

# \* as a Code

## コンテンツをプログラミングする方法

Render any content with code

国立研究開発法人 産業技術総合研究所

**加藤 淳**

<http://junkato.jp/ja> (@arcatdmz)

2015/12/9 明治大学CE概論

# 加藤 淳



arcatdmz

<http://junkato.jp/ja>

研究テーマ

**Human-Computer Interaction** (人とコンピュータの関係改善)

とくに、**統合開発環境**を使いやすくする研究を通して  
より多くの人が快適にプログラミングできるようにしてきた

経歴

- **産総研** メディアインタラクション研究グループ 研究員 2014.4-
- **東京大学** 五十嵐健夫研究室 修士・博士 2009.4-2013.3
- **JST ERATO** 五十嵐プロジェクト 2008.4-2013.2 (学部4年～)
- **Adobe Research** Seattle インターン 2013.8-11
- **Microsoft Research** Asia/Redmond インターン 2012.1-4, 6-9
- **Microsoft Development** Software Engineer インターン 2009.8-10
- **東京大学エッジキャピタル** (ベンチャーキャピタル) インターン 2010.6-8

# digitalmuseum

<http://digitalmuseum.jp>

'00-

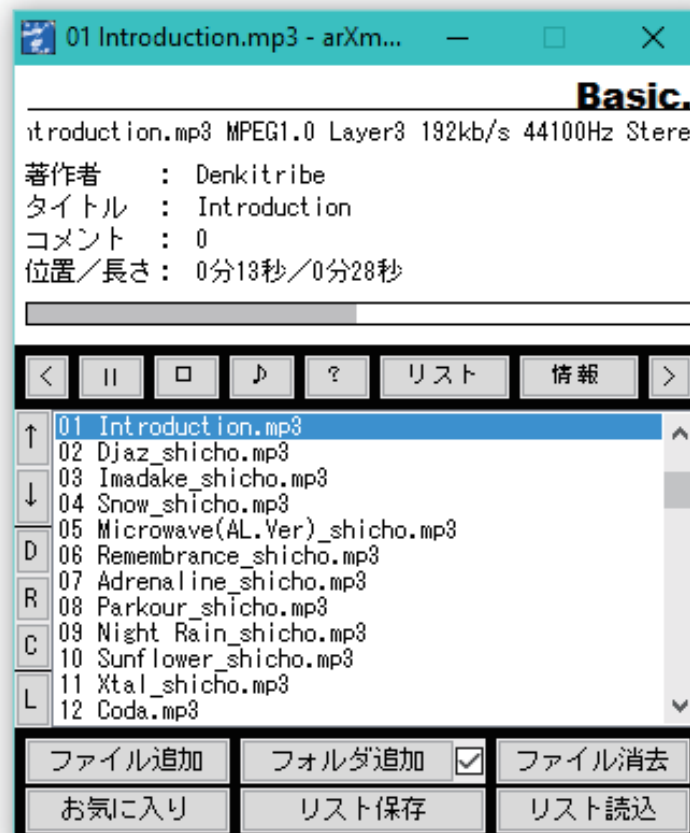
- **主宰**; 高校の頃から運営
- フリーウェアやデザインした作品を展示
- 友人との創作活動の拠点
- 産学交流の場づくり



# arX MusicPlayer

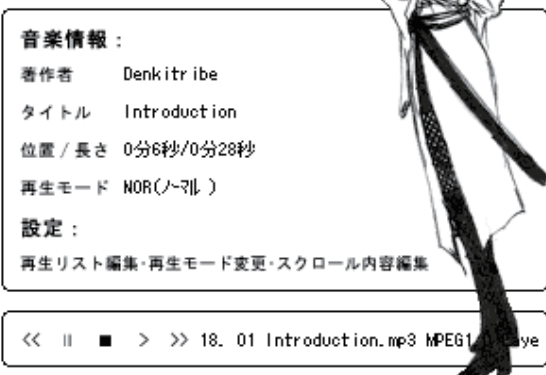
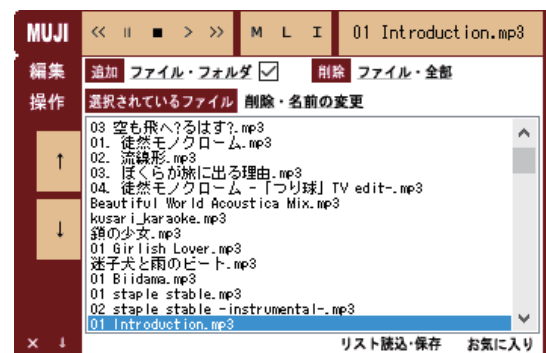
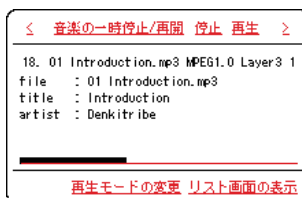
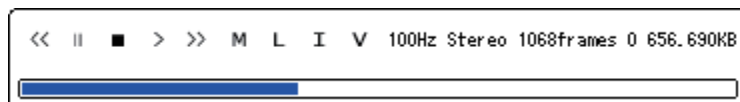
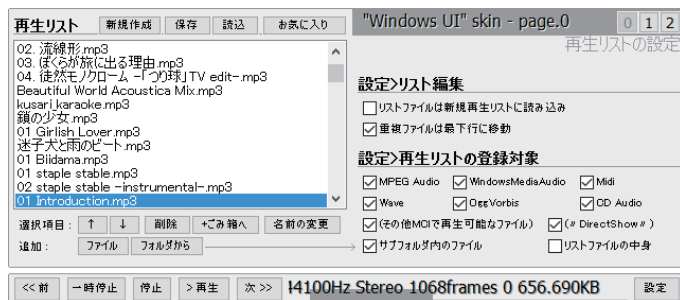
<http://digitalmuseum.jp/software/arxmp>

- 音楽プレイヤー
- Hot Soup Processor製
- 1万行くらい?
- 拡張機能・スキンに対応



# arX MusicPlayer

<http://digitalmuseum.jp/software/arxmp>



Special thanks to all arXmp users!

\* as a Code

# 時代背景: 20世紀

- **ディスプレイ**: 1024×768px (16ビットカラー)
- **CPU**: Pentium III 500MHz
- **メモリ**: 256MB (増設済み)
- **HDD**: 80GB
- **OS**: Windows 98 SE

**HIGH PERFORMANCE  
DESKTOP PC SYSTEM!!!**



# ~~無料アプリ~~フリーウェア

- 通信環境: **56kbps**
- “トップページ10KB以下”
- “CD-ROM650MBすごい”

---

2005/02	<a href="#">WindowsStart</a>	毎日コミュニケーションズ
2005/01	<a href="#">フリーウェア年鑑2005</a>	エンターブレイン
2005/01	<a href="#">WindowsStart</a>	毎日コミュニケーションズ
2004/11	<a href="#">ウィンドウズROM! 超オススメソフトBEST50 他</a>	毎日コミュニケーションズ
2004/08	<a href="#">Windows 100%</a>	晋遊舎
2004/08	<a href="#">iP!</a>	晋遊舎
2004/07	<a href="#">DiGi/USER</a>	宝島社
2004/06	<a href="#">ウィンドウズROM!</a>	毎日コミュニケーションズ
2004/06	<a href="#">DOS/V USER DX</a>	宝島社
2004/04	<a href="#">ネットプレイヤー vol.17</a>	セブン新社
2004/03	<a href="#">DOS/V USER DX</a>	宝島社
2003/10	<a href="#">WindowsPower フリーウェア5000</a>	エンターブレイン
2003/05	<a href="#">DOS/V magazine</a>	ソフトバンクパブリッシング
2003/05	<a href="#">WindowsPower JUMBO</a>	エンターブレイン
2003/03	<a href="#">WindowsPower フリーウェア4000</a>	エンターブレイン
2003/03	<a href="#">CD-ROM Fan フリーウェア Vol.4</a>	毎日コミュニケーションズ
2002/10	<a href="#">Windows 100%</a>	晋遊舎
2002/秋	<a href="#">WindowsPower フリーウェア3000</a>	エンターブレイン
2002/07	<a href="#">TECH Win</a> オンラインソフト新定番70連発	エンターブレイン
2002/04	<a href="#">TECH Win</a>	エンターブレイン
2002/04	<a href="#">iP!</a>	晋遊舎
2002/03	<a href="#">WindowsPower</a>	エンターブレイン
2002/03	<a href="#">ウィンドウズROM!</a>	毎日コミュニケーションズ
2002/03	<a href="#">Windows 100%</a>	晋遊舎
2002/02	<a href="#">DOS/V magazine</a>	ソフトバンクパブリッシング

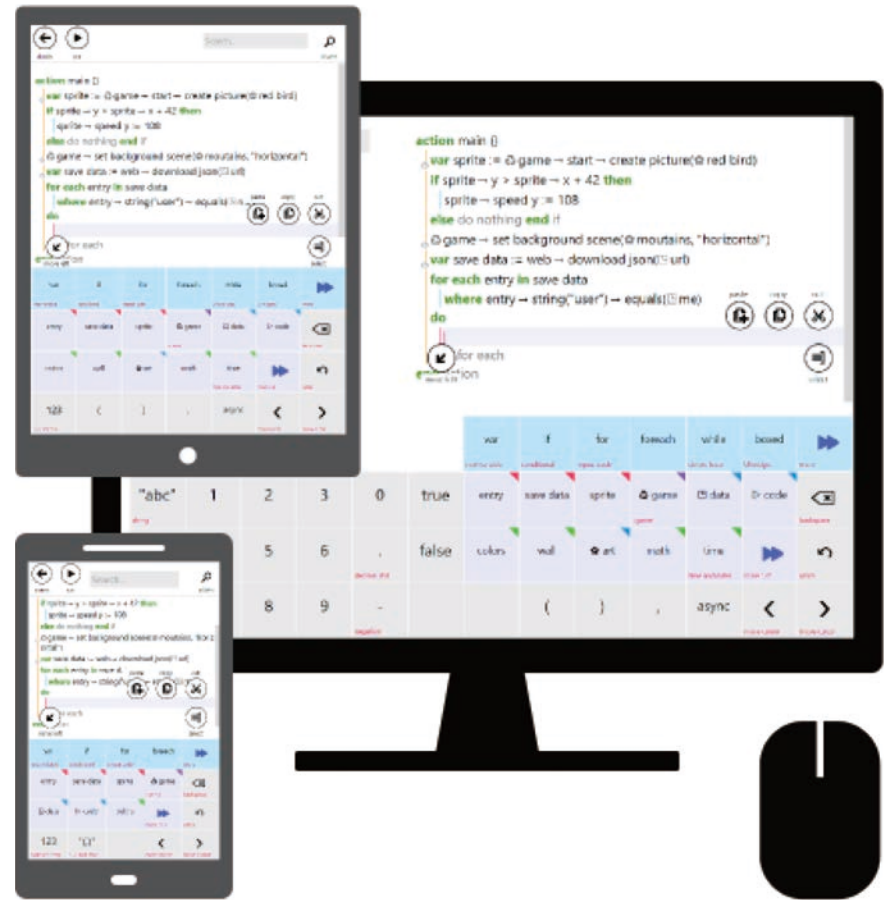
---



# 今?

- デスクトップPC
- ラップトップ
- スマートフォン
- タブレット
- スマートウォッチ
- スマートグラス

タッチ端末でも  
プログラミングできる時代



 Microsoft  
Touch Develop

<https://www.touchdevelop.com/>



# 統合開発環境 1964

Dartmouth Time-Sharing System (DTSS) [1964-]

- ソースコードの読み込み・保存、編集、コンパイル、実行が一通りできた初めての統合開発環境

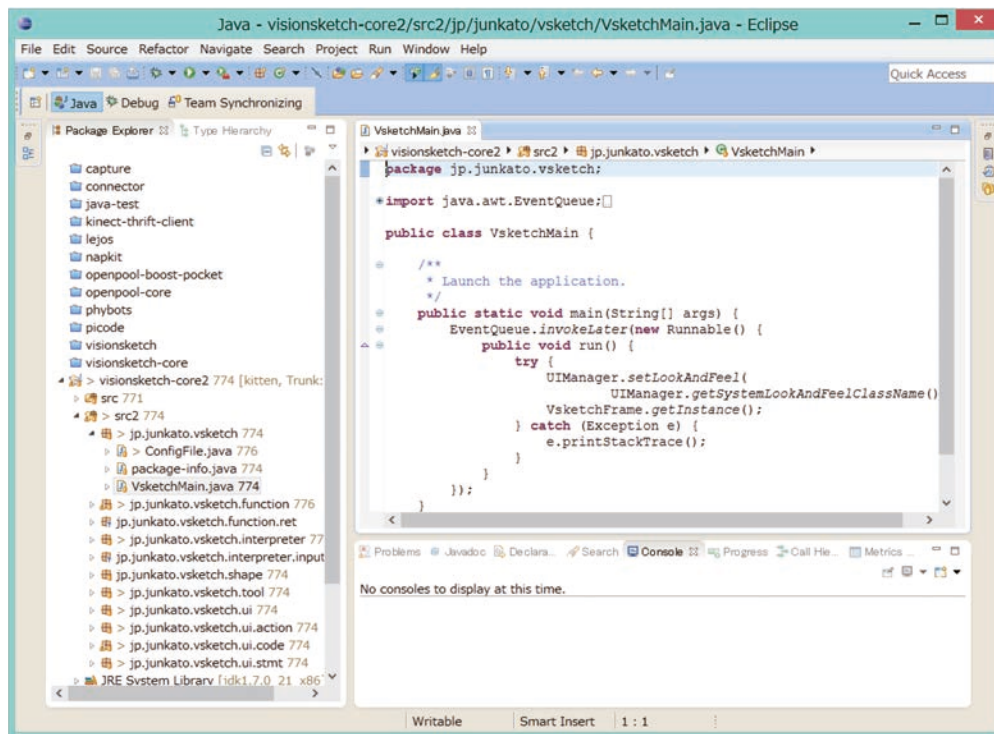
```
GW-BASIC 3.23
(C) Copyright Microsoft 1983,1984,1985,1986,1987,1988
60300 Bytes free
Ok
-
1LIST 2RUN< 3LOAD" 4SAVE" 5CONT< 6,"LPT1 7TRON< 8TROFF< 9KEY 0SCREEN
```

GW-BASIC

DTSSに似た  
Microsoft製方言

# 統合開発環境 2015

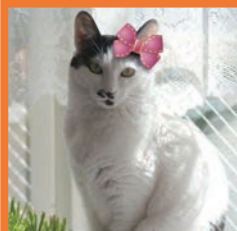
Visual Studio (Microsoft C) [1983-], Eclipse (IBM VisualAge) [1984-], Xcode [2003-], …



インタフェースの大部分が  
文字ベースという点では  
**50年前と同じ**

# 今日の講義の目標

- プログラミングって何だっけ?を知る
- プログラミングでコンテンツを作ってみる
- プログラミングの方法をプログラミングしてみる



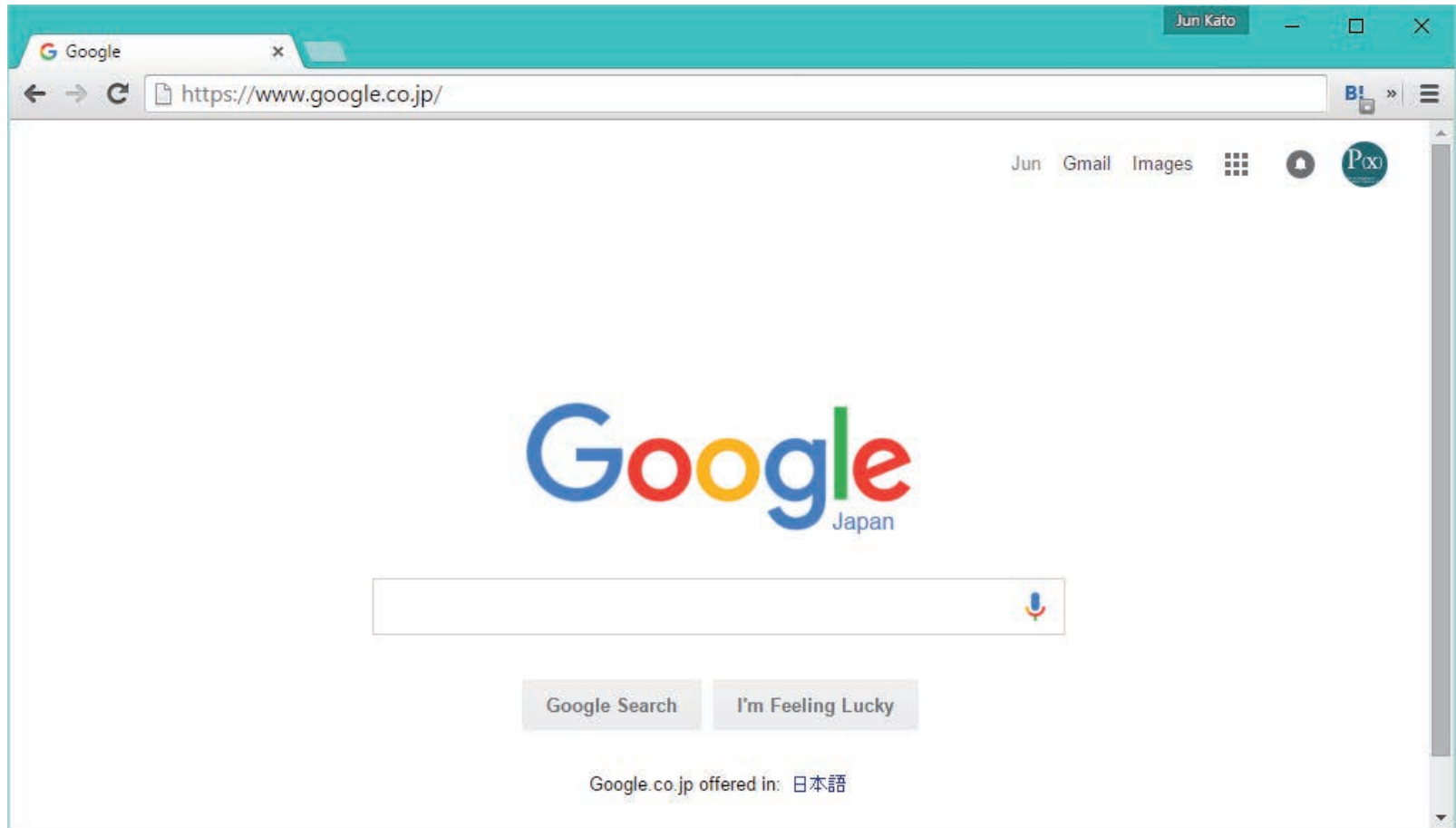
よろしくお願いします!

# CE概論

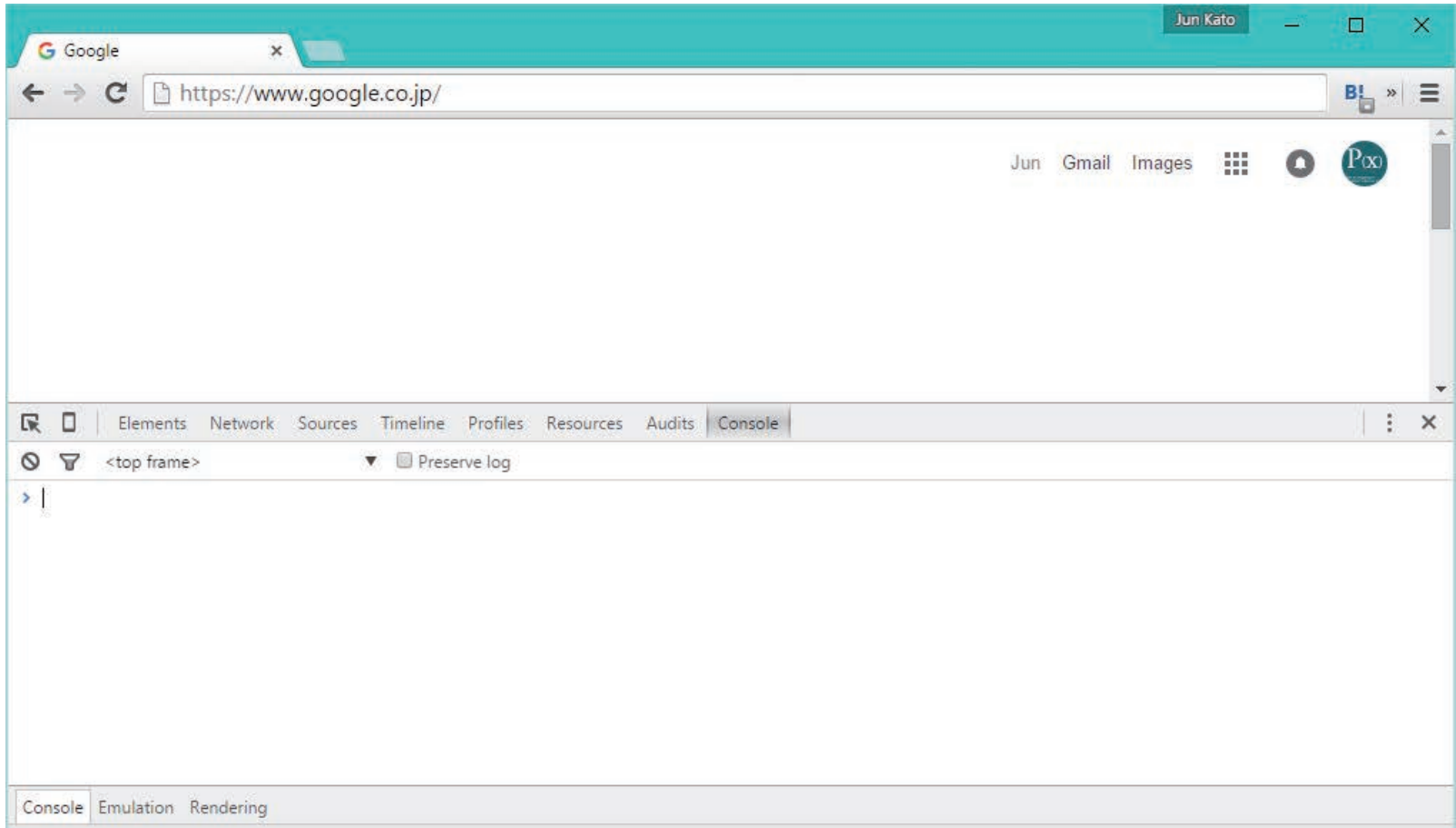
# プログラミングって 何だっけ？

\* as a Code —コンテンツをプログラミングする方法—

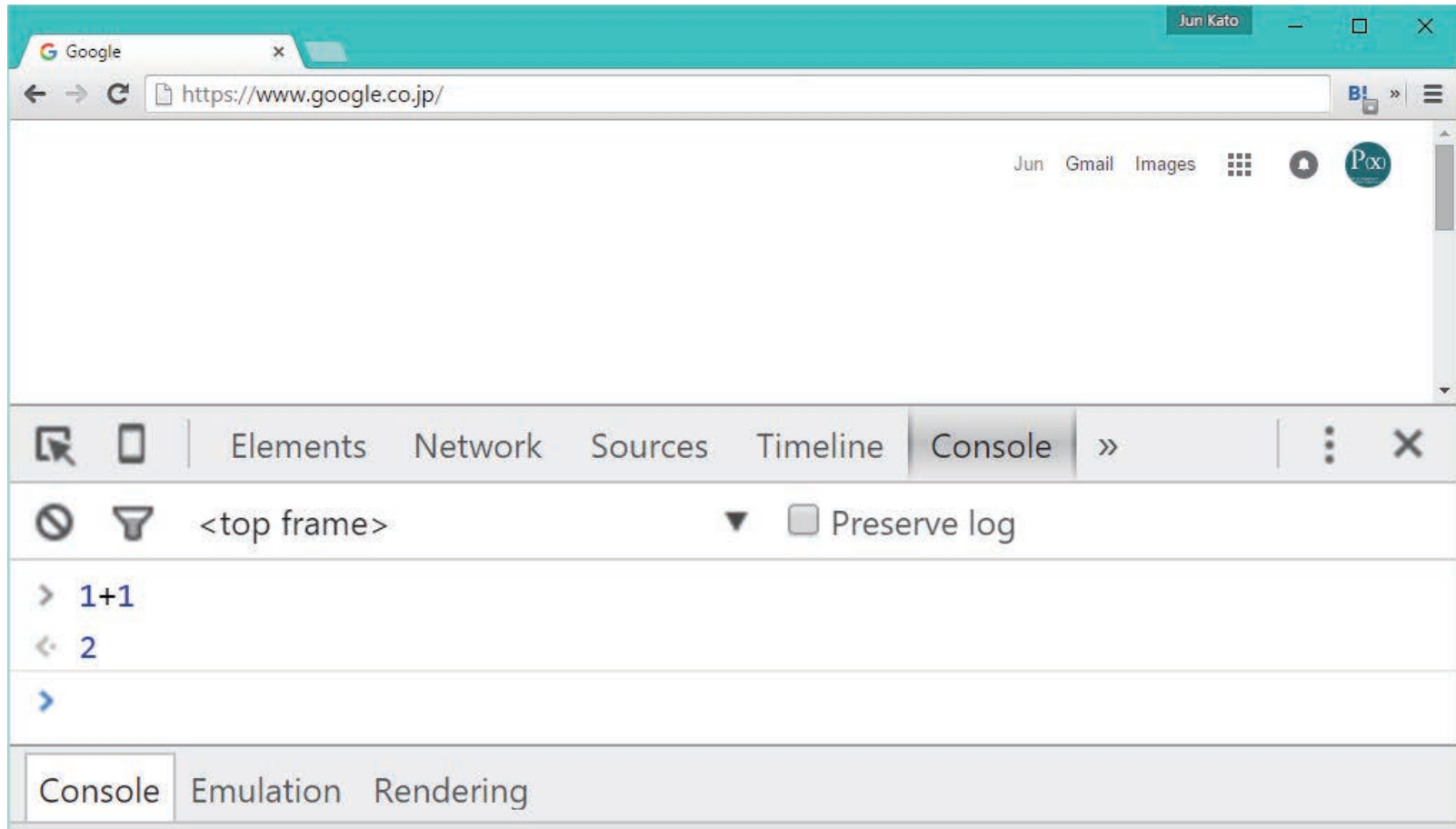
# ブラウザを開こう



# F12キーを押そう

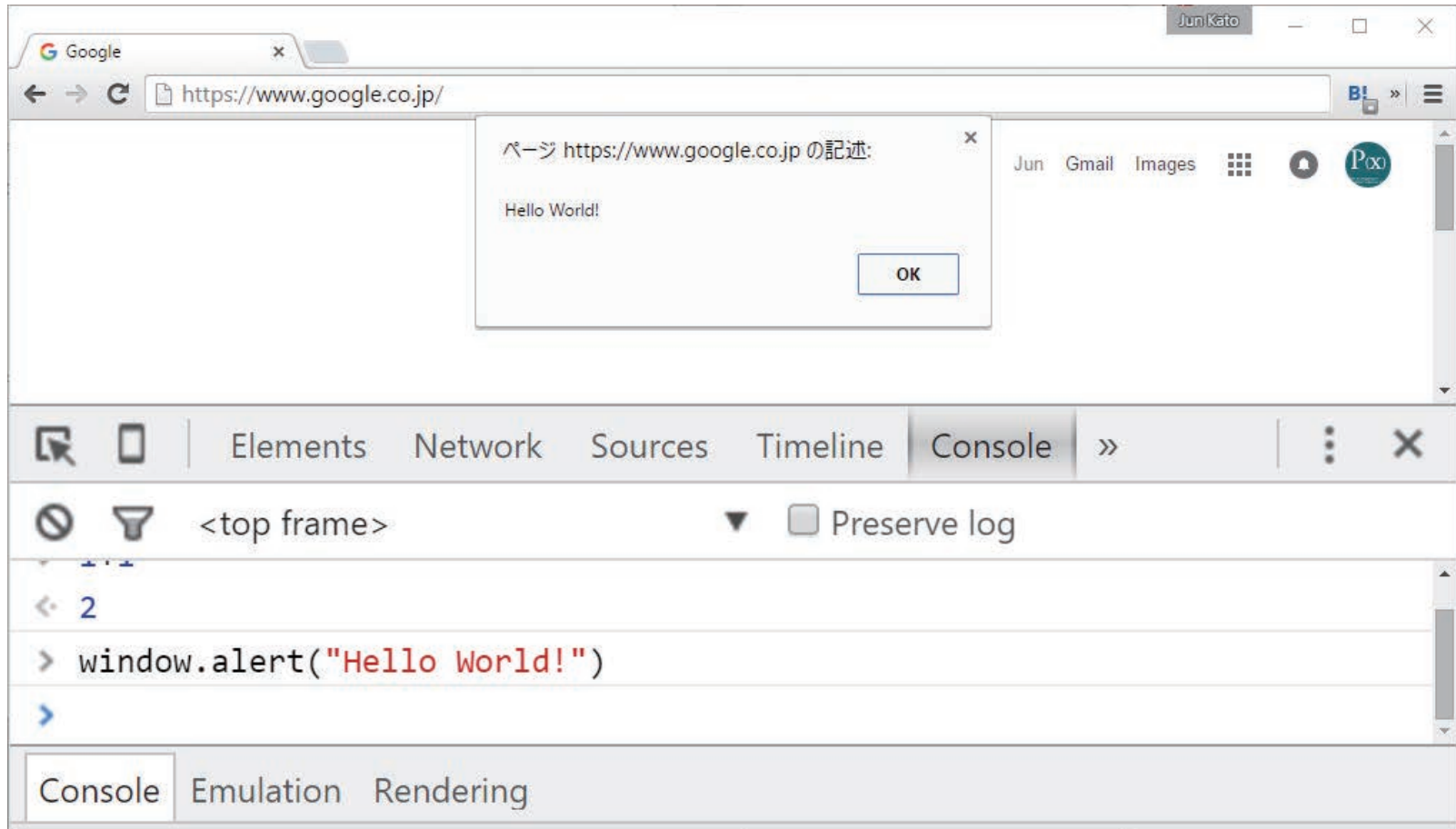


# 1+1

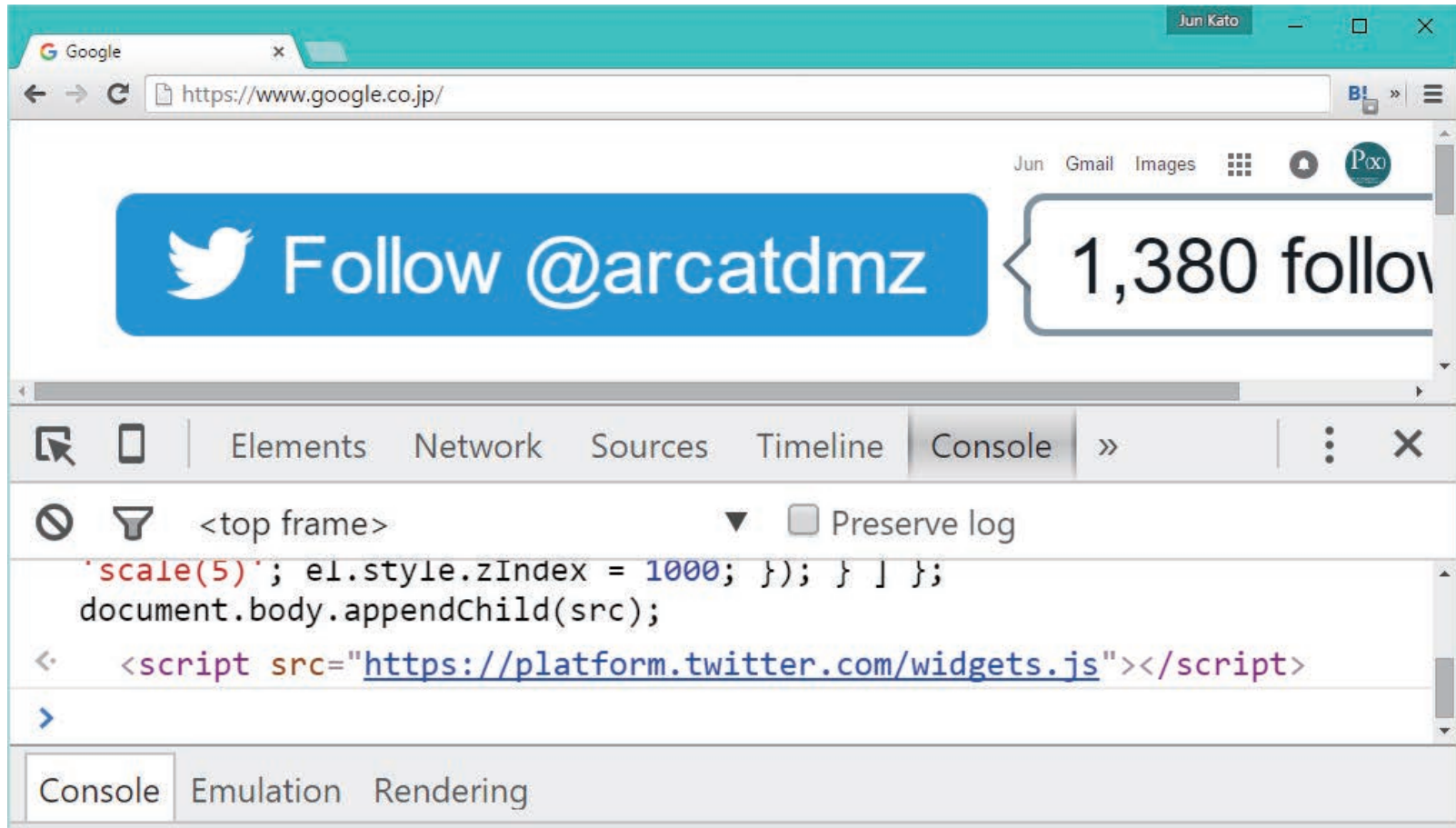




# window.alert("Hello World!")



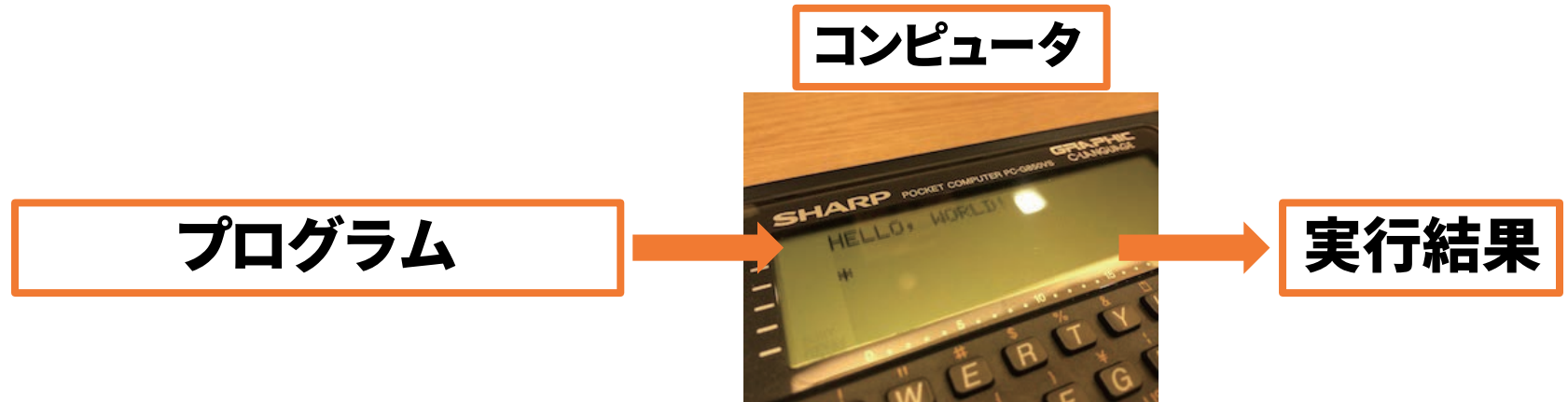
```
var src = document.createElement("script"); src.src =  
"https://platform.twitter.com/widgets.js"; window.twtr = { _e: [ function(tw)  
{ tw.widgets.createFollowButton("arcatdmz", document.body).then(function (el)  
{ el.style.position = "absolute"; el.style.top = "50%"; el.style.left = "50%"; el.style.transform =  
"scale(5)"; el.style.zIndex = 1000; }); } ] }; document.body.appendChild(src);
```



# 文字列を入力したら何か起きた



# {高級言語|アセンブラ言語|機械語} を入力したら何か起きた



タブレットはプログラムがなければ  
ただのまな板

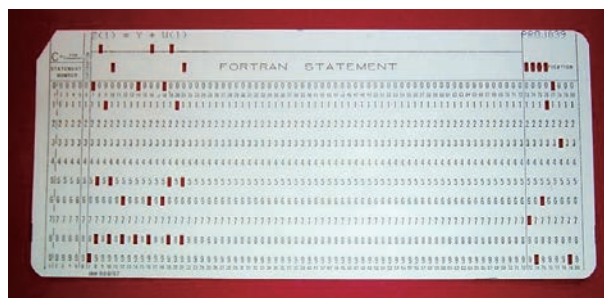
# 1960年代前半までの開発環境

- コンピュータは自動計算のためのもの
- **コーディング**はカードにパンチすることを指した



記録する内容は  
自力で計算

機械語をカードに記録



実行



プログラミングは全くインタラクティブじゃなかった😞

# 統合開発環境 1964

- コンピュータ上でインタラクティブに開発できる
- ワークフロー全体を支援してくれる

```
public class HelloWorld {  
    // Hello World!  
    public static void  
        main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

スクリーン上で  
高級言語を記述

機械語を電子的に保存

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	0123456789ABCDEF
000000	CA	FE	BA	BE	00	00	00	33	00	14	07	00	02	01	00	0A	.....3.....
000010	48	85	6C	6C	6F	57	6F	72	6C	64	07	00	04	01	00	10	HelloWorld.....
000020	6A	61	76	61	2F	6C	61	6E	67	2F	4F	62	6A	65	63	74	java/lang/Object
000030	01	00	06	3C	69	6E	69	74	3E	01	00	03	28	29	56	01	...<init>...()V.
000040	00	04	43	6F	64	65	0A	00	03	00	09	0C	00	05	00	06	..Code.....
000050	01	00	0F	4C	69	6E	65	4E	75	6D	62	65	72	54	61	62	...LineNumberTab
000060	6C	65	01	00	12	4C	6F	63	61	6C	56	61	72	69	61	62	le...LocalVariab
000070	6C	65	54	61	62	6C	65	01	00	04	74	68	69	73	01	00	leTable...this.
000080	0C	4C	48	65	6C	6C	6F	57	6F	72	6C	64	3B	01	00	04	..LHelloWorld;..
000090	6D	61	69	6E	01	00	16	28	5B	4C	6A	61	76	61	2F	6C	main...([Ljava

実行

```
Console [X]  
  
<terminated> HelloWorld  
Hello World!
```

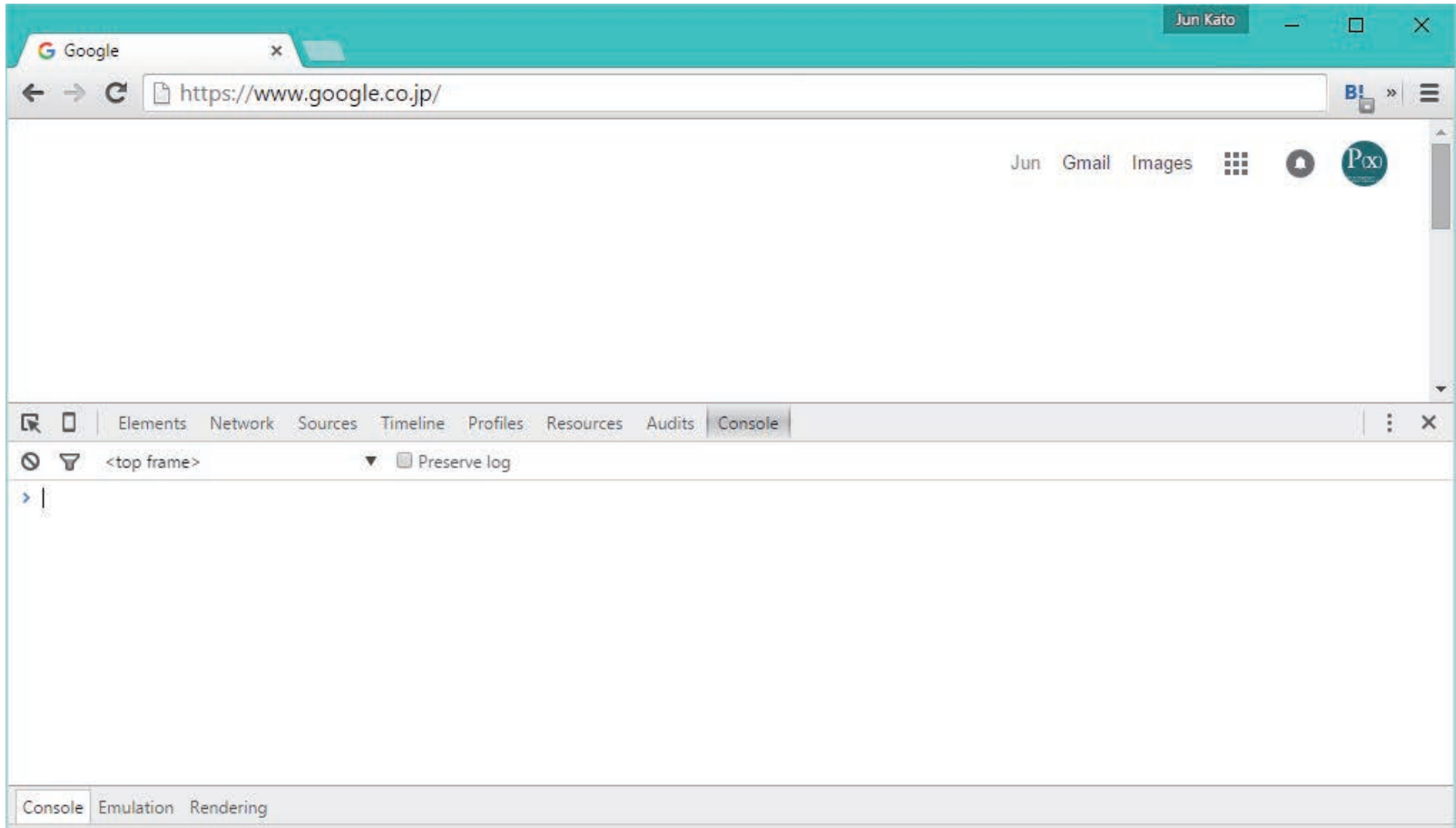
エディタ

コンパイラ

デバッガ

インタラクティブにプログラムを作れるようになった😊

# ブラウザは統合開発環境



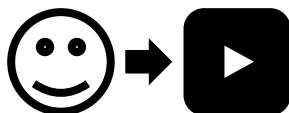


# プログラミングって何だっけ？

コンピュータ(汎用計算機)に**目的**を与える

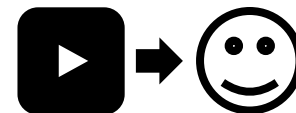
## Step.1

プログラムを設計して  
コンピュータに入力する



## Step.2

自分の意図を託して  
自動で行ってもらう



- 自分が死んでも動き続ける自分の化身を作る
- みんなに使ってもらいたい便利な道具を作る
- みんなに楽しんでもらいたいゲームを作る
- みんなに考えてもらいたい課題を提示する

何らかの体験を提供するメディアを作る行為

# プログラミングで コンテンツを作ってみよう

\* as a Code —コンテンツをプログラミングする方法—

# 静止画gは平面の関数 $f(x, y)$

for each (水平ピクセル $x$ )

for each (垂直ピクセル $y$ )

$\{r, g, b\} = f(x, y)$

色 $(r, g, b)$ で $(x, y)$ を塗りつぶし

線を引くには？円を描くには？

# 静止画gは平面の関数 $f(x,y)$

for each (描画コマンド)

for each (水平ピクセルx)

for each (垂直ピクセルy)

if (dot && x == 描画先x && y == 描画先y)

色(r, g, b)で(x, y)を塗りつぶし

else if (line && ...)

意外と、  
そんなものです

場合分けの嵐？

# 動画vは時間の関数f(t)

for each (全描画対象オブジェクト)

描画のためのパラメタを操作

for each (グラフィック)

座標系操作と描画

for each (フレーズ)

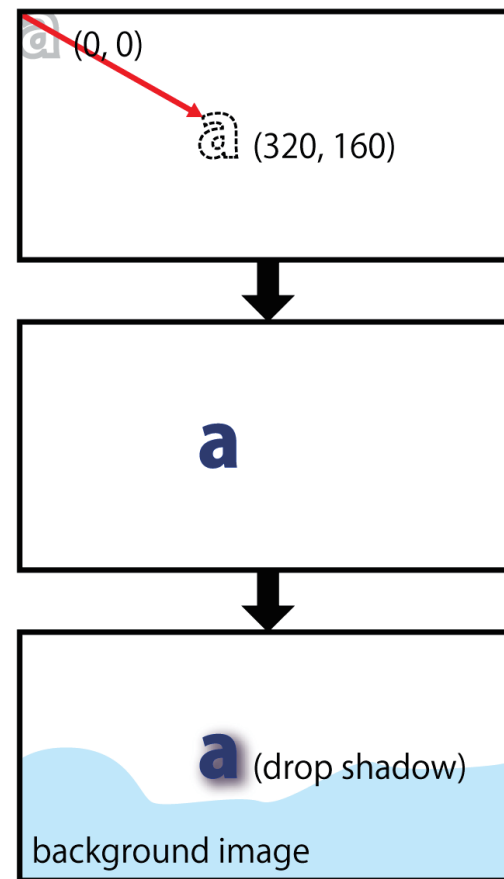
座標系操作

for each (単語)

座標系操作

for each (文字)

描画



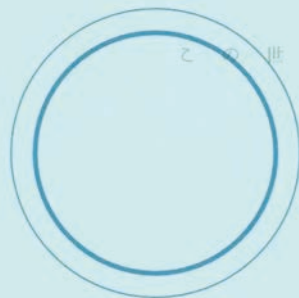
# 歌詞アニメーション



この世界の

この世界のX

この世界のX



□

X

この世界のXロディー

この世界のXロディー

more at <http://textalive.jp>

# 歌詞アニメーションの制作

カラオケ表示

わたしの歌声

- 表示タイミング
- フォントサイズ
- 色

歌詞アニメーション



- 表示タイミング
- フォントサイズ**の変化**
- 色**の変化、場所の変化、変形や視覚的なエフェクト**

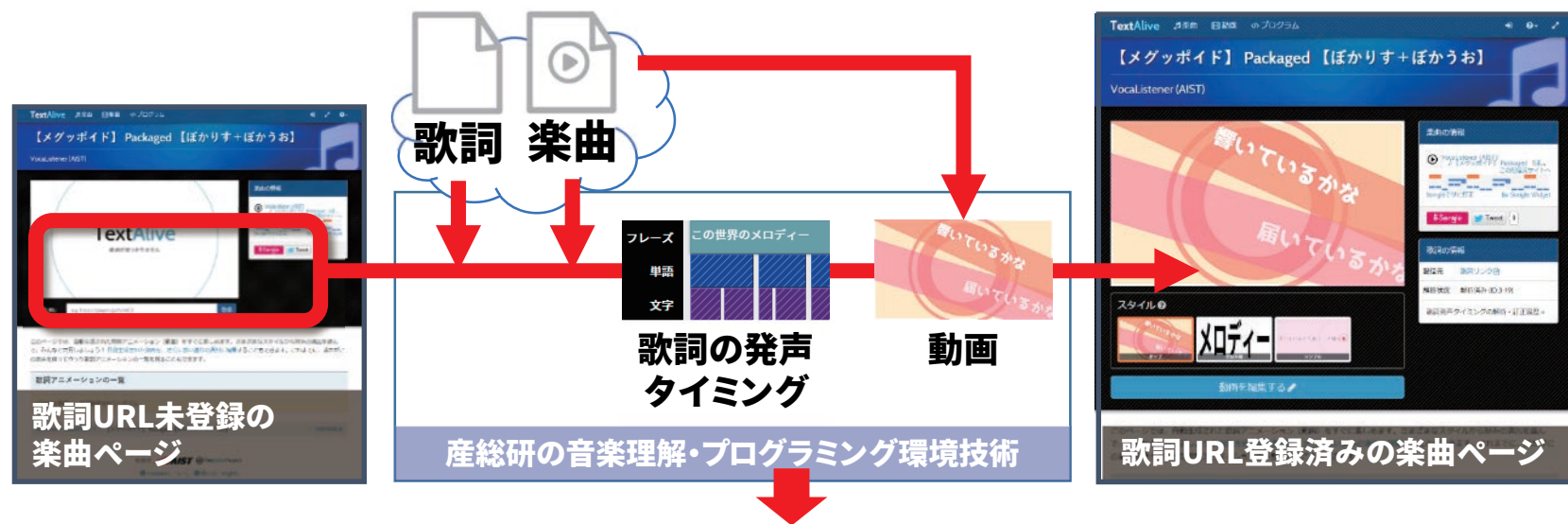


歌詞アニメーションは**作るのが難しく手間がかかる**



# 歌詞アニメーションの、 プログラミングによる自動生成

- 歌詞・楽曲から歌詞アニメーションを自動生成
- スタイルを選んで演出を瞬時に切り替えられる



**TextAliveは創作をすぐに楽しめる**

# Internet of Things

f3.js 作例集

# Internet of Things

- ソフトウェアだけでなくハードウェアのユーザインタフェース設計が必要
- これまでは2種類のツールを使っていた

統合開発環境



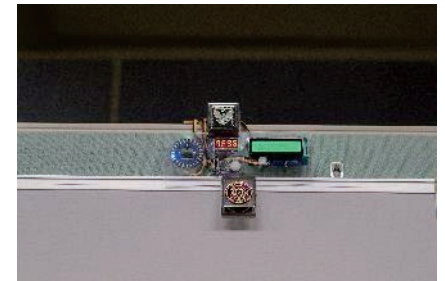
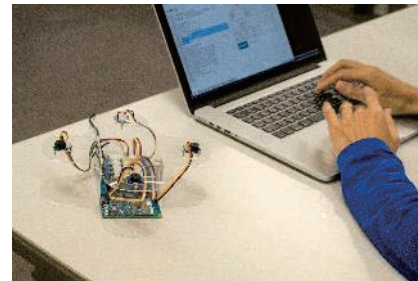
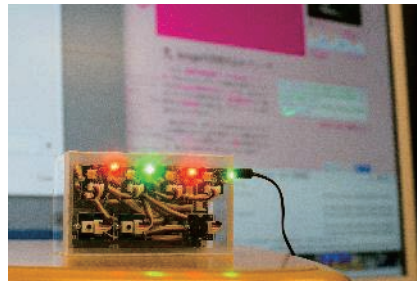
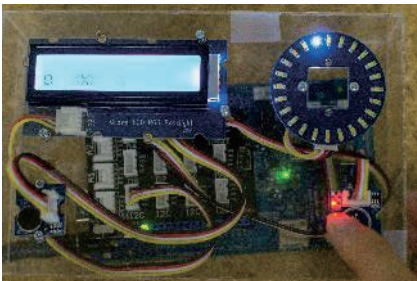
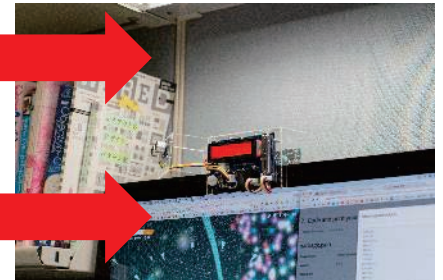
ファームウェア



CADソフトウェア



筐体デザイン



# IoTの、 プログラミングによる全体設計

プログラミングで筐体とファームウェアを開発

筐体デザイン

f3.js

ファームウェア

完成



**f3.jsは再現性と再利用性が高い**

# プログラミングで プログラミング環境を 作ってみよう

\* as a Code —コンテンツをプログラミングする方法—

# ブラウザは統合開発環境



# 統合開発環境

- コンピュータ上でインタラクティブに開発できる
- ワークフロー全体を支援してくれる

```
public class HelloWorld {  
    // Hello World!  
    public static void  
        main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

スクリーン上で  
高級言語を記述

機械語を電子的に保存

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	0123456789ABCDEF
000000	CA	FE	BA	BE	00	00	00	33	00	14	07	00	02	01	00	0A	.....3.....
000010	48	85	6C	6C	6F	57	6F	72	6C	64	07	00	04	01	00	10	HelloWorld.....
000020	6A	61	76	61	2F	6C	61	6E	67	2F	4F	62	6A	65	63	74	java/lang/Object
000030	01	00	06	3C	69	6E	69	74	3E	01	00	03	28	29	56	01	...<init>...()V.
000040	00	04	43	6F	64	65	0A	00	03	00	09	0C	00	05	00	06	..Code.....
000050	01	00	0F	4C	69	6E	65	4E	75	6D	62	65	72	54	61	62	...LineNumberTab
000060	6C	65	01	00	12	4C	6F	63	61	6C	56	61	72	69	61	62	le...LocalVariab
000070	6C	65	54	61	62	6C	65	01	00	04	74	68	69	73	01	00	leTable...this.
000080	0C	4C	48	65	6C	6C	6F	57	6F	72	6C	64	3B	01	00	04	..LHelloWorld;..
000090	6D	61	69	6E	01	00	16	28	5B	4C	6A	61	76	61	2F	6C	main...([Ljava

実行

```
Console [X]  
  
<terminated> HelloWorld  
Hello World!
```

エディタ

コンパイラ

デバッガ

インタラクティブにプログラムを作れる😊



# \* as a Code

コンピュータはプログラムがなければ  
ただの箱

コンピュータはどんなメディアでも作れる



プログラミングできればどんなメディアでも作れる

ただし、どこまでプログラミングすべきかは常に考えておかないといけない

# Appendix

\* as a Code —コンテンツをプログラミングする方法—

# 1980年代 パーソナルコンピュータ登場: コンピュータ科学者分断の時代

- プログラミングできない「エンドユーザ」の登場
- ユーザインタフェース・HCI研究の勃興
  - UIST [1982-], CHI [1982-], VL/HCC [1984-]
- プログラミングに関する研究が分断された

プログラミング言語・  
ソフトウェア工学  
に関する研究

PLDI, OOPSLA, ICSE

ユーザインタ  
フェース・HCI  
に関する研究

UIST, CHI

エンドユーザ  
プログラミング  
に関する研究

VL/HCC