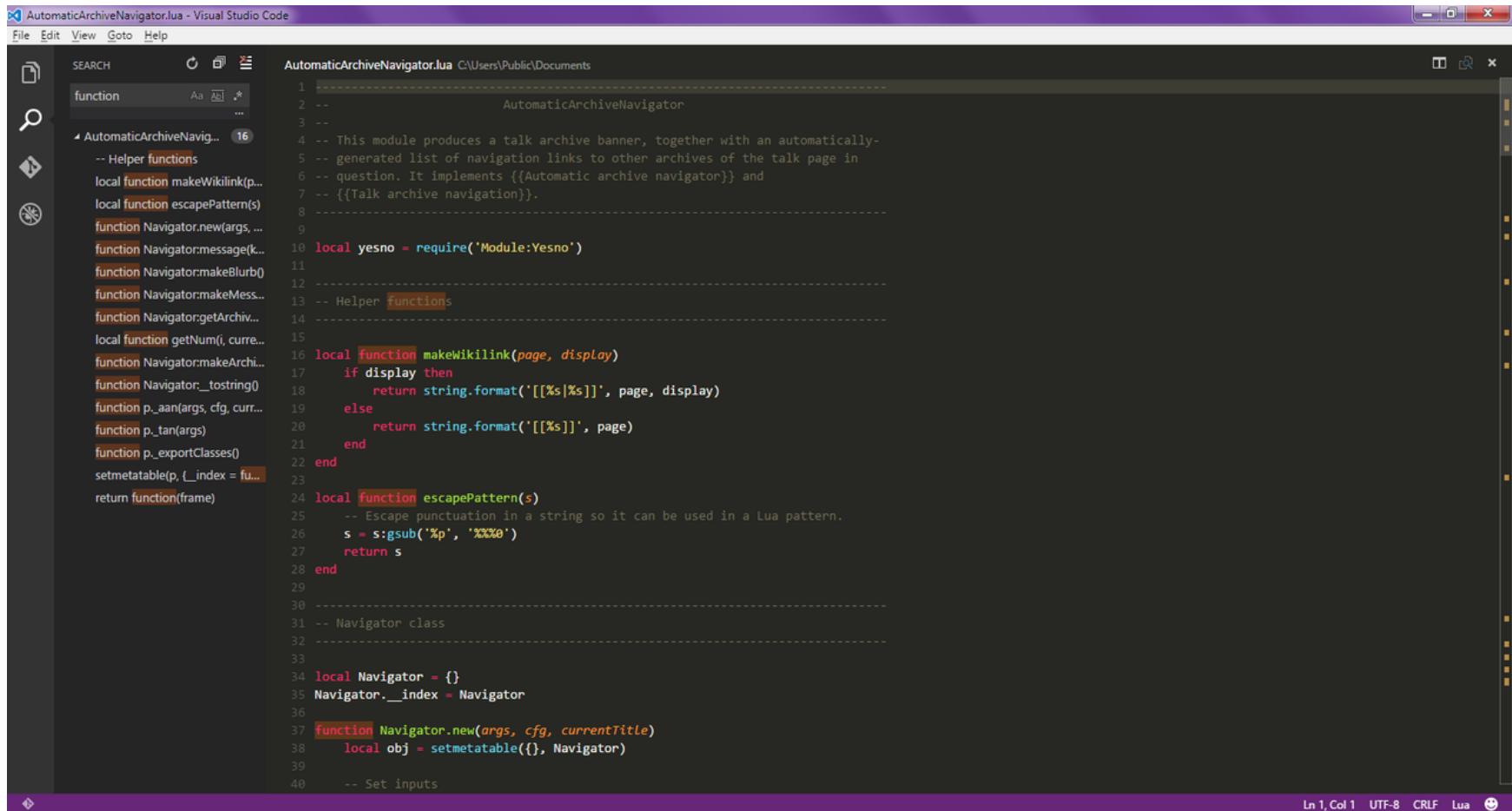
# Visual Studio Code

Visual Studio Code is a source code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring.

It is also customizable, so users can change the editor's theme, keyboard shortcuts, and preferences. It is free and open-source

[https://en.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://en.wikipedia.org/wiki/Visual_Studio_Code) より

# Visual Studio Code



```
AutomaticArchiveNavigator.lua C:\Users\Public\Documents
1 -----
2 -- AutomaticArchiveNavigator
3 --
4 -- This module produces a talk archive banner, together with an automatically-
5 -- generated list of navigation links to other archives of the talk page in
6 -- question. It implements {{Automatic archive navigator}} and
7 -- {{Talk archive navigation}}.
8 -----
9
10 local yesno = require('Module:Yesno')
11
12 -----
13 -- Helper functions
14 -----
15
16 local function makeWikilink(page, display)
17     if display then
18         return string.format('[[%s|]]', page, display)
19     else
20         return string.format('[[%s]]', page)
21     end
22 end
23
24 local function escapePattern(s)
25     -- Escape punctuation in a string so it can be used in a Lua pattern.
26     s = s:gsub('%p', '%%%0')
27     return s
28 end
29
30 -----
31 -- Navigator class
32 -----
33
34 local Navigator = {}
35 Navigator.__index = Navigator
36
37 function Navigator.new(args, cfg, currentTitle)
38     local obj = setmetatable({}, Navigator)
39
40     -- Set inputs
```

[https://ja.wikipedia.org/wiki/Visual\\_Studio\\_Code#/media/File:Visual\\_Studio\\_Code\\_0.10.1\\_on\\_Windows\\_7,\\_with\\_search.png](https://ja.wikipedia.org/wiki/Visual_Studio_Code#/media/File:Visual_Studio_Code_0.10.1_on_Windows_7,_with_search.png)

MIT License



# Java Tools and Technologies Landscape Report 2016: Trends and Historical data

July 14, 2016

|      | Eclipse | IntelliJ IDEA | NetBeans |
|------|---------|---------------|----------|
| 2012 | 62%     | 24%           | 14%      |
| 2014 | 54%     | 33%           | 10%      |
| 2016 | 41%     | 46%           | 10%      |

<https://zeroturnaround.com/rebellabs/java-tools-and-technologies-landscape-2016-trends/>

他の調査でも2016年で, Eclipse = IntelliJ IDEA

<http://www.baeldung.com/java-ides-2016>

# Android Studio

Android Studio（アンドロイド スタジオ）は、Googleが提供するAndroidプラットフォームに対応する統合開発環境（IDE）。JetBrains社が開発したIntelliJ IDEAをベースにAndroid開発に最適化されており、Windows、macOSおよびLinux用が存在する。

2013年5月16日、Google I/Oカンファレンスにおいて発表され[5]、同年5月にアーリーアクセスプレビュー版がリリースされた[4]。その後、2014年6月に公開されたバージョン0.8.0からは、開発段階がベータ版へ移行した[6]。

更にその後、2014年12月8日に正式バージョン1.0.0が公開され、従来EclipseとADT（Android Developer Tools）により実現されていた開発環境を、Android Studioで実現できるようになった。2015年末をもってEclipseとADTのサポートが打ち切れ[7]、Android開発環境はAndroid Studioに完全移行された。

# Developer Survey Results 2016

by stack overflow

## VII. Development Environments

<https://insights.stackoverflow.com/survey/2016>

# Java IDE: Eclipse, IntelliJ, NetBeans 3大 Java IDEを比較(2017年版)

@jhChoi 2017年09月28日に更新

シンプルさ (学習コスト)

- Eclipse: △  
package explorer 画面、debugging画面、team synchronizing画面など、画面数が多く、最初は混乱する。
- IntelliJ: ○  
Eclipseよりはシンプル。
- NetBeans: ◎  
わかりやすく初心者にも良い。

<https://qiita.com/jhChoi/items/d60fe8cf396dfc11d51d>

# Eclipseの歴史

1990年代後半の状況は、JBuilder、VisualCafe（英語版）、そしてIBMのVisualAge、PFUのteikadeなど第1世代のJava開発ツールが存在している。IBMは様々なプラットフォームの製品を抱えていることから、Javaのマルチプラットフォームの可能性に注目していた。単なるVisualAgeの代替ではなく、IBMや他社のツールを統合するための共通プラットフォームの開発という基本構想の下、1998年11月にIBMカナダでプロジェクトが開始された。開発に携わったのは、VisualAgeの開発を行ったObject Technology International (OTI) 研究所である。

その後、IBMはこのプラットフォームに搭載するツールの開発のために組織の編成を行い、さらにオープンソース化することで新しい開発者の引き込みを図った。2001年11月、IBMはEclipseをオープンソース化するとともに、

[https://ja.wikipedia.org/wiki/Eclipse\\_\(統合開発環境\)](https://ja.wikipedia.org/wiki/Eclipse_(統合開発環境))

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE(6). It contains a base workspace and an extensible plug-in system for customizing the environment.

(6)<https://zeroturnaround.com/rebellabs/ides-vs-build-tools-how-eclipse-intellij-idea-netbeans-users-work-with-maven-ant-sbt-gradle/>

## History

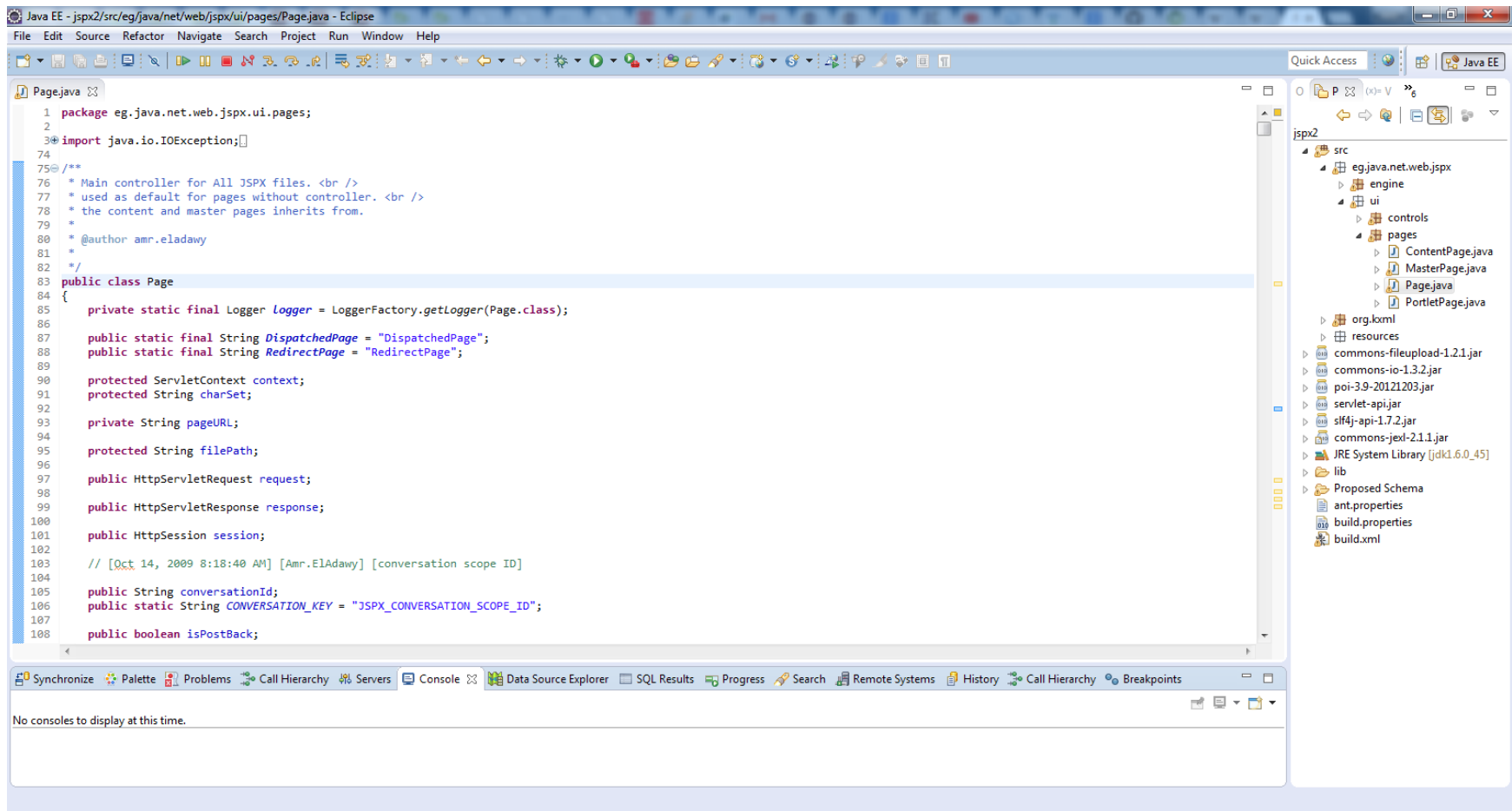
Eclipse was inspired by the Smalltalk-based VisualAge family of integrated development environment (IDE) products.

[https://en.wikipedia.org/wiki/Eclipse\\_\(software\)](https://en.wikipedia.org/wiki/Eclipse_(software)) より

# Eclipseの名前

| バージョン | リリース日        | コードネーム   | 由来  |
|-------|--------------|----------|---|
| 3.2   | 2006/06/30   | Callisto | 木星の第4衛星 <b>カリスト</b> 。ガリレオ衛星の1つ。                             |
| 3.3   | 2007/06/29   | Europa   | 木星の第2衛星 <b>エウロパ</b> 。ガリレオ衛星の1つ。                             |
| 3.4   | 2008/06/25   | Ganymede | 木星の第3衛星 <b>ガニメデ</b> 。ガリレオ衛星の1つ。                             |
| 3.5   | 2009/06/24   | Galileo  | ガリレオ衛星の発見者 <b>ガリレオ・ガリレイ</b> 。                               |
| 3.6   | 2010/06/23   | Helios   | ギリシア神話の太陽神 <b>ヘーリオス</b> 。                                   |
| 3.7   | 2011/06/22   | Indigo   | <b>ニュートン</b> が <b>プリズム</b> によって光を7色に分解できることを発前。 <b>藍色</b> 。 |
| 4.2   | 2012/06/27   | Juno     | ローマ神話に出てくる、女性と結婚を守護する女神 <b>ユノー</b> 。を意味する"June"の由来。         |
| 4.3   | 2013/06/26   | Kepler   | 「 <b>ケプラーの法則</b> 」で有名なドイツの天文学者 <b>ヨハネス・ケプラー</b> 。           |
| 4.4   | 2014/06/25   | Luna     | <b>月</b> を意味する。   |
| 4.5   | 2015/06/24   | Mars     | <b>火星</b> を意味する。  |
| 4.6   | 2016/06/22   | Neon     | 元素の一つ、 <b>ネオン</b> を意味する。                                    |
| 4.7   | 2017/06/28   | Oxygen   | 元素の一つ、 <b>酸素</b> を意味する。                                     |
| 4.8   | 2018/06/27予定 | Photon   | <b>光子</b> を意味する。  |

[https://ja.wikipedia.org/wiki/Eclipse\\_\(統合開発環境\)](https://ja.wikipedia.org/wiki/Eclipse_(統合開発環境))



[https://ja.wikipedia.org/wiki/Eclipse\\_\(統合開発環境\)#/media/File:Eclipse\\_4.5.png](https://ja.wikipedia.org/wiki/Eclipse_(統合開発環境)#/media/File:Eclipse_4.5.png)



# Eclipse日本語版

Pleiades - 日本語化プラグイン Eclipse, IntelliJ, PhpStorm...

<http://mergedoc.osdn.jp/>

- Pleiades All in One ダウンロード
- Pleiades プラグイン・ダウンロード

Pleiades は Eclipse や Android Studio のような Java アプリケーションを日本語化するためのツール

Jetbrains社の IDE (IntelliJ IDEA, PhpStorm, RubyMine, PyCharm など) の日本語化も可能

# IntelliJと前田の出会い ♡

- JavaScriptのIDEを探していた
- WebStormに出会い、そして惚れた ♡
- 同じ会社の製品にIntelliJ IDEAがある
- あれ？ IntelliJ IDEAって聞いたことある  
どこかのオープンソースイベントだったような...



# WebStormの価格 (2018年5月)

## ビジネス向け

- US \$129.00 / 1st year
- US \$103.00 / 2nd year
- US \$77.00 / 3rd yearとそれ以降

## 個人向け

- US \$59.00 / 1st year
- US \$47.00 / 2nd year
- US \$35.00 / 3rd yearとそれ以降

## 無料の対象

- 学生と先生
- 教育用 (クラスライセンス)
- 非商用オープンソースプロジェクト

# クラスライセンスを取得

Type: Classroom License

Reference No\*: LC-xxxxx-Dxxxxx

Date of Issue: 25 November 2010

Expiration Date: 25 November 2011

Number of Authorized Users: not limited

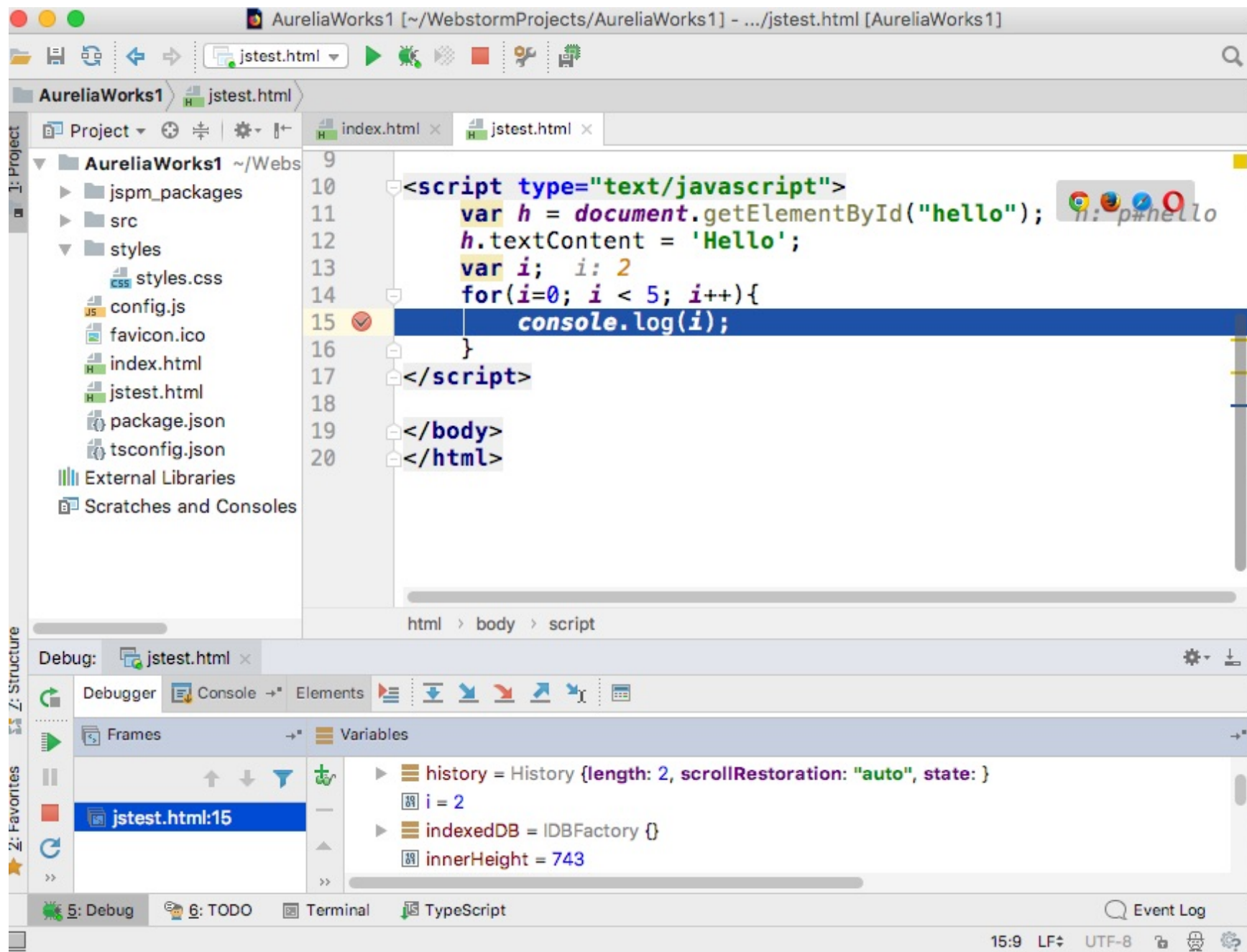
===== LICENSEE =====

Name: Chubu University

===== SOFTWARE PRODUCT =====

Product Name: WebStorm

# WebStormでデバッグ中



# JetBrains社の製品

- IntelliJ IDEA  
Ultimate版 と Community版
- WebStorm
- PhpStorm
- PyCharm  
Professional版 と Community版
- RubyMine
- CLion
- GoLand

## Education向けの無料版

- Java Edu
- Python Edu
- Kotlin Edu

# IntelliJ IDEA Ultimateの価格 (2018年5月)

## ビジネス向け

- US \$499.00 / 1st year
- US \$399.00 / 2nd year
- US \$299.00 / 3rd yearとそれ以降

## 個人向け

- US \$149.00 / 1st year
- US \$119.00 / 2nd year
- US \$89.00 / 3rd yearとそれ以降

## 無料の対象

- 学生と先生
- 教育用 (クラスライセンス)
- 非商用オープンソースプロジェクト

# IntelliJ IDEA

IntelliJ IDEA(インテリジェイ アイディア)は、チェコに本社を置く JetBrains社が開発した、Javaなど数多くのプログラミング言語に対応した統合開発環境である。

リファクタリング機能をJava用の統合開発環境としては初めて搭載したことでも知られる。変数に型のないプログラミング言語に対してもリファクタリングを提供している。

ZeroTurnaroundの調査によると、Javaの統合開発環境としてはEclipseに続いて2番目に人気である (2011年はシェア22%、2012年はシェア28%。

[https://ja.wikipedia.org/wiki/IntelliJ\\_IDEA](https://ja.wikipedia.org/wiki/IntelliJ_IDEA)

2012年:24%, 2014年:33%, 2016年:46%

<https://zeroturnaround.com/rebellabs/java-tools-and-technologies-landscape-2016-trends/>



# History of IntelliJ IDEA

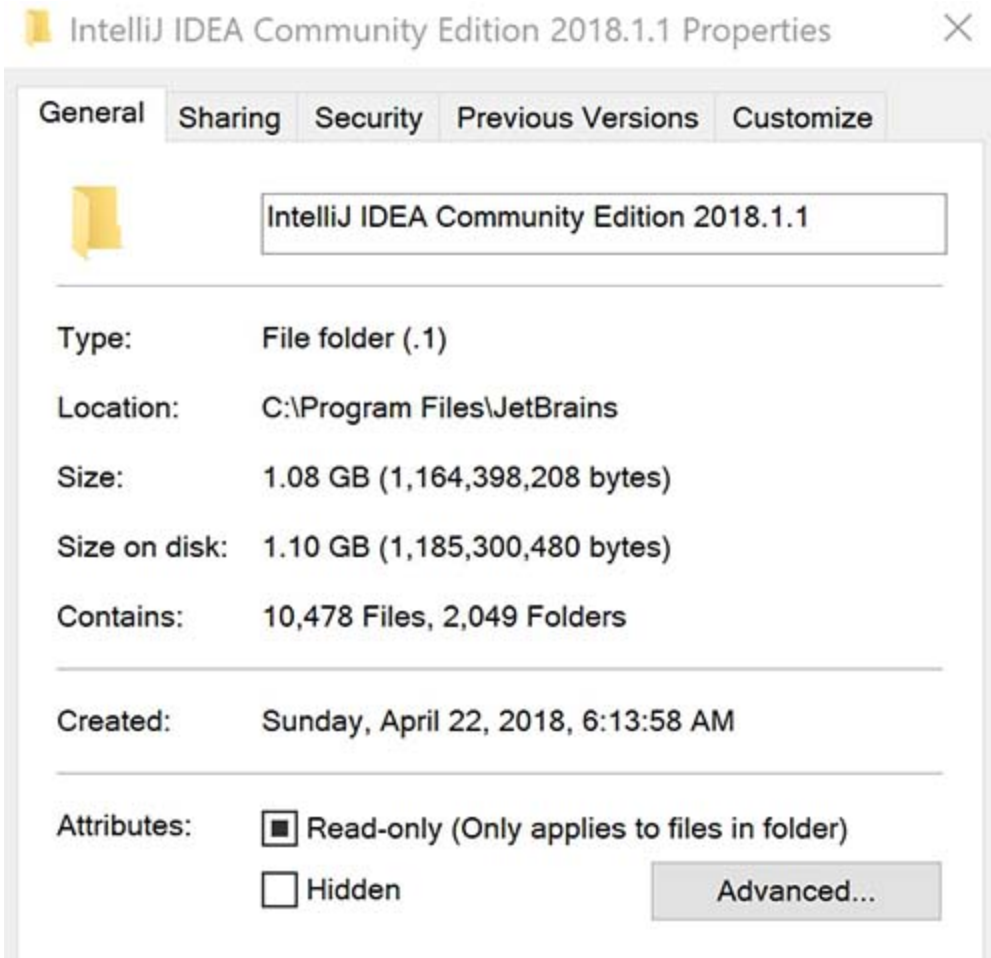
The first version of IntelliJ IDEA was released in January 2001, and was one of the first available Java IDEs with advanced code navigation and code refactoring capabilities integrated.

In December 2014, Google announced version 1.0 of Android Studio, an open source IDE for Android apps, based on the open source community edition of IntelliJ IDEA.

Other development environments based on IntelliJ's framework include AppCode, CLion, PhpStorm, PyCharm, RubyMine, WebStorm.

[https://en.wikipedia.org/wiki/IntelliJ\\_IDEA](https://en.wikipedia.org/wiki/IntelliJ_IDEA)

# IntelliJ IDEA Community版



# IntelliJ IDEA Community版

ソースコード

<https://github.com/JetBrains/intellij-community>



# IntelliJ IDEA Community版のデモ

# Developer Survey Results 2016

(by stack overflow)

(Technology) VII. Development Environments

<https://insights.stackoverflow.com/survey/2016>

## 今絶好調のIntelliJ IDEA

<https://zeroturnaround.com/rebellabs/java-tools-and-technologies-landscape-2016-trends/>

これからの活躍が楽しみです！