



Windows アプリ 開発入門

～Universal Windows Platform,
cocos-2dxゲーム開発の基礎～

自己紹介



- 梶井 良太(とがい りょうた)
 - 静岡大学大学院情報学専攻在籍中 (M1)
 - Microsoft Student Partners
 - Microsoft MVP for Windows Development
 - 好きなもの
 - Windows Phone or Tablet etc..
 - 音声対話システム
 - 自称Cortana芸人
 - Twitter : [@garicchi](https://twitter.com/garicchi)



目次

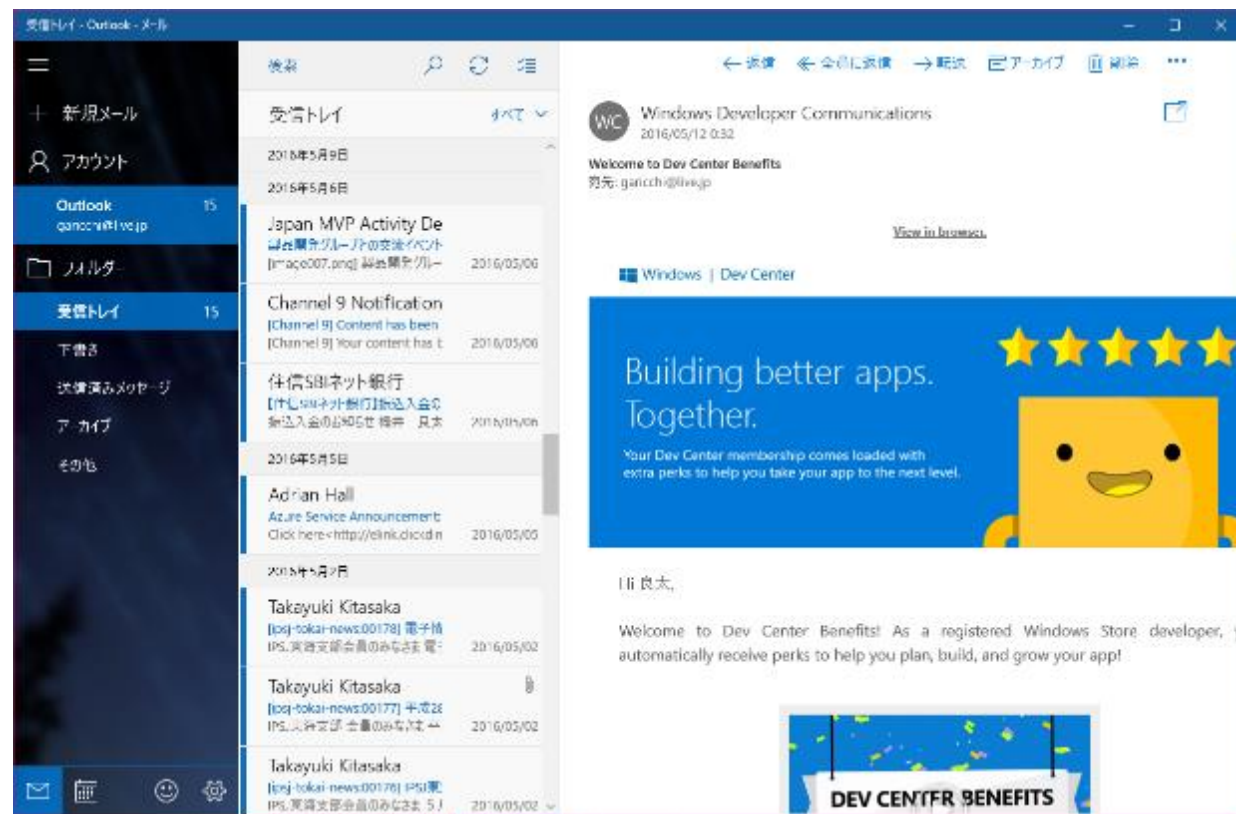
- 概要
 - Universal Windows Platformとは
 - 必要な開発環境
 - 競合製品と比べて
 - 学生にとってうれしいこと
 - 無料でできること、できないこと
- 開発入門
 - 開発環境構築方法
 - 新しくプロジェクトを作る
 - 実行方法
 - デモ
- さらに学ぶには
 - 公式サイトの見方
 - 見るべき入門サイト、情報源
 - ユーザーコミュニティ

概要

Universal Windows Platformとは

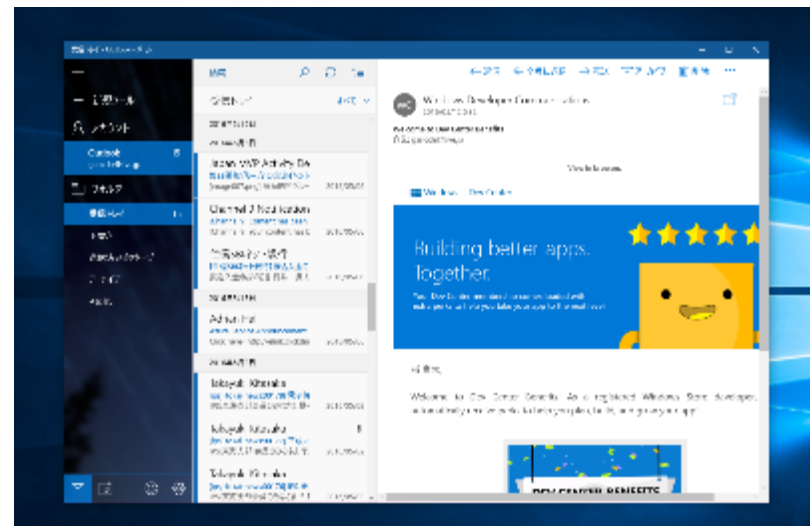
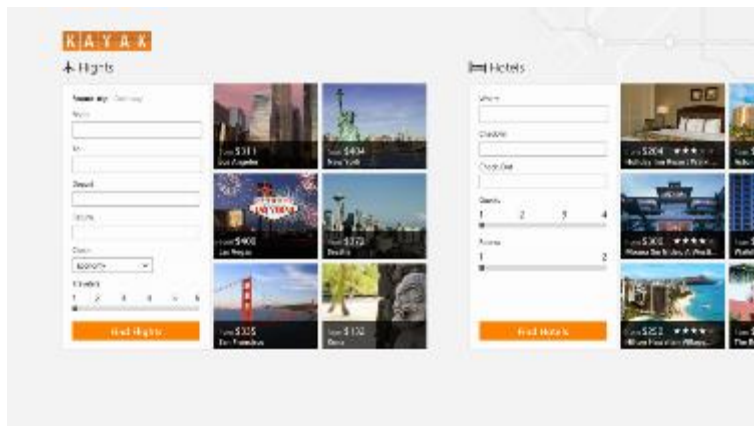
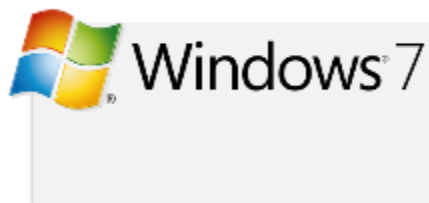
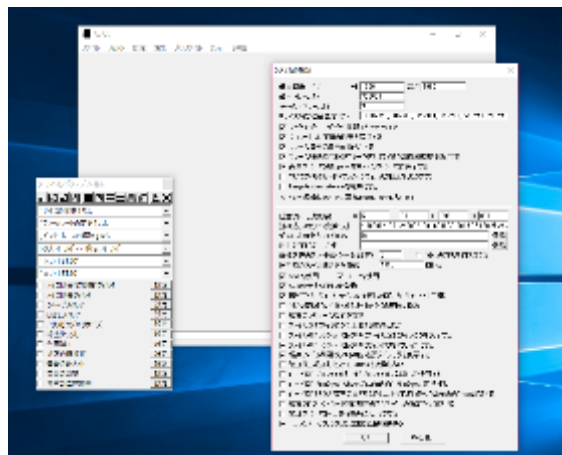
- Universal Windows Platform

- Windows10 で登場したWindowsアプリの新しいプラットフォーム
- プラットフォーム -> (アプリの見た目、プログラミング方法、動作方法、Cortana)



Universal Windows Platformとは

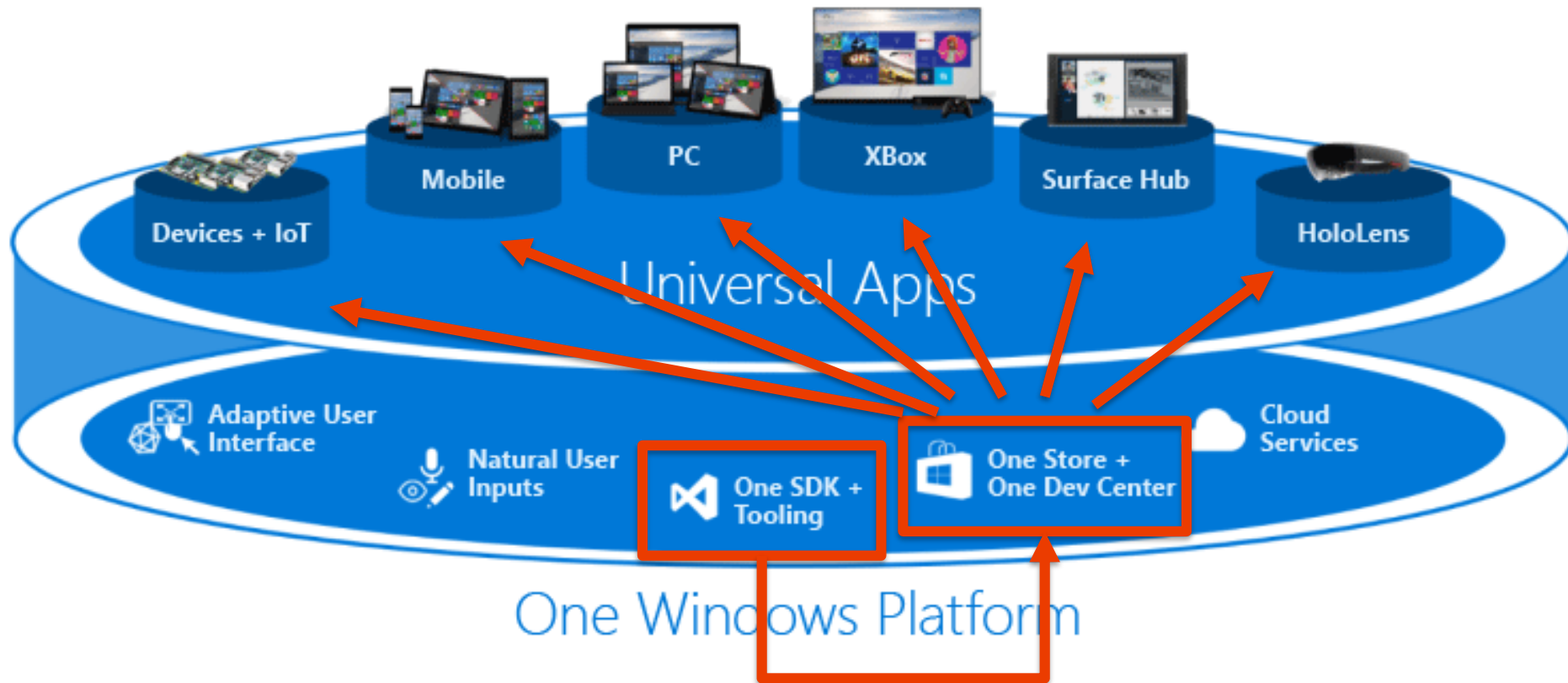
- Windowsの歴史とアプリのインターフェースの変化



Universal Windows Platform

Universal Windows Platformとは

- 多様なデバイスへの対応



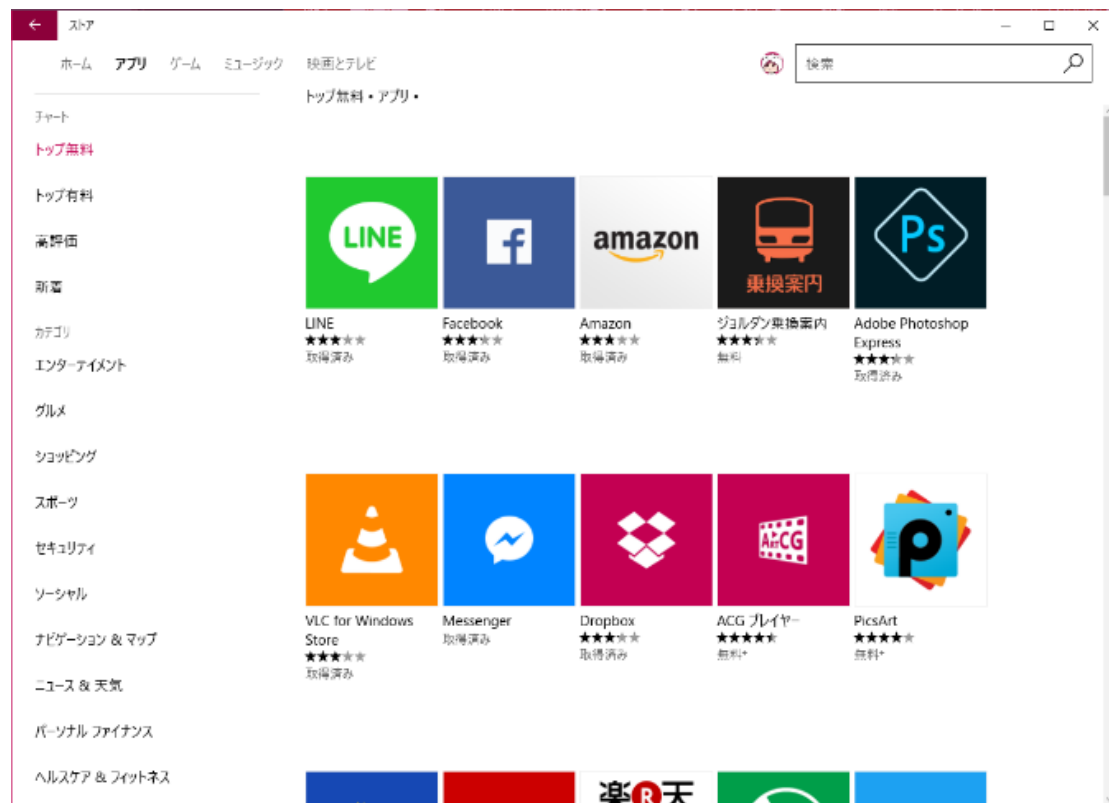
HoloLends



<https://www.youtube.com/watch?v=A60FhfG8U64> - by kaorun55さん

Universal Windows Platformとは

- ソフトウェアの配布方法の進化



- プログラミング言語の多言語化

C#

Visual Basic

Java Script

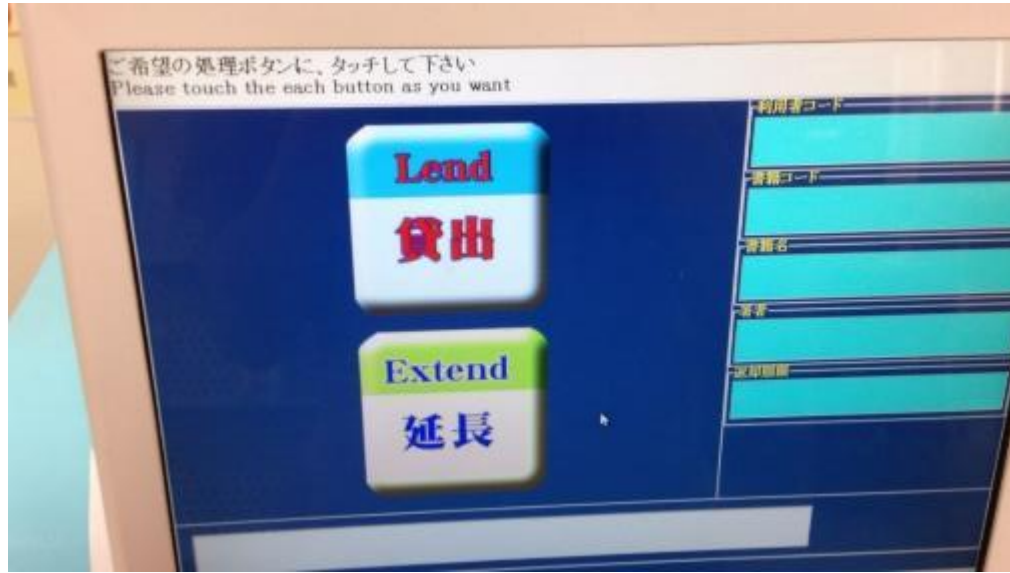
C++

Universal Windows Platformとは

- **たくさん**のプログラミング言語を選択でき
- **1つの**開発環境(VisualStudio)で開発し
- **1つの**ストアで配信を行い
- **たくさん**のデバイスで動作をするWindowsアプリケーション

Universal Windows Platform

Windowsのタッチ化でもたらされるもの



必要な開発環境

- その製品を使うために必要な開発環境
 - Windows 10 (エディションは問わない)
 - Visual Studio 2015 (エディションは問わない)
 - Windows10 SDK (Software Development Kit) ※
- あるとよいもの
 - Windows10タブレット(実機)
 - Windows Phone (実機)
- ストアにアプリを公開するために必要なもの
 - 約2000円の開発者アカウント (更新不要)
 - <https://developer.microsoft.com/ja-jp/windows/programs/faq>

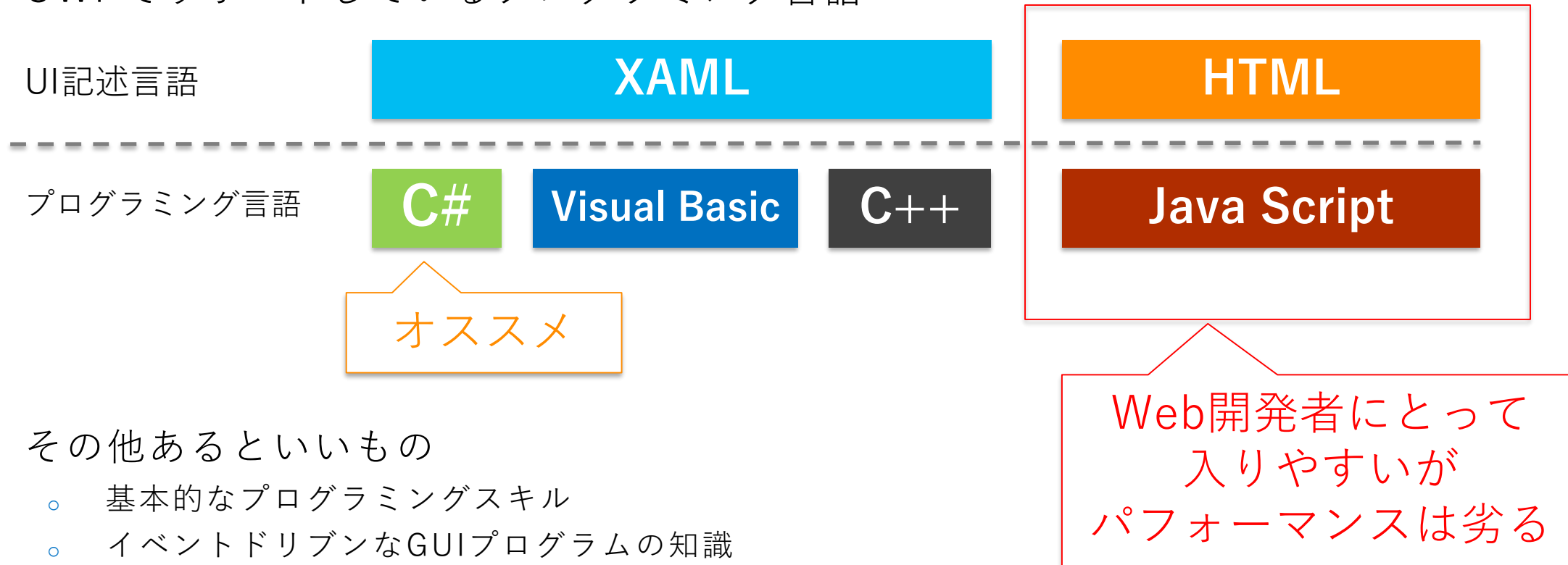


DreamSparkで無料！！

※ Visual Studio 2015 インストール時に同時にインストール可能
(詳細は後述)

必要なスキル

- UWPでサポートしているプログラミング言語



- その他あるといいもの
 - 基本的なプログラミングスキル
 - イベントドリブンなGUIプログラムの知識
 - XAMLなプラットフォームの知識(例 WPF)

競合製品と比べて

- 競合製品



iOS



Android

- UWPがダメなところ

市場規模

開発者数

- UWPがいいかもしれないところ

XAMLなアプリの開発経験

他言語で作ったライブラリが生かせる

動作デバイスの多様さ

無料でできること、できないこと

- 学生はどこまでが無料？

すべて無料で開発可能

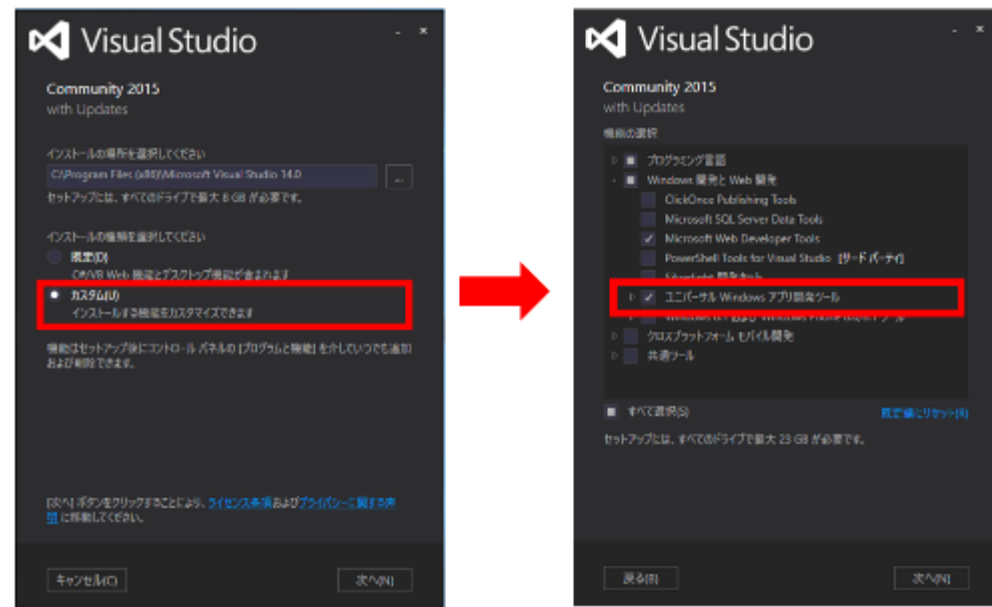
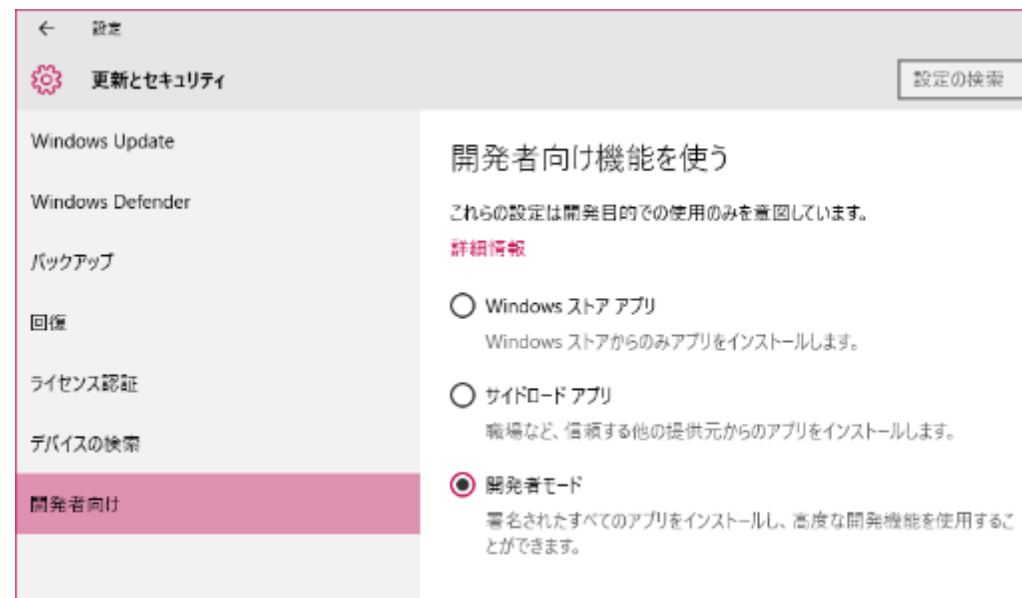
- Visual Studio Professional Editionは有料
- Community Editionは無料
- UWPを開発する分にならCommunity Editionで十分

※PC,OS費、電気代などは除く

開発入門

開発環境構築方法

- Windows10を開発者モードに
 - 設定アプリ>更新とセキュリティ
>開発者向け>開発者モード
- Visual Studioのインストール
 - <http://visualstudio.com>から
Community Edition(無償版)を
 - インストール時に[カスタム]
->[ユニバーサルWindowsアプリ開発ツール]に
チェックを入れる
 - インストールには時間がかかるので注意



UWPアプリの作り方

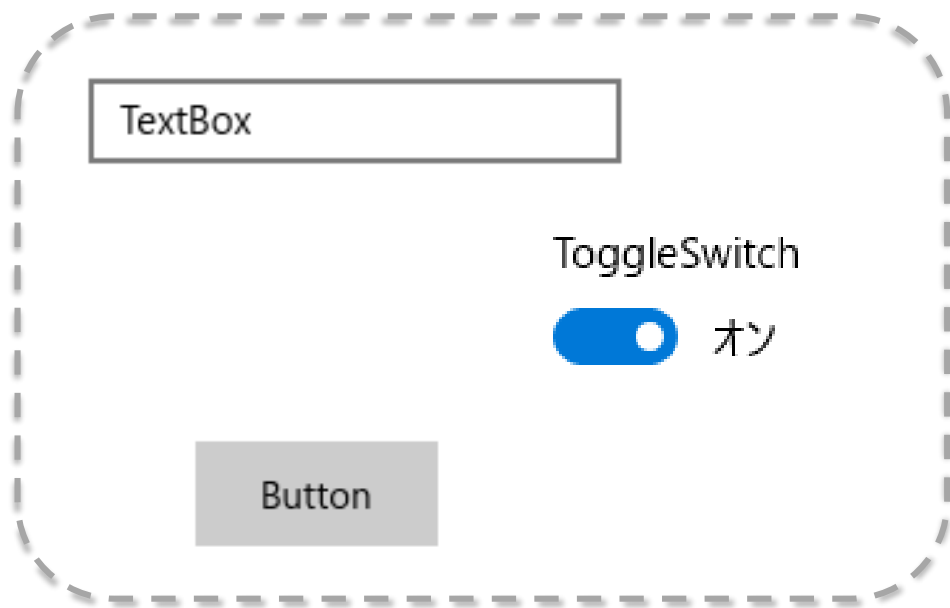
基本はこの3ステップ

1. UIコントロール配置
2. イベント登録
3. プロパティ編集

C#の場合

UWPアプリの作り方

- UIコントロール配置



UIコントロール

画面

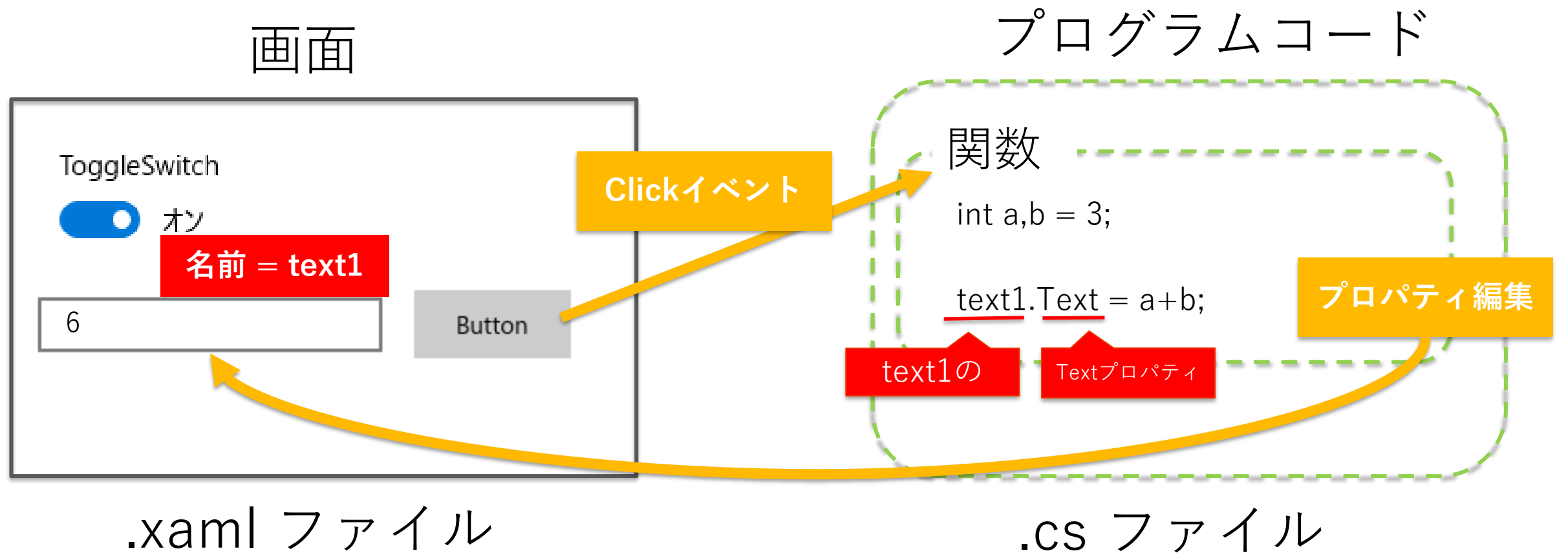


.xaml ファイル

C#の場合

UWPアプリの作り方

- イベント登録 & プロパティ編集



C#の場合

アプリの基本はイベントとプロパティ

- これを繰り返すだけでアプリが作れる



プログラムコード

関数

```
int a,b = 3;
```

```
text1.Text = a+b;
```

プロパティ編集

.CS ファイル

C#の場合

デモ

UWPアプリの作り方

- 詳細手順

1. UIコントロール配置
2. プログラムから操作したいUIコントロールに名前を付ける
3. ユーザーからの挙動を受け取りたいUIコントロールのイベントに関数を登録
4. 関数からUIコントロールのプロパティを変える

C#の場合

UWPアプリの作り方

- Webページを作る感覚でHTML、CSS、JavaScriptを使ってUIを記述
- JavaScriptからWindowsのAPIを呼び出す

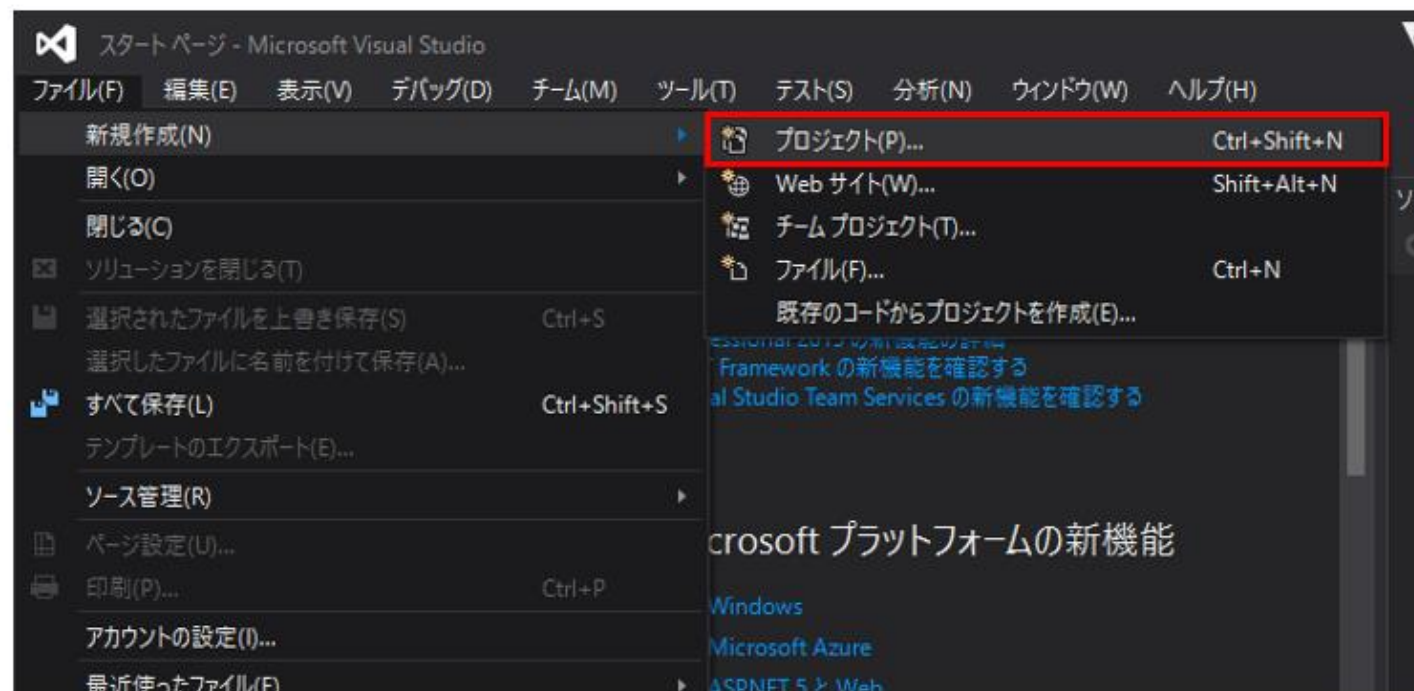
JavaScriptの場合

UWPの作り方

C#編

新しくプロジェクトを作る - C#

- Visual Studioではアプリなどのソースコードをまとめた1つの単位を[プロジェクト]と呼ぶ
- 新規プロジェクト作成方法
 - メニューバー > [ファイル] > [新規作成(N)] > [プロジェクト]

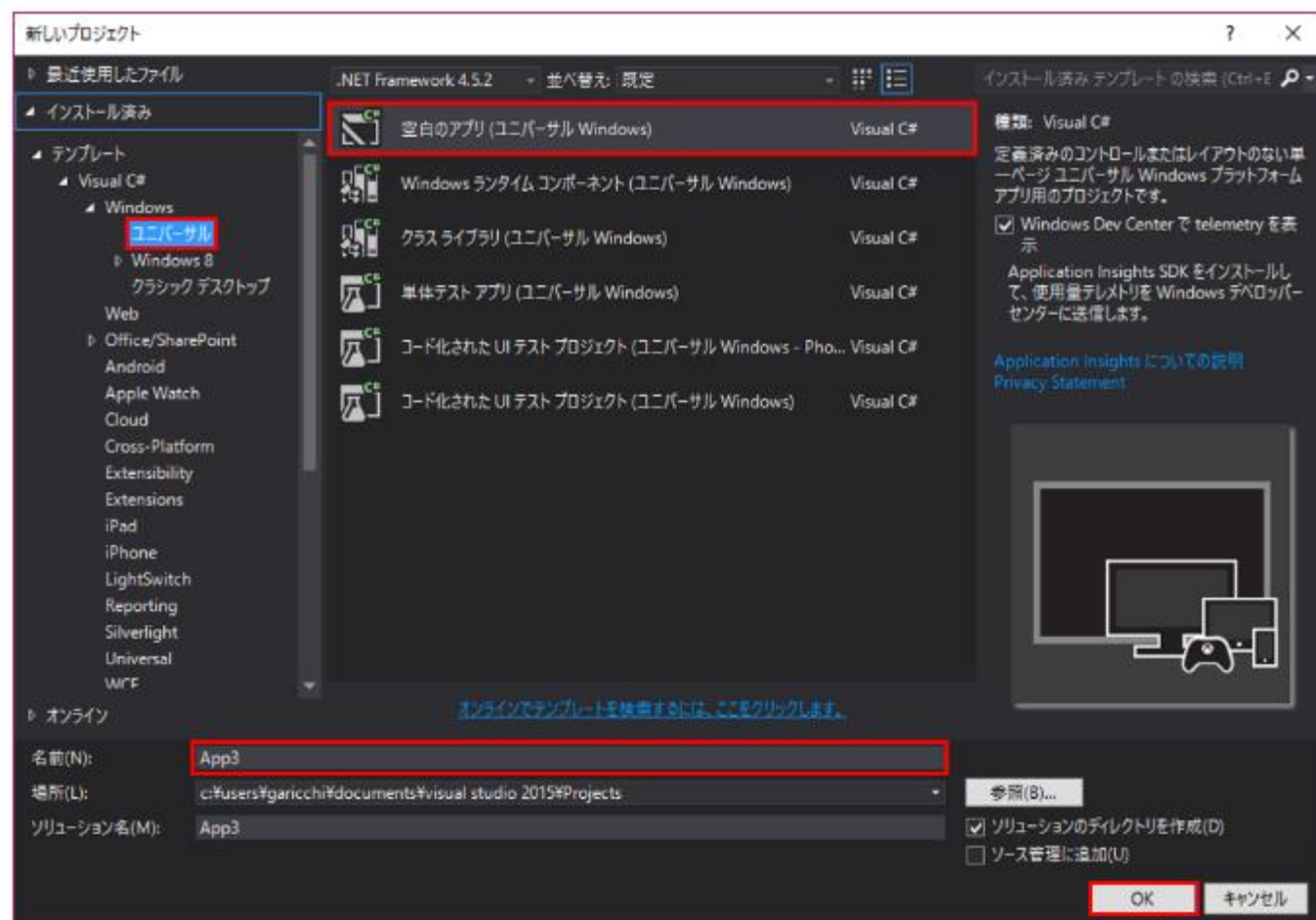


新しくプロジェクトを作る - C#

- テンプレートの選択
 - 左のリストから[インストール済み]>[テンプレート]>[VisualC#]>[Windows]>[ユニバーサル]
 - [空白のアプリ](ユニバーサルWindows)を選択

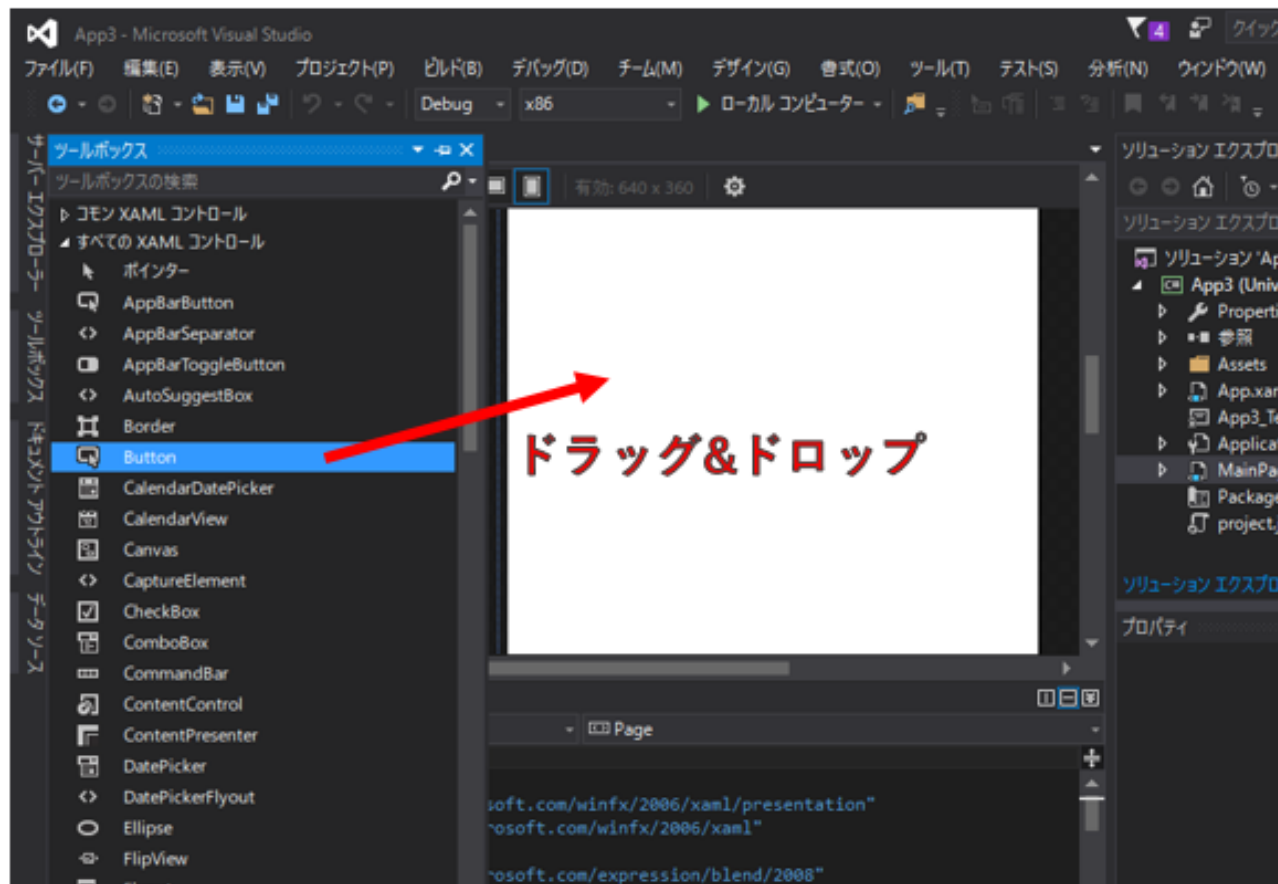
- 名前の決定
 - [名前(N):]の欄にアプリの名前を入れる

- [OK]ボタンを押す



アプリの基本形を作る - C#

- XAMLデザイナーで画面を作る
 - MainPage.xamlファイルを開く
 - 左の[ツールボックス]から、[Button]をXAML画面にドラッグアンドドロップ
 - 画面の好きな位置に好きなコンポーネントを配置していく



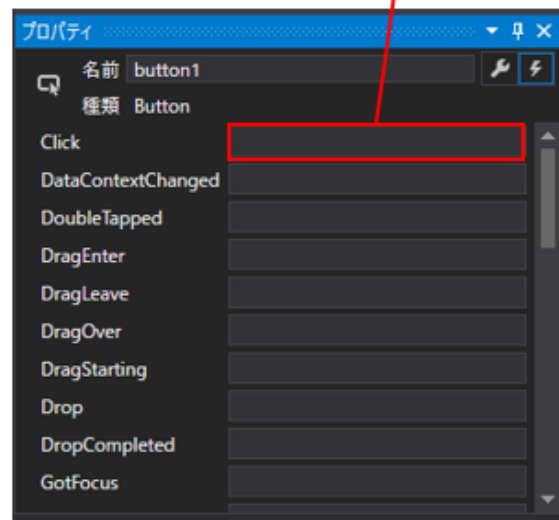
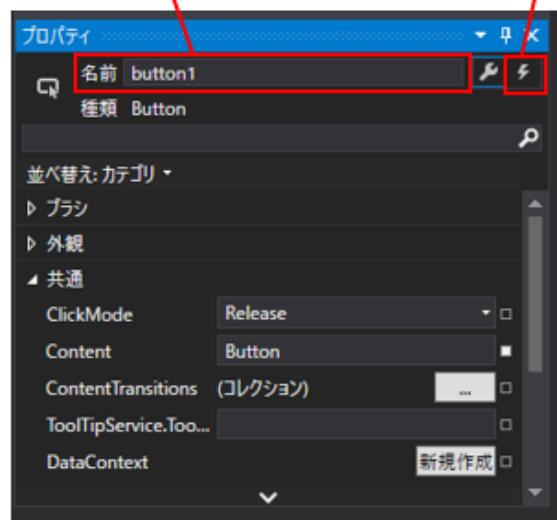
アプリの基本形を作る - C#

- UIコンポーネントのプロパティを編集する
 - UIコンポーネントの各種設定となる
 - 名前、Content、Background etc...
- UIコンポーネントのイベントを登録する
 - UIコンポーネントに起こるさまざまなイベントを受け取る
 - Buttonの場合 -> Clickイベントをダブルクリック

①名前を[button1]に変更

②クリック

③ダブルクリック



アプリの基本形を作る - C#

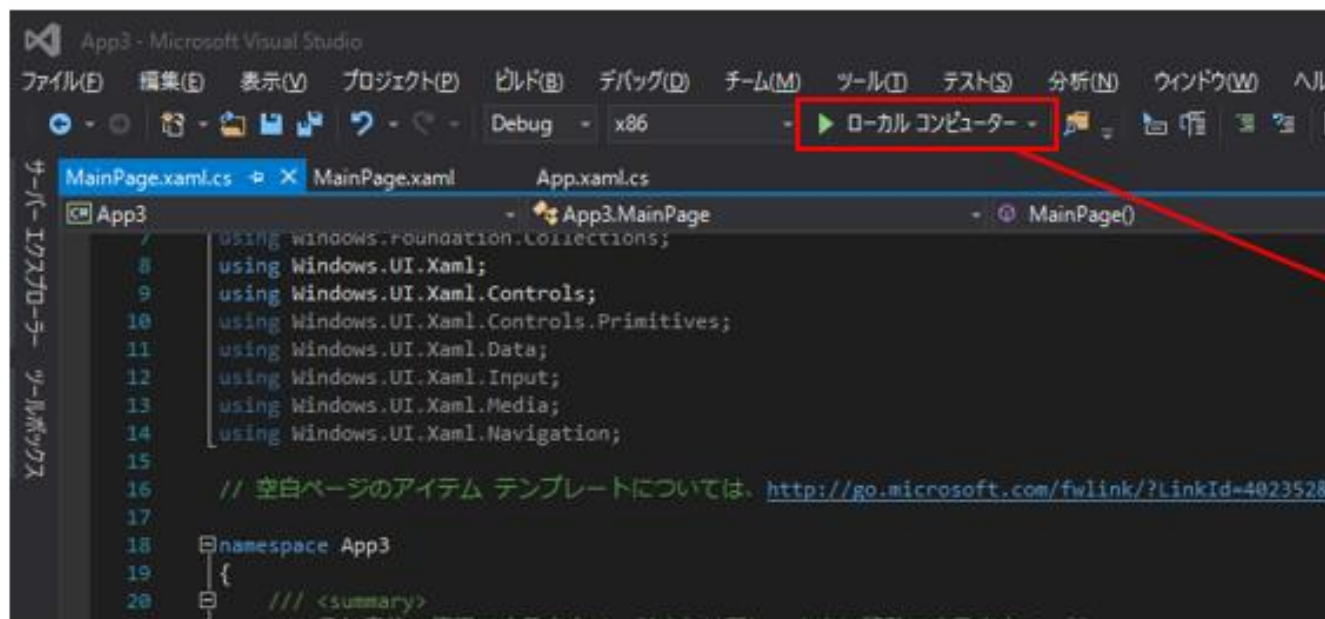
- イベントを受け取った後の処理を書く
 - イベントをダブルクリックすると MainPage.xaml.cs に移る
 - イベントが発生したときに実行されるメソッドが自動生成されるので実行したい処理を書く
 - 下図の場合、ボタンに表示されている文字を「Hello,World!」にする

```
public sealed partial class MainPage : Page
{
    public MainPage()
    {
        this.InitializeComponent();
    }

    private void button1_Click(object sender, RoutedEventArgs e)
    {
        button1.Content = "Hello,World!";
    }
}
```

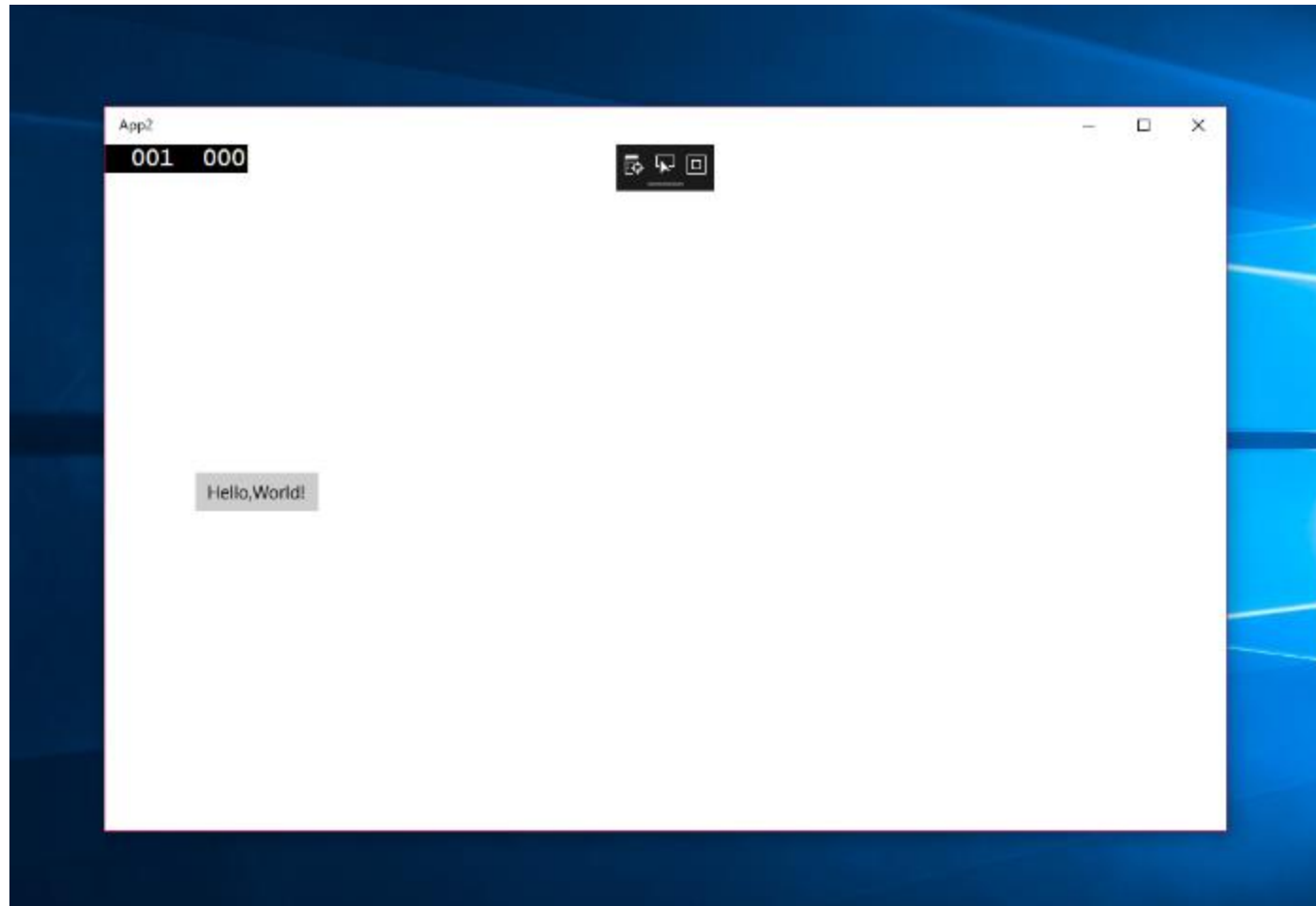
実行方法 - C#

- アプリの実行方法
 - ツールバーにある[ローカルコンピューター]をクリック



実行方法 - C#

- Hello,World!



さらに学ぶには

ゲームプログラミングについて

- Universal Windows Platformでゲームを作る方法



C++

C++

or

JS

C#

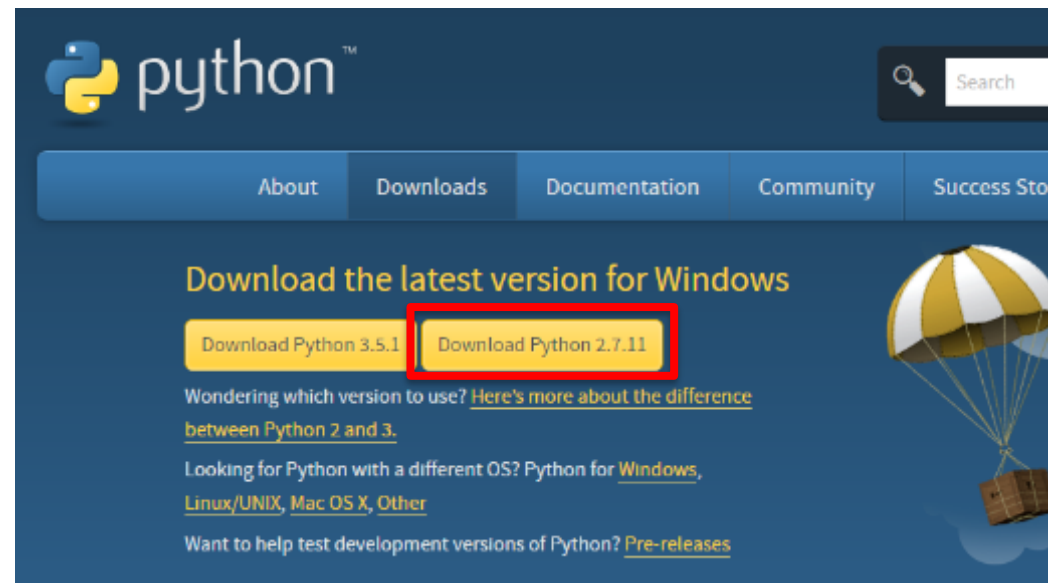
難易度

Cocos-2dx UWP開発

- Cocos 2dx UWP開発に必要なもの
 - cocos 2dx本体
 - Python 2.7.x
 - Windows10 SDK
 - Visual Studio 2015
- 開発に必要な知識
 - C++プログラミングの知識
 - 基本的なシェルの扱い(cdとかができればOK)

Python 2.7.xのインストール

- Python 2.7.xのインストール
 - <https://www.python.org/downloads/>
 - ※3.5系統じゃないことに注意
 - ダウンロードしたインストーラを起動
 - インストール
- インストールの確認
 - PowerShellで[python --version]を実行
 - 2.7.xな表示が出ればOK

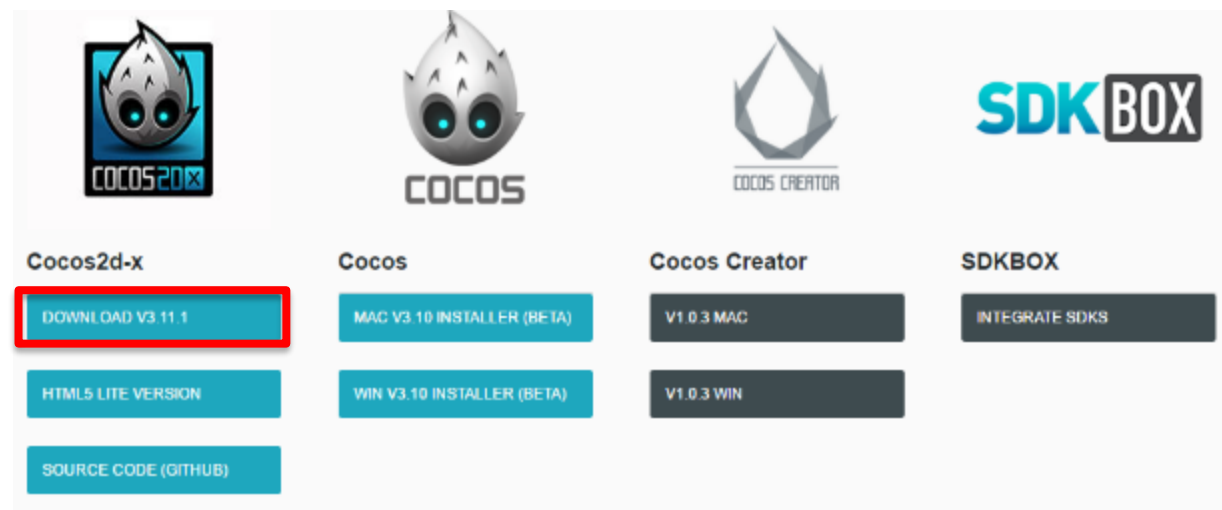


```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\garicchi> python --version
Python 2.7.5
PS C:\Users\garicchi>
```


Cocos-2dxのインストール

- Cocos 2dx本体のダウンロード
 - <http://www.cocos2d-x.org/download>



- インストール
 - ダウンロードしたcocos2d-x-3.11.1.zipを展開
 - 展開したフォルダ内のsetup.pyをpythonで実行
 - `python setup.py`

```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\garicchi> cd .\Downloads\cocos2d-x-3.11.1\cocos2d-x-3.11.1\
PS C:\Users\garicchi\Downloads\cocos2d-x-3.11.1\cocos2d-x-3.11.1> python setup.py
```

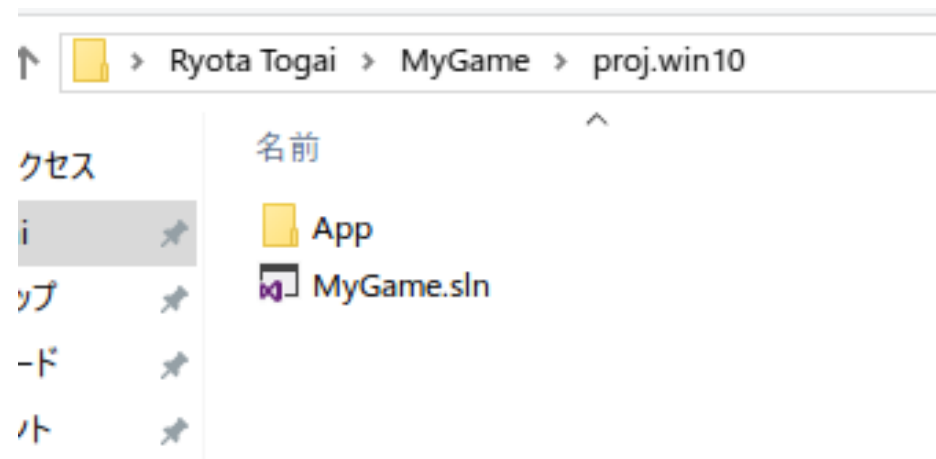
新しくプロジェクトを作る

- cocos new コマンドを実行
 - MyGame: ゲームのタイトル
 - com.your_company.mygame: ゲームのID

```
Windows PowerShell

PS C:\Users\garicchi> cocos new MyGame -p com.your_company.mygame -l cpp
```

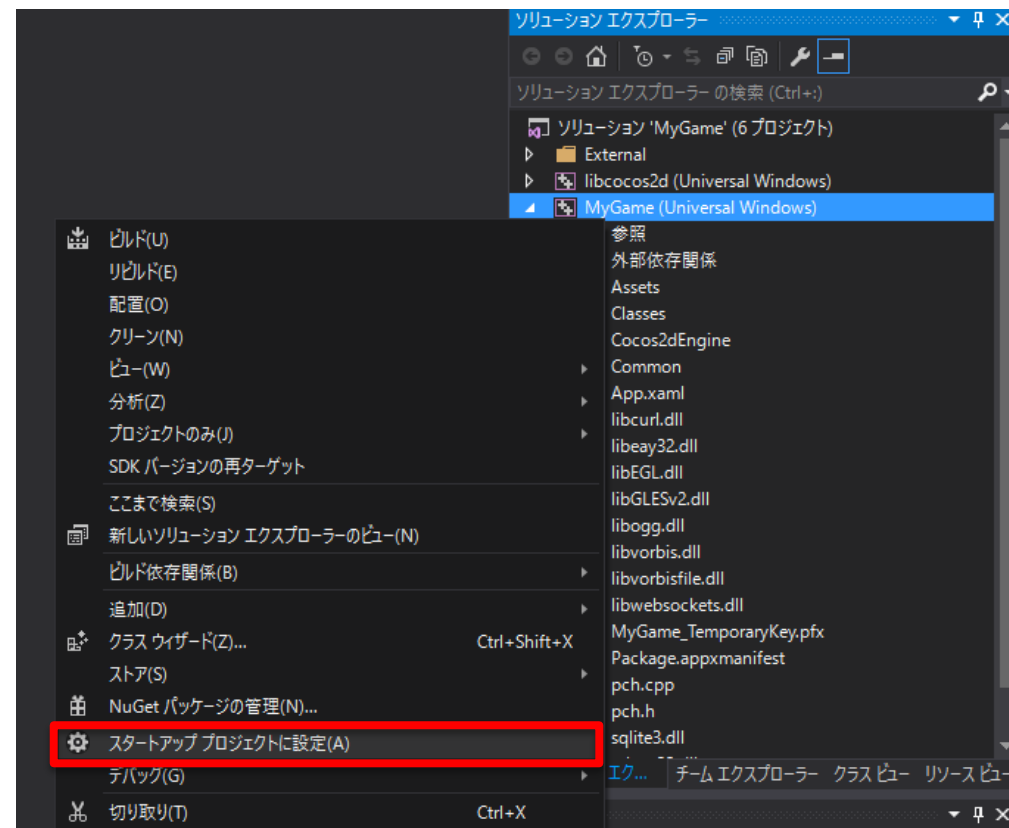
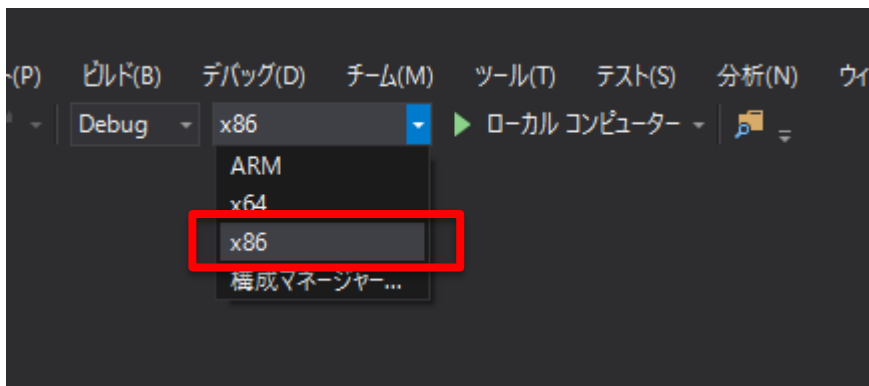
- ソリューションを開く
 - [MyGame\proj.win10] に移動
 - [MyGame.sln]をVisual Studioで開く



プロジェクトを実行する

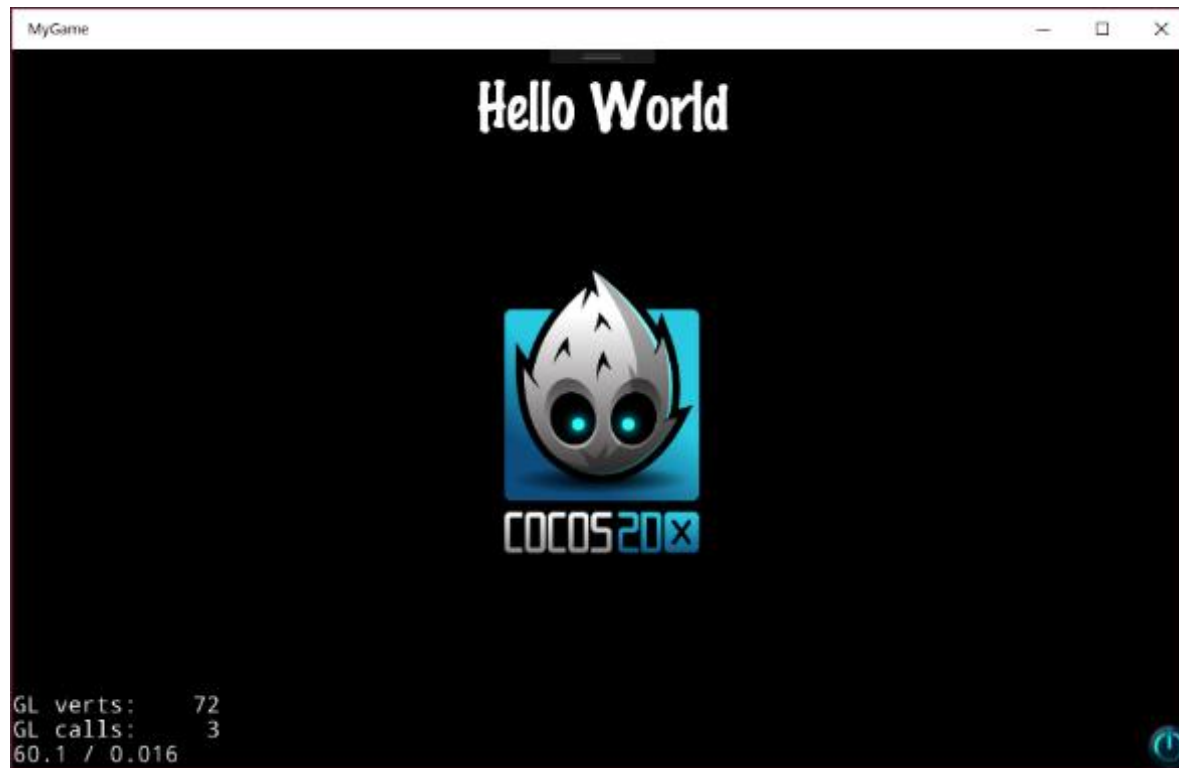
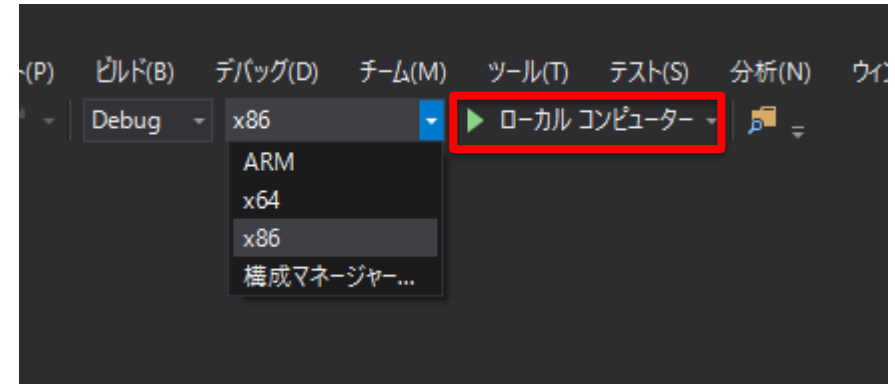
- スタートアッププロジェクトの変更
 - ソリューションエクスプローラーから[MyGame]->右クリック->[スタートアッププロジェクトに設定]

- ビルドターゲットの変更
 - ARMからx86に

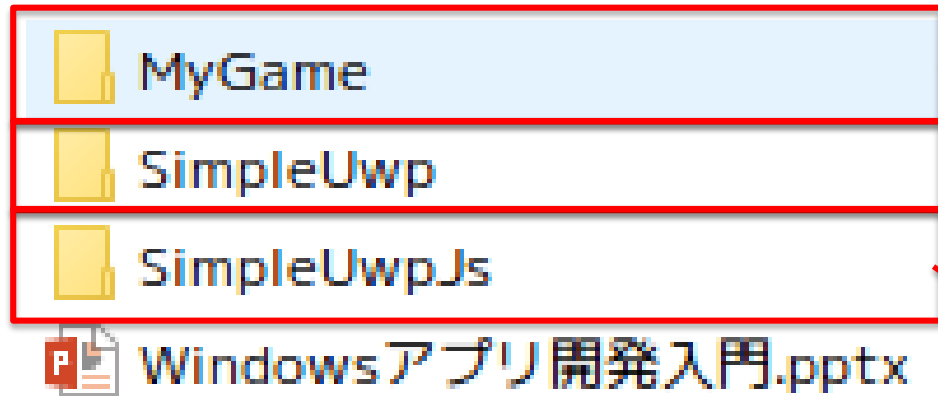


プロジェクトを実行する

- [ローカルコンピュータ]を押す



サンプルコードについて



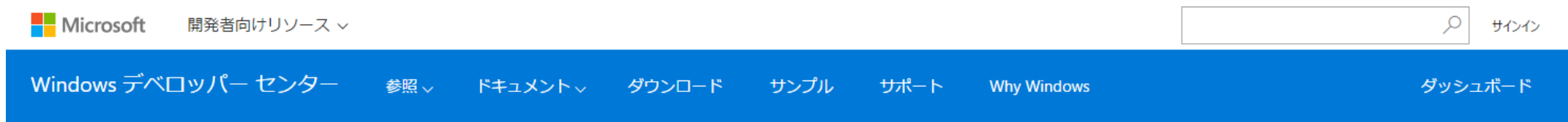
cocos-2dxのサンプル
(変更点のあるコードのみ)

C#によるUWPのサンプル
(そのまま実行可能)

JavaScriptによる
UWPのサンプル
(そのまま実行可能)

公式サイトの見方 - 入門編

- Windows デベロッパーセンター - 作業の開始
 - <https://developer.microsoft.com/ja-jp/windows/getstarted>



Windows アプリ > 作業の開始

Windows アプリの概要

今すぐ Windows アプリの構築を始めましょう。Windows 10 を使うことで、ユニバーサル Windows プラットフォーム用アプリの開発がこれまでよりも簡単になります。1 つの API セットと 1 つのアプリ パッケージを使うだけで、すべての Windows 10 デバイス (PC、タブレット、電話など) で利用可能なアプリを構築できます。

ここから開始

ユニバーサル Windows プラットフォームは、すべての Windows デバイス ファミリーで利用可能です。PC、タブレット、電話など、すべてのデバイスで実行および適切に表示されるアプリを、単一のストアと単一のコードベースを使って構築することができます。

[Windows アプリの詳細](#)

準備

Visual Studio Community 2015 には、アプリの開発を開始するために必要なものが揃っています。しかも無料です。

[Visual Studio Community 2015 を入手する](#)

[設定の手順](#)

サインアップ

アプリ開発者として登録します。開発者アカウントでは、Windows ストアにアプリを公開できます。

[今すぐサインアップ](#)

[開発者アカウントの詳細](#)

アプリの作成

試してみましょう。初めての Windows アプリをお好きな言語で作ってください。チュートリアルで説明する内容は次のとおりです。

[C# と XAML のチュートリアル](#)

[JavaScript と HTML のチュートリアル](#)

[C++ と XAML のチュートリアル](#)

入門記事

公式サイトの見方 - 応用編

- Windows デベロッパーセンター - Windows 10アプリに関するハウツーガイド
<https://msdn.microsoft.com/ja-jp/library/windows/apps/mt244352.aspx>

カテゴリ別に分かれている
自分のしたいことをここから検索
アプリに実装

Windows アプリ > 開発

- Windows 10 アプリに関するハウツーガイド
 - アクセシビリティ
 - アプリ データと設定
 - アプリ間通信
 - オーディオ、ビデオ、およびカメラ
 - 連絡先とカレンダー
 - コントロール、レイアウト、テキスト
 - カスタム ユーザー操作
 - データ バインディング
 - デバッグ、テスト、パフォーマンス
 - デバイス、センサーの使用
 - ファイル、フォルダー、およびライブラリ
 - ゲームと DirectX
 - グローバリゼーションとローカライズ
 - グラフィックスとアニメーション
 - 起動、再開、バックグラウンド タスク

Windows 10 アプリに関するハウツー ガイド

[Windows 10 の UWP アプリ向けに更新。Windows 8.x の記事については、「[archive](#)」をご覧ください]

電話、タブレット、PC など、Windows 10 ベースのデバイスで実行できるユニバーサル Windows アプリを作成するのに必要となる情報について記事の種類別にまとめられた手順とコード例を掲載します。

ユニバーサル Windows プラットフォーム (UWP) の概要、またこれによって、同じコードを使って複数の Windows デバイスの種類に合わせたエク次の記事をご覧ください。

- 初めてのユニバーサル Windows アプリの作成
- ユニバーサル Windows プラットフォーム (UWP) アプリのガイド
- ユニバーサル Windows アプリとは?

このセクションの内容

トピック	説明
アクセシビリティ	障害を持つ人も含め、できるだけ多くの人ができるだけ広範囲な種類のデバイスにアクセスできるようにユニバーサル Windows

公式サイトの見方 - 新しいほう

- 最近日本語資料がアップデートされました
- 新しいほうが見やすい
- <https://developer.microsoft.com/ja-jp/windows/develop>

目次

- ▶ ユニバーサル Windows プラットフォームの概要
- ▼ 設計および UI
 - ▶ レイアウト
 - ▶ スタイル
 - ▼ コントロールとパターン
 - はじめに
 - 機能別コントロールのインデックス
 - アクティブなキャンバス
 - アプリ バーとコマンド バー
 - 自動提案ボックス
 - ボタン
 - ▶ 日付と時刻
 - チェックボックス
 - フリップ ビュー
 - ハブ
 - ハイパーリンク

テキスト ブロック

 Jim Walker | 最終更新日: 2016/06/06 | 1 投稿者 

テキスト ブロックは、アプリで読み取り専用テキストを表示するためのプライマリ コントロールです。これを使用すると、単一行または複数行のテキスト、インライン ハイパーリンク、書式 (太字、斜体、下線付きなど) が設定されたテキストを表示できます。

重要な API

- [TextBlock クラス](#)
- [Text プロパティ](#)
- [Inlines プロパティ](#)

適切なコントロールの選択

テキスト ブロックは、一般的に、リッチ テキスト ブロックより使い方が簡単で、テキスト レンダリングのパフォーマンスが優れているため、ほとんどのアプリで UI テキストに適しています。 [Text](#) プロパティの値を取得することによって、アプリ内でテキスト ブロックのテキストに容易にアクセスして使用することができます。テキストのレンダリング方法をカスタマイズするための書式設定オプションも、同じものが数多く用意されています。

テキスト内に改行を配置することはできますが、テキスト ブロックは単一の段落を表示するために設計されており、テキストのインデントはサポートされていません。複数の段落、段組テキスト、インライン UI 要素 (画像など) をサポートする必要がある場合は、[RichTextBlock](#) を使います。

公式サイトの見方 – GitHub編

- GitHub – Windows-universal-samples
 - <https://github.com/Microsoft/Windows-universal-samples>
 - かなりサンプルが充実している
 - テキトーに実行してどういう機能が調べるのによい

The screenshot shows the GitHub repository page for `Microsoft / Windows-universal-samples`. The repository is described as "API samples for the Universal Windows Platform." It has 33 commits, 3 branches, 35 releases, and 3 contributors. The page includes navigation tabs for Code, Issues (61), Pull requests (10), Wiki, Pulse, and Graphs. Below the repository information, there are buttons for "Branch: master", "New pull request", "Create new file", "Upload files", "Find file", and "Clone or download". The file list shows the following files and their commit history:

File	Commit Message	Commit Date
<code>Samples</code>	Windows 10 1511 Release - May 2016 Update	5 days ago
<code>SharedContent</code>	Windows 10 1511 Release - April 2016 Update	a month ago
<code>.gitattributes</code>	Windows 10 RTM Release - August 2015 Update	10 months ago
<code>.gitignore</code>	Windows 10 RTM Release - May 2016 Update	5 days ago
<code>LICENSE</code>	Windows 10 RTM updates	10 months ago
<code>README.md</code>	Windows 10 1511 Release - May 2016 Update	5 days ago

見るべき入門サイト、情報源

- WinRT/Metro TIPS
 - MVPの山本さんによるTips集
 - @ITの記事なので非常に丁寧
 - http://www.atmarkit.co.jp/ait/subtop/features/dotnet/app/winrttps_index.html
- かずきのBlog@Hatena
 - MVPの大田さんのブログ
 - 上級者向け
 - 最新技術についていち早く触れてくれる
 - <http://blog.okazuki.jp/>
- garicchi.com
 - 一応それなりの記事をそろえているつもりです
 - http://garicchi.com/?page_id=111
 - UWPの入門動画も作成しています
 - http://garicchi.com/?page_id=19366

ユーザーコミュニティ

- まどすた(旧めとべや)
 - <https://roommetro.doorkeeper.jp/>
 - ほぼ唯一とっていいUWPの勉強会を行っているコミュニティ
 - メインは東京、大阪でもやっています
 - 開催頻度は数か月に1度
 - 濃ゆい人が集まっている



まとめ

- Universal Windows AppはWindowsの新しいアプリの形である
- 様々なデバイス(IoT～HoloLensまで)においてUWPが動く
- 様々なプログラミング言語で作ることができる
- 無料で開発、学生なら無料でストアに公開することもできる
- 基本は3ステップ！

ご清聴ありがとうございました！
気軽に質問してください

