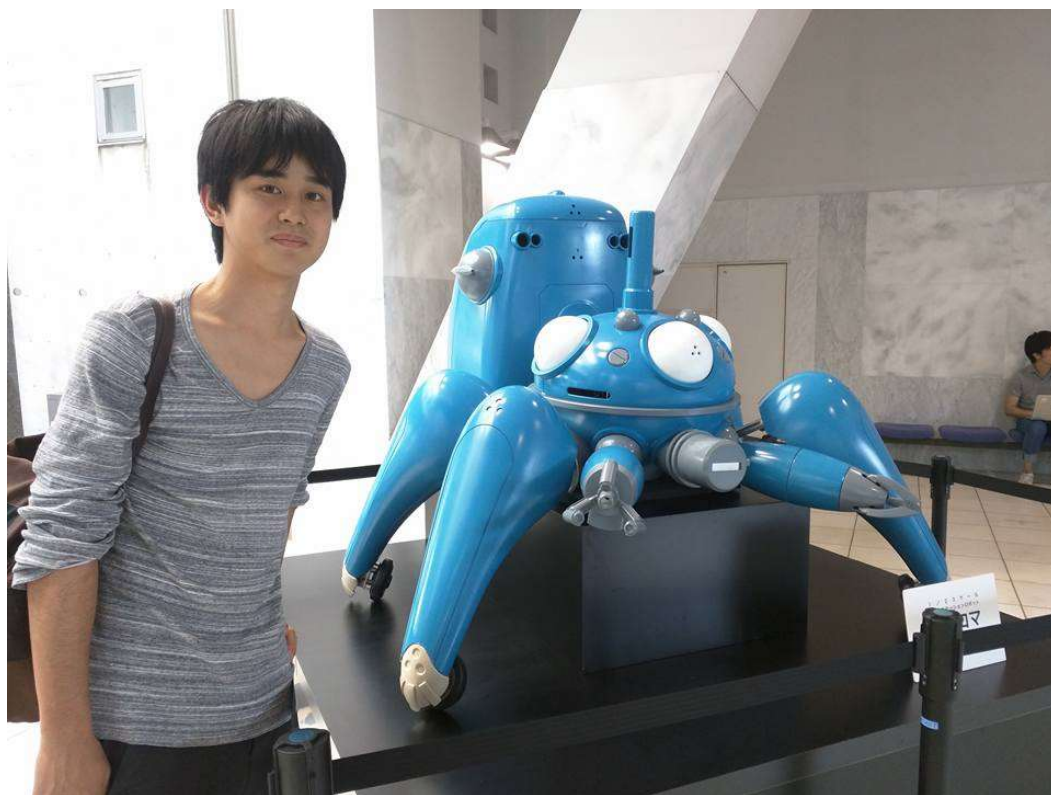


Web技術入門

ASP.NETの基礎と
Webアプリ作成方法

Microsoft Student Partners

自己紹介



奥村圭祐

Keisuke Okumura

東京工業大学
工学部 情報工学科3年

 @okkunokkun18

 kei.okumura18

目次

- 概要
 - ASP.NET について
 - 必要な開発環境、本番環境
 - ASP.NET を使用するメリット・デメリット
- 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

ASP.NETの基礎

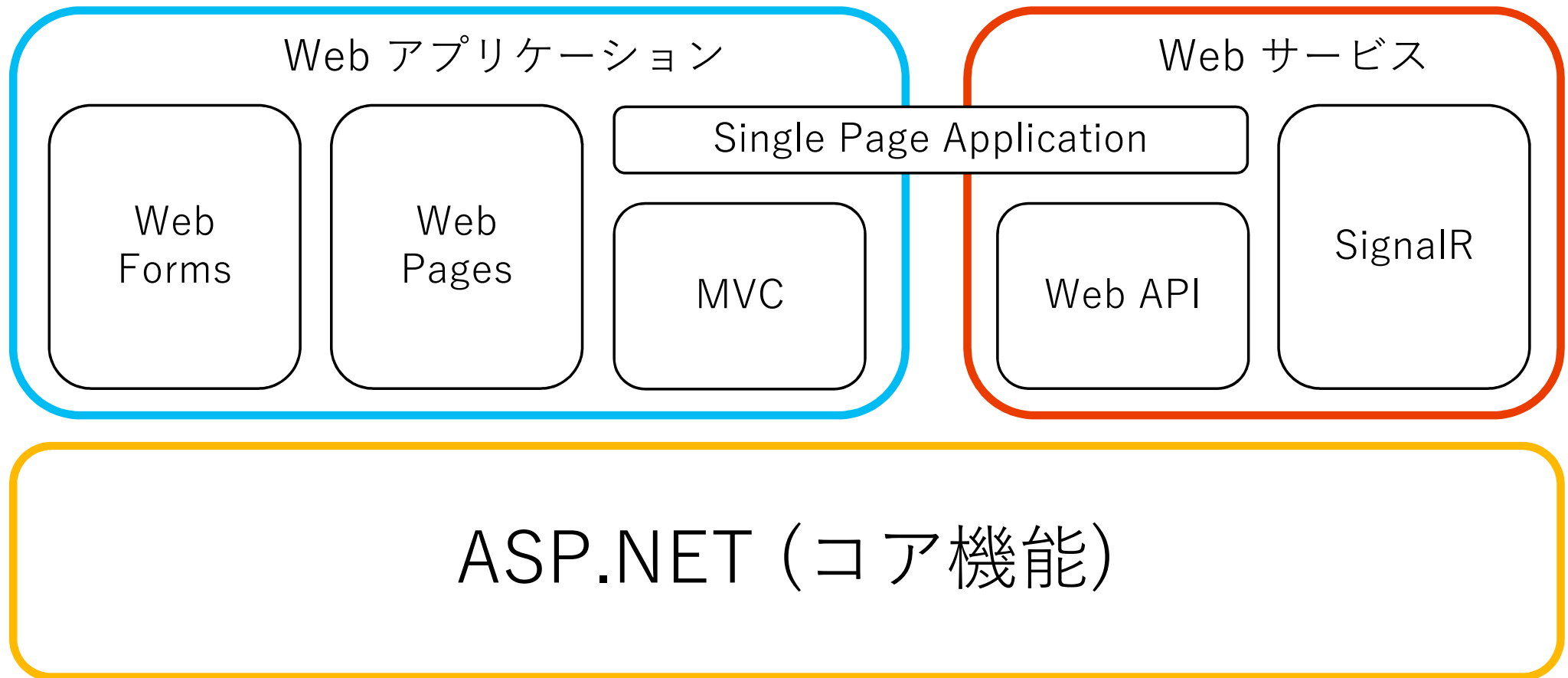
Microsoft Student Partners

ASP.NETとは① - 基本 -

- .NET Frameworkの上に構築されたWebアプリケーションのためのフレームワーク群
- Visual Studioと組み合わせることで、簡単に高機能なWebアプリケーションの開発をすることができる

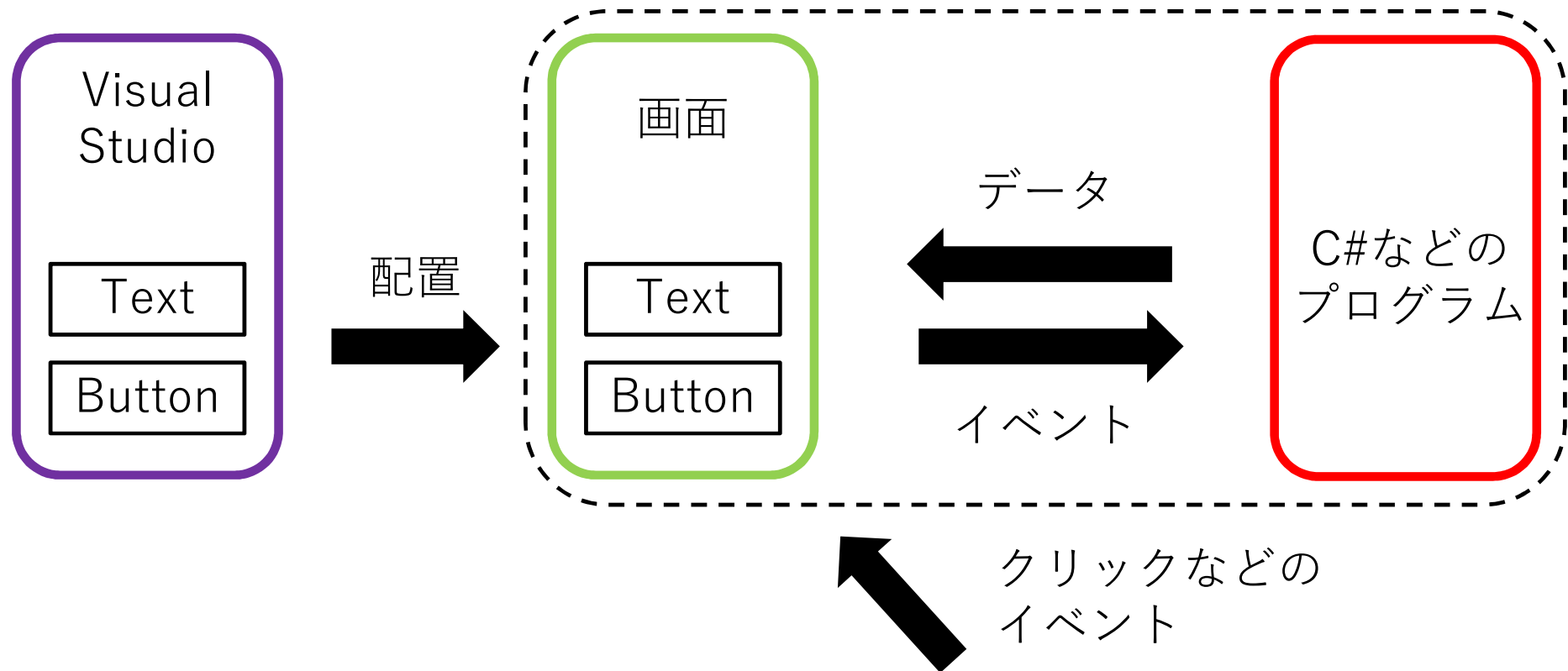
ASP.NET とは② - フレームワークの全体像 -

One ASP.NET



ASP.NETとは③ – Web Forms の仕組み –

Web Forms の仕組み



ASP.NETとは④ – Web Forms の特徴 –

- 特徴
 - ASP.NETフレームワークの中で一番歴史が古い
 - Windows のアプリケーションによく似たイベント駆動型システム
 - Windowsアプリの開発手法を知っていれば Web も簡単に作ることができた
- デメリット
 - APS.NET5からサポートされていない
 - Web Forms 独自のUIでは現代のWebのトレンドについていけなくなった

ASP.NETとは⑤ – Web Pages の仕組み・特徴 –

- Web Pages の仕組み



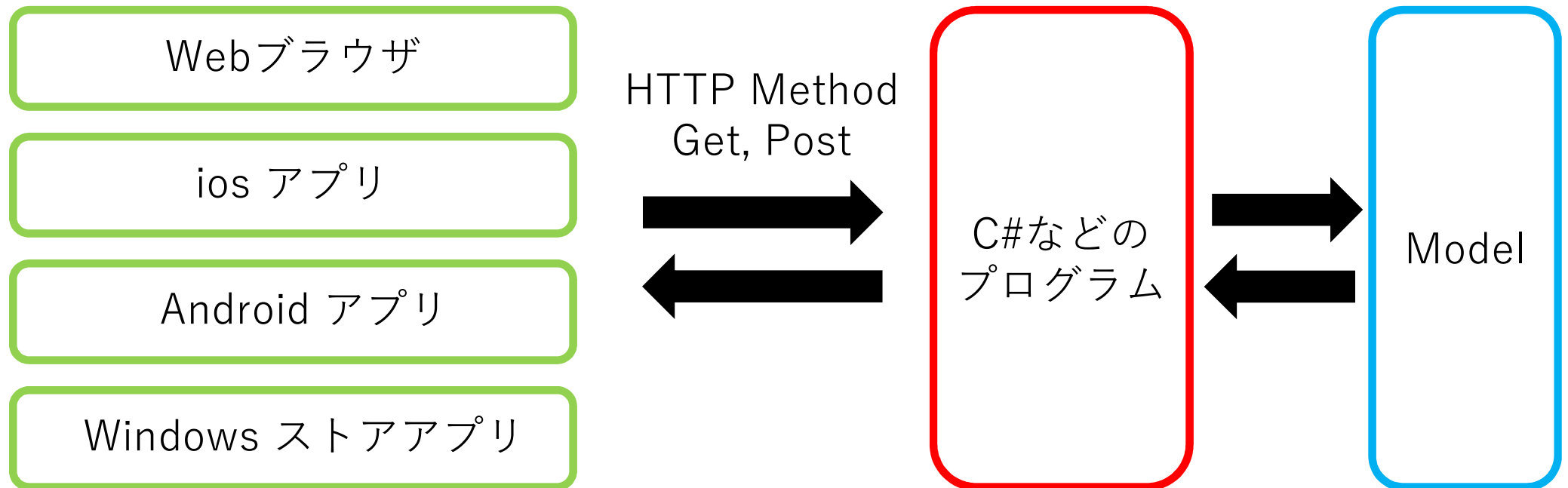
画面

- 特徴

- PHPによく似たHTMLの中にC#を埋め込む構成
- 小さいサイトで使いやすい
- とにかく気軽に開発できる

ASP.NETとは⑥ – Web API の仕組み・特徴 –

- Web API の仕組み

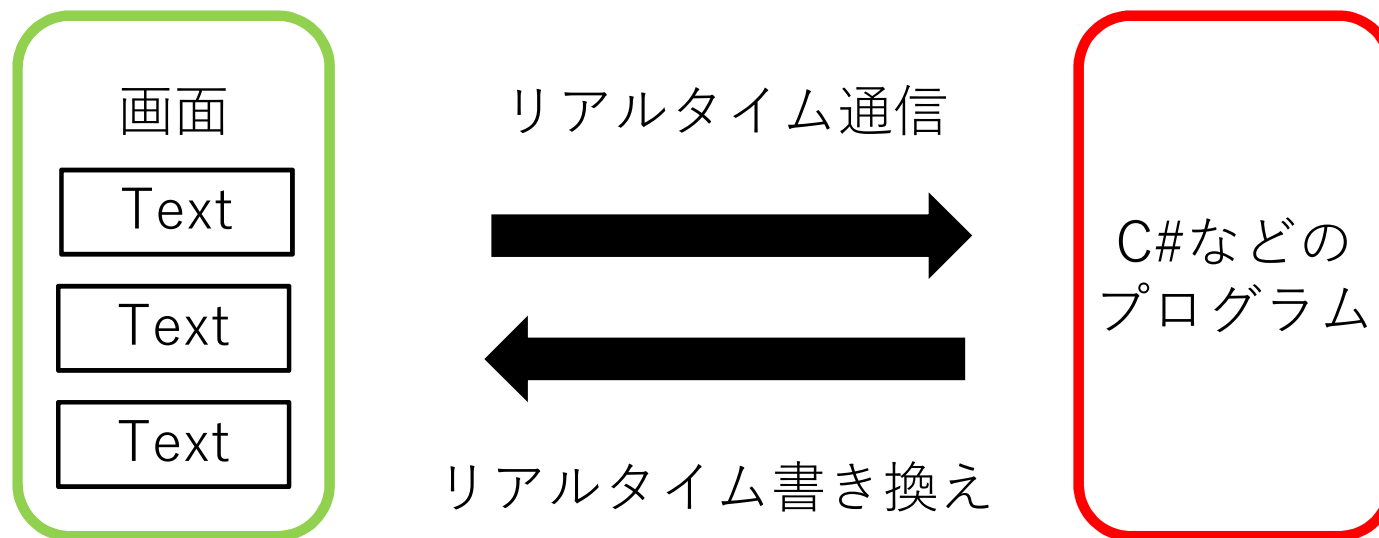


- 特徴

- クライアントを選ばない、
- twitter API のような Web API を使うことができる

ASP.NET とは⑦ – SignalR の仕組み・特徴 –

- SignalR の仕組み

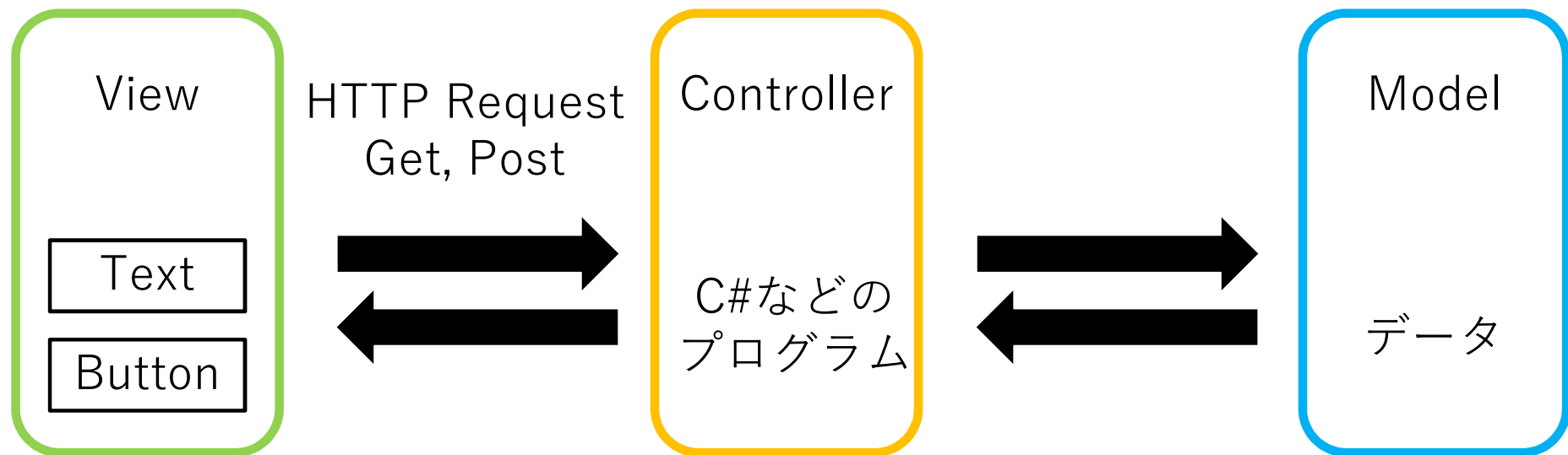


- 特徴

- チャットなどリアルタイムにサーバーと通信
- 基本は WebSocket を使用

ASP.NET MVCとは① – MVC の仕組み –

MVC の仕組み

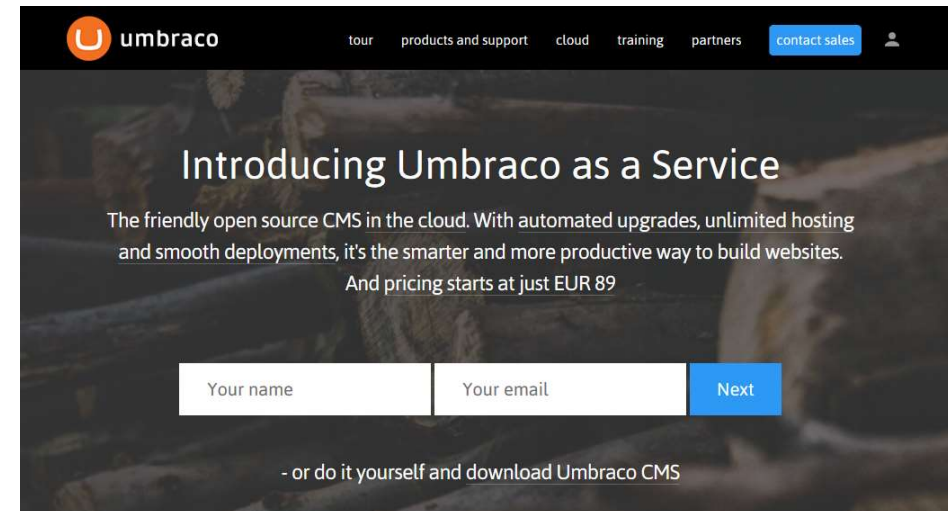


ASP.NET MVCとは② – 特徴 –

- MVCが出てきた背景
 - Web Forms には固有概念が多く、その特殊性ゆえに現在のトレンドに合わなくなってきた（イベントドリブン、ポストバックなど）
 - フロントサイドの出力を制御しにくい ← 綺麗じゃない！
 - テストを自動化しにくい…
- 特徴
 - (Model) – View – Controller パターンを採用
 - Java や PHP, Ruby のフレームワークを使って開発した経験がある人なら馴染みやすい！
 - ルーティング機能、認証・キャッシュ、Scaffolding 機能、Razor 構文、Entity Framework

ASP.NET MVCとは③ - 導入事例 -

- 導入事例



ASP.NET MVCとは④ - 開発環境 -

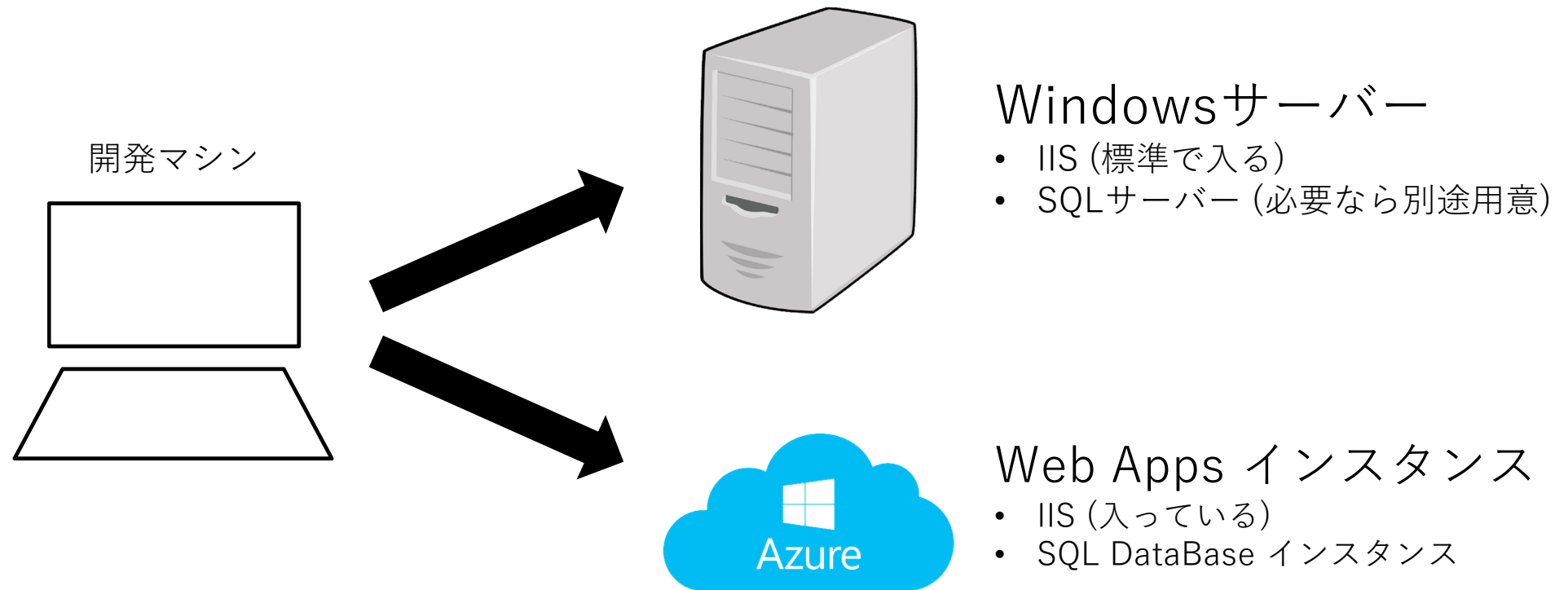
- 開発環境
 - 言語は C#, Visual Basic
 - OS : Windows7,8,10 (VS Code を使えば Mac でも一応...)
 - IIS Express (HTTPサーバー), SQL Server (DBサーバー, MySQLは非推奨), Visual Studio (統合開発環境)



Visual Studio Community 2015 を
Windows PCに入ればOK！！(無料)

ASP.NET MVCとは⑤ - 本番環境 -

- 本番環境 (DreamSparkで無料！)



ASP.NET MVCとは⑥ – メリット・デメリット –

- メリット
 - **Visual Studio** の強力なコーディング支援のもとで開発ができること！！（入力補完、デバッグなど）
 - Microsoft のクラウド Azure へのデプロイがスムーズ（DreamSpark を使えば無料でホスティングもできる）
 - 安全・高速に開発することができ、拡張しやすい
- デメリット
 - Windows での開発がメインになってしまう
 - オンラインのリファレンスは日本語が少ない

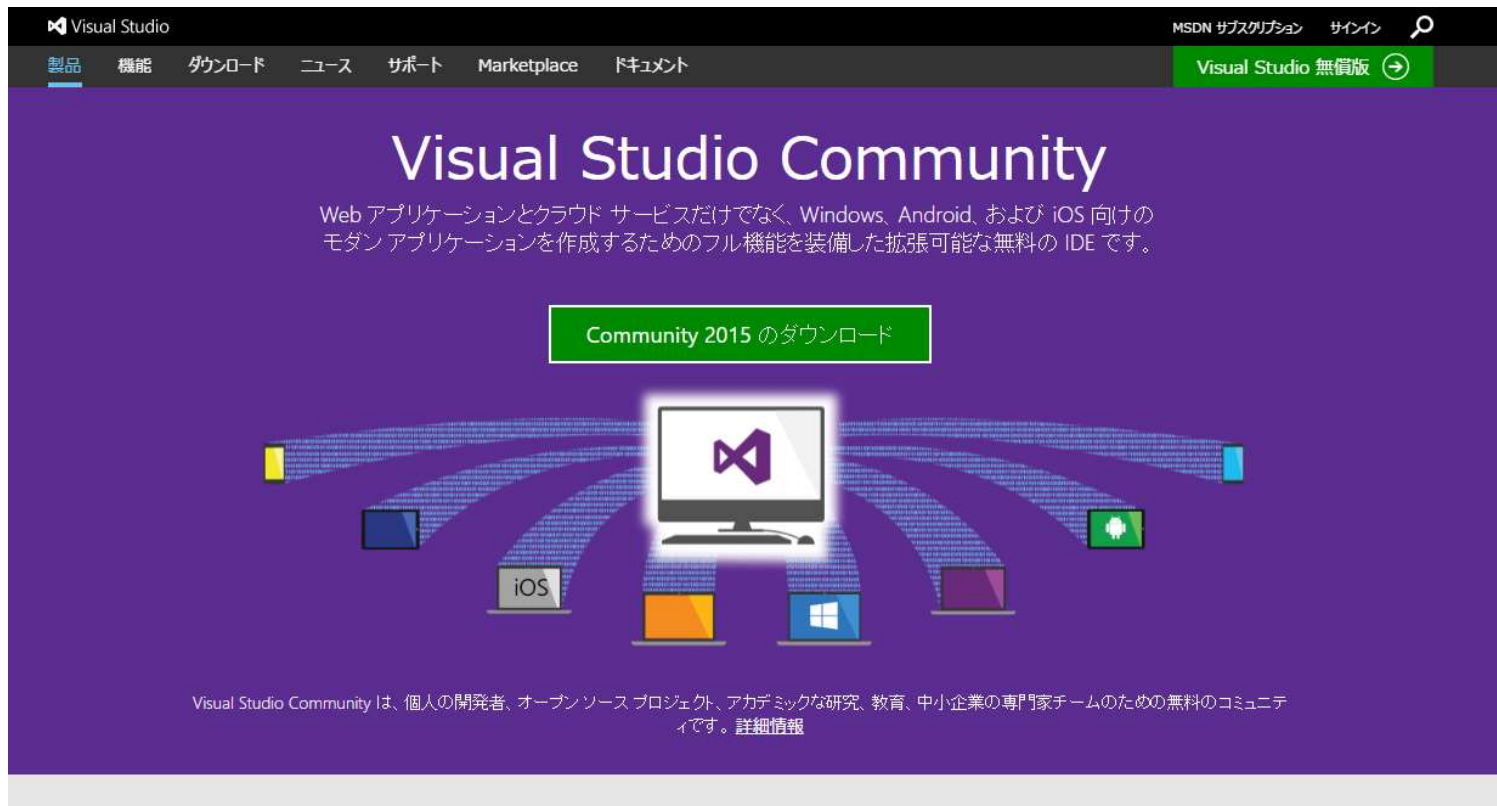
開発入門

新しいプロジェクト作成からデモまで

Microsoft Student Partners

開発環境構築方法

- Visual Studio Community 2015 をインストール



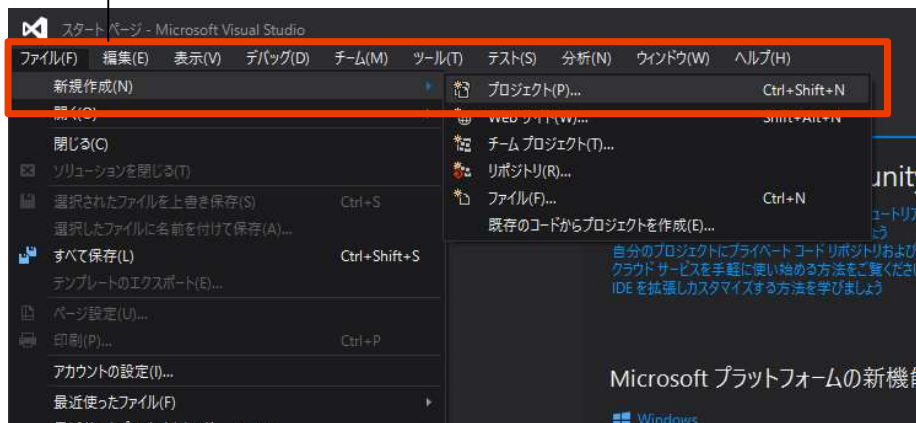
<https://www.visualstudio.com/ja-jp/products/visual-studio-community-vs.aspx>

デモ

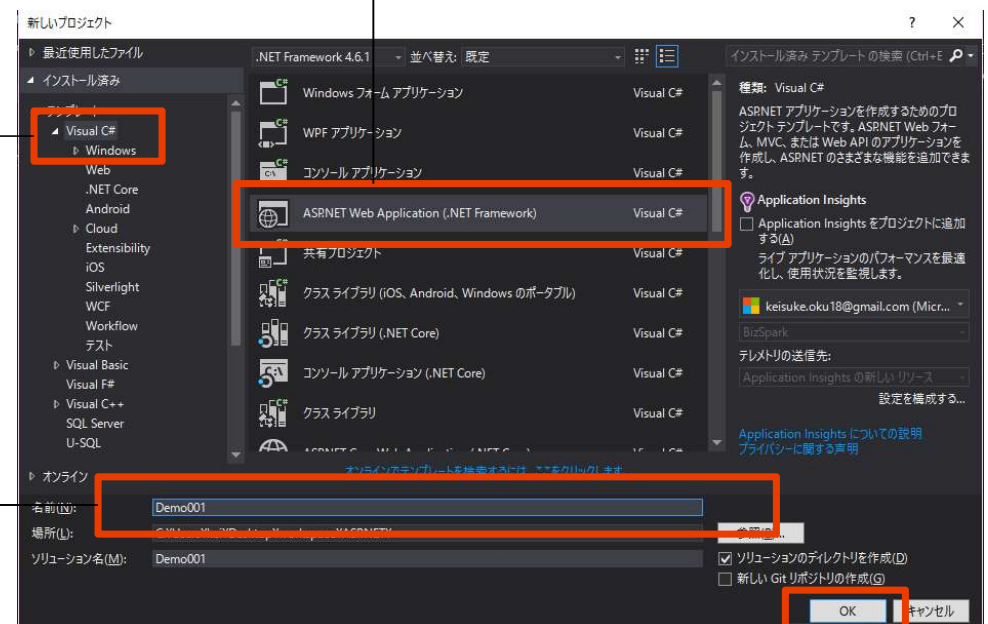
- 新規プロジェクト作成->開発->実行
- 自動生成されたファイル群の構成の説明
- 「Hello World」の出力
- 簡単なGetリクエストを受けとり、表示されるhtmlを変更するようなプログラムを解説、実行

新しくプロジェクトを作る①

① 「ファイル」 → 「新規作成」 → 「プロジェクト」



③ 「ASP.NET Web Application (.NET Framework)」



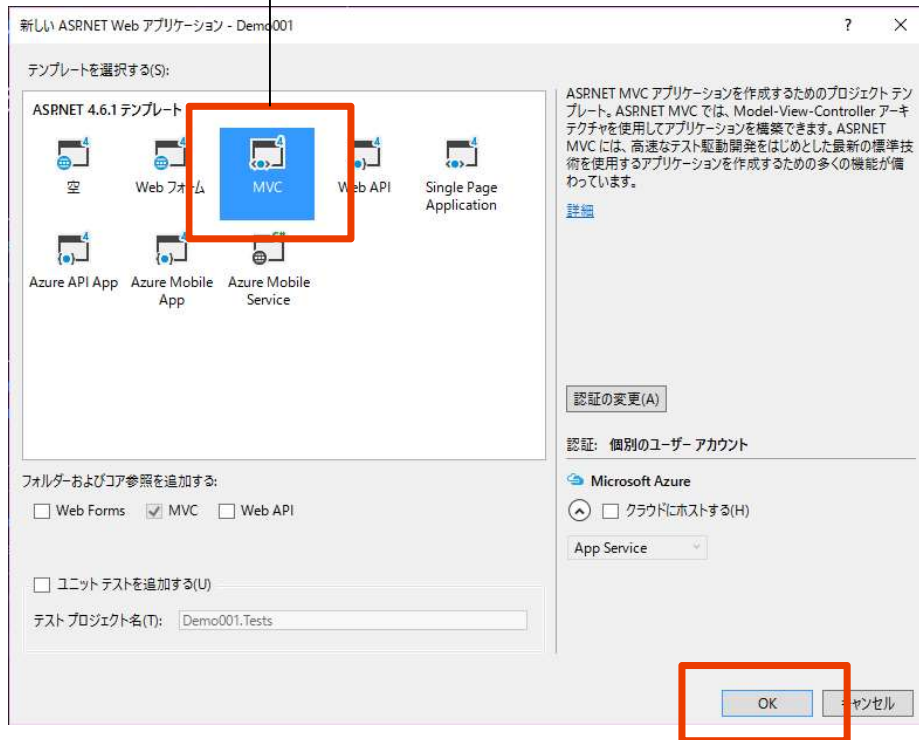
② 「Visual C#」

④ プロジェクトの名前を記入

⑤ 「OK」を押す

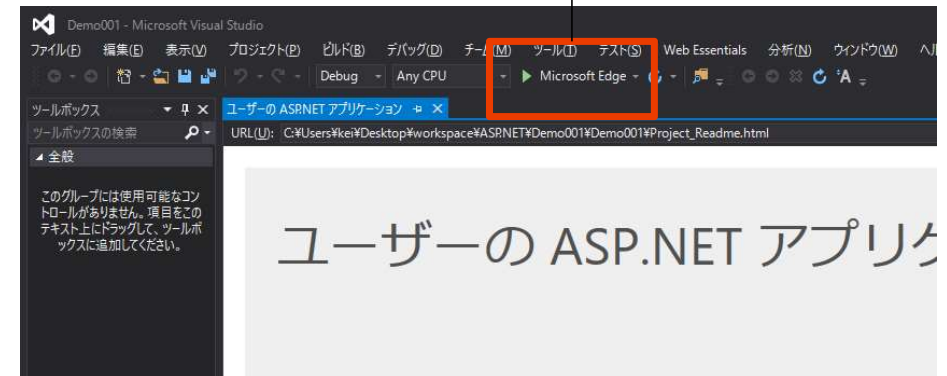
新しくプロジェクトを作る②

⑥ 「MVC」を選択

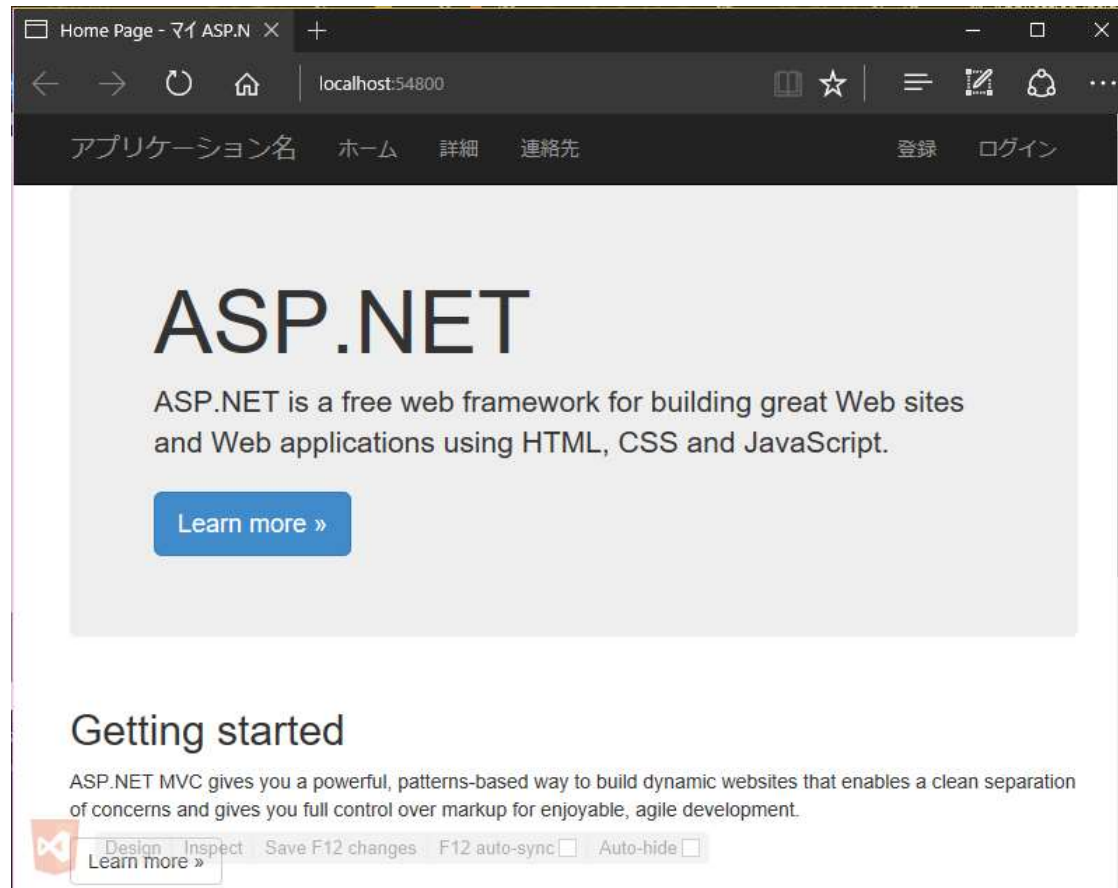


⑦ 「OK」を押す

⑧ 実行ボタン(緑の三角)を押す

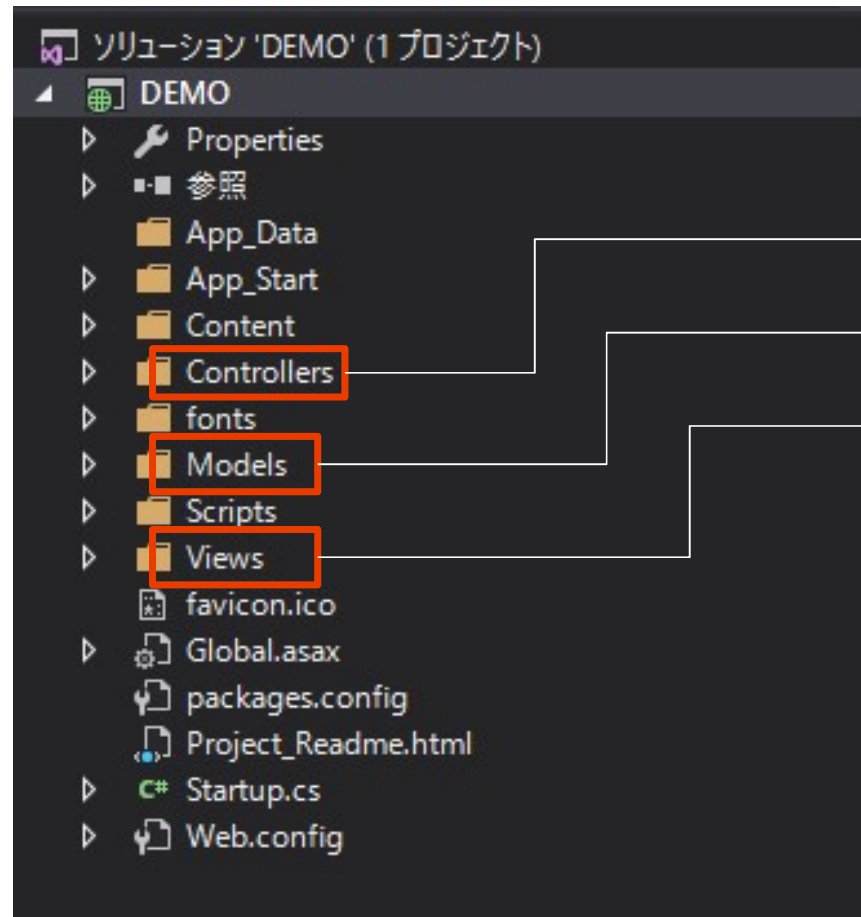


新しくプロジェクトを作る③



ビルドに成功！！

フォルダ構成について

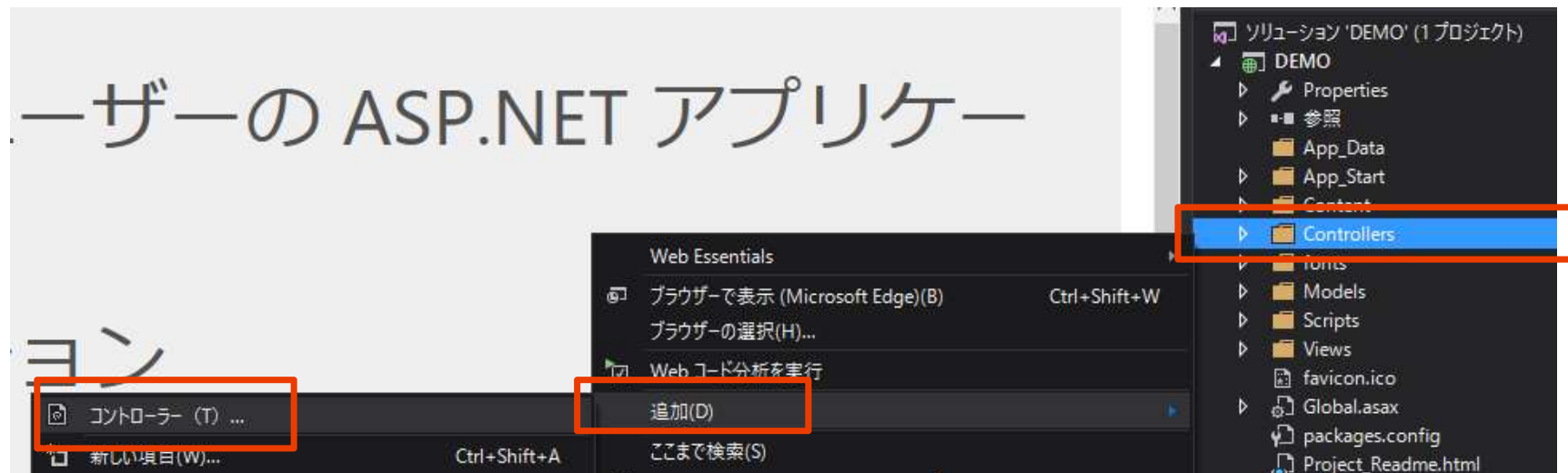


Controller

Model

View

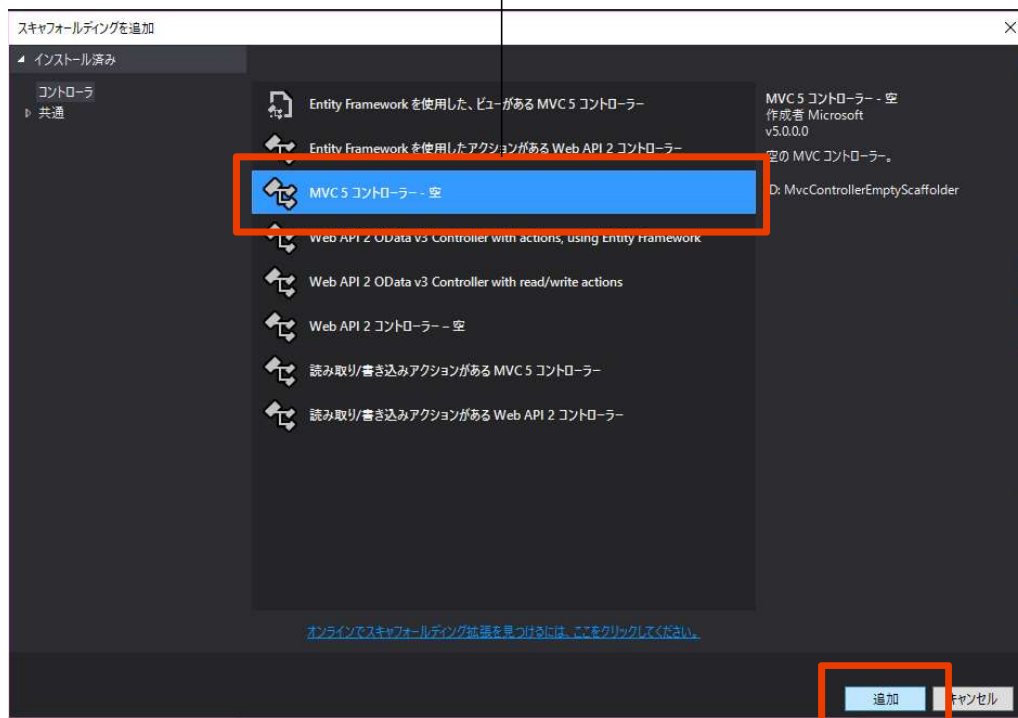
Controllerの追加①



①Controllersの上で右クリック後、「追加」->「コントローラー」を選択

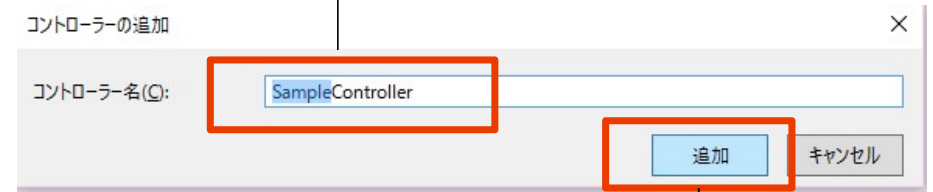
Controller の追加②

② 「MVC5 コントローラー・空」 を選択



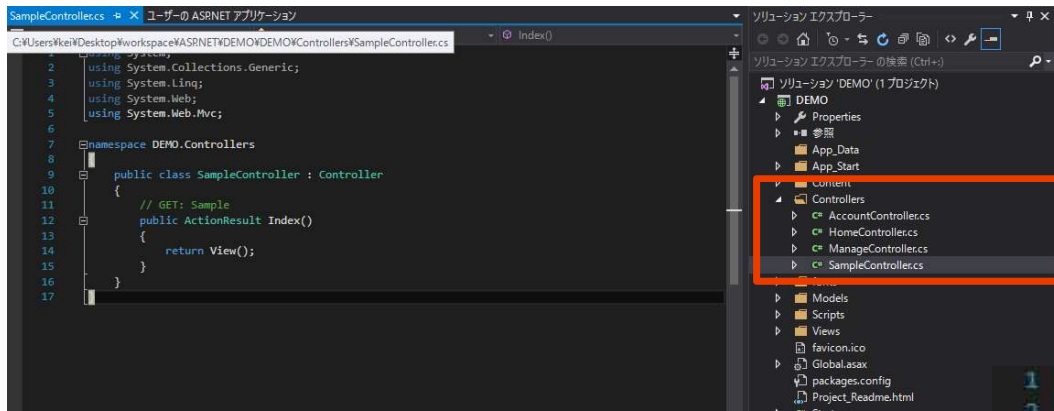
③ 「追加」 を押す

④ XxxController を記入



⑤ 「追加を押す」

Hello World の出力①



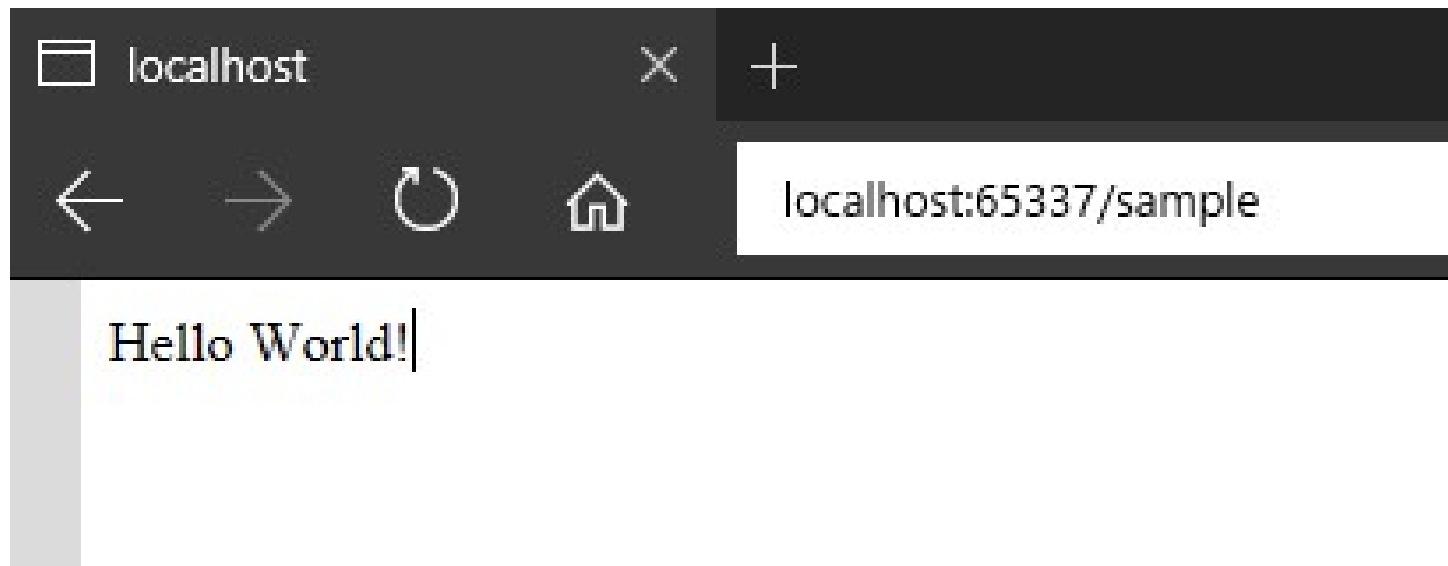
Controllersの中に先ほど作成した
ファイルがあるので、編集する

「Return Content(“Hello World”);」に変更

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.Mvc;
6
7 namespace DEMO.Controllers
8 {
9     public class SampleController : Controller
10     {
11         // GET: Sample
12         public ActionResult Index()
13         {
14             return Content("Hello World");
15         }
16     }
17 }
```

Hello World の出力②

/コントローラー名 にアクセスしてHello Worldが出力されればOK

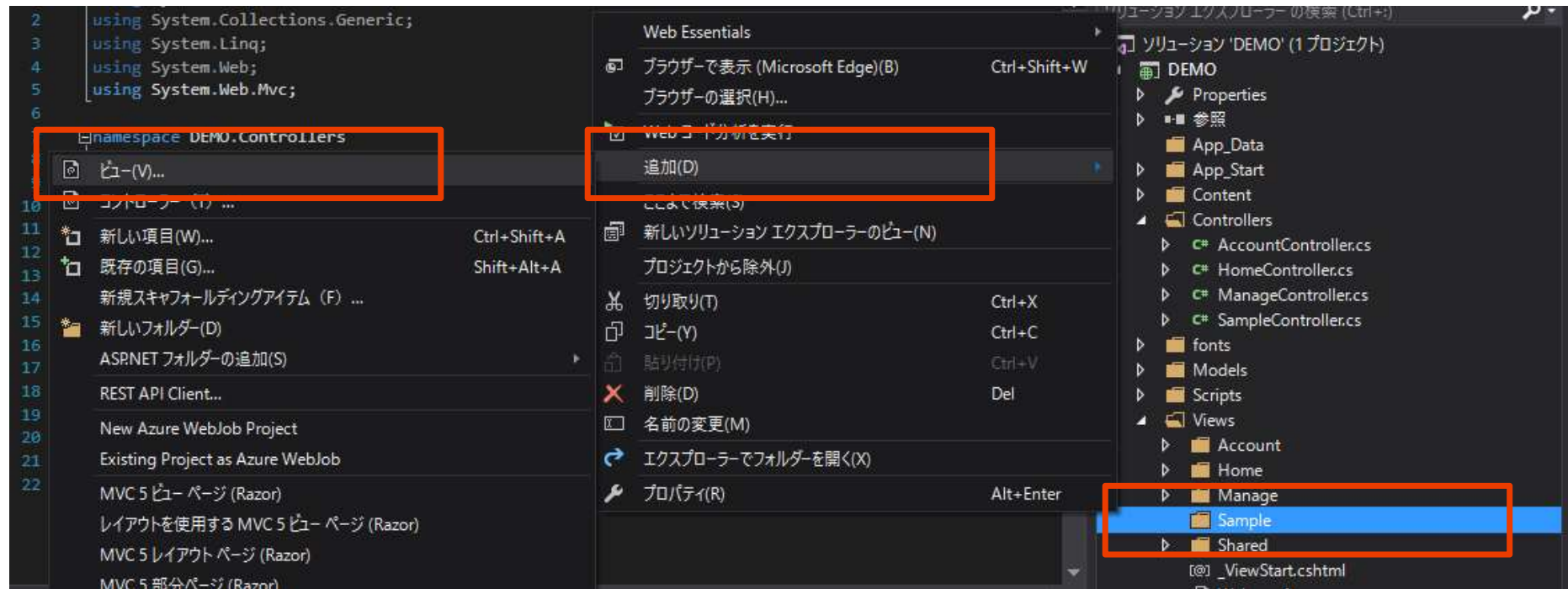


HTMLファイルを出力してみる①

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6
7  namespace DEMO.Controllers
8  {
9      public class SampleController : Controller
10     {
11         // GET: Sample
12         public ActionResult Index()
13         {
14             return Content("Hello World");
15         }
16
17         public ActionResult HtmlView()
18         {
19             return View();
20         }
21     }
22 }
```

この部分を追加

HTMLファイルを出力してみる②



「Views > コントローラー名」の上で右クリック後、「追加」->「ビュー」を選択

HTMLファイルを出力してみる③

名前を追加した関数名と同じにする

ビューの追加

ビュー名(N):

テンプレート(T): Empty (モデルなし)

モデルクラス(M):

データ コンテキスト クラス(D):

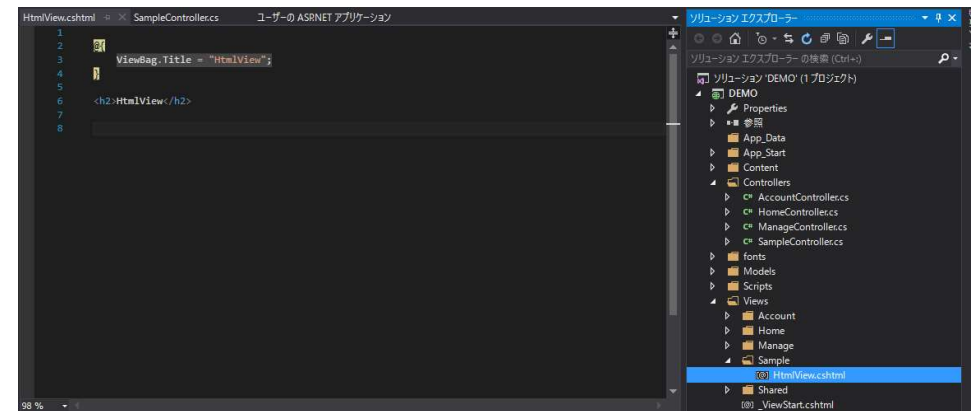
オプション:

☐ 部分ビューとして作成(C)

☒ スクリプトライブラリの参照(R)

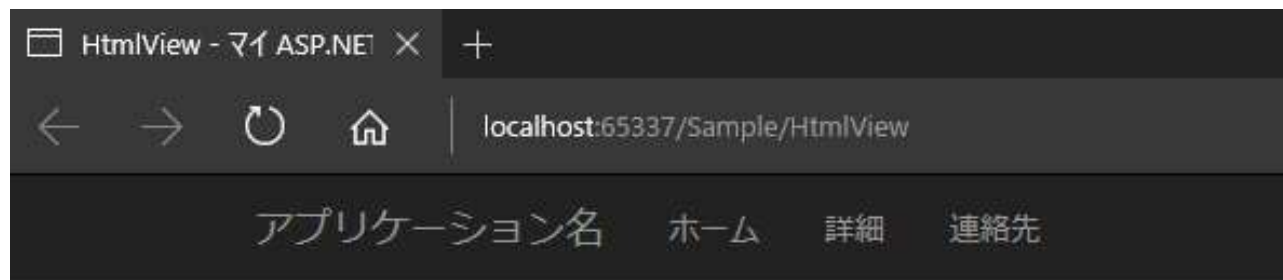
☒ レイアウト ページの使用(U):

(Razor _viewstart ファイルで設定する場合は空のままにしてください)



HTMLファイルを出力してみる④

/コントローラー名/関数名 にアクセスしてHTMLが出力されていればOK



HtmlView

© 2016 - マイ ASP.NET アプリケーション

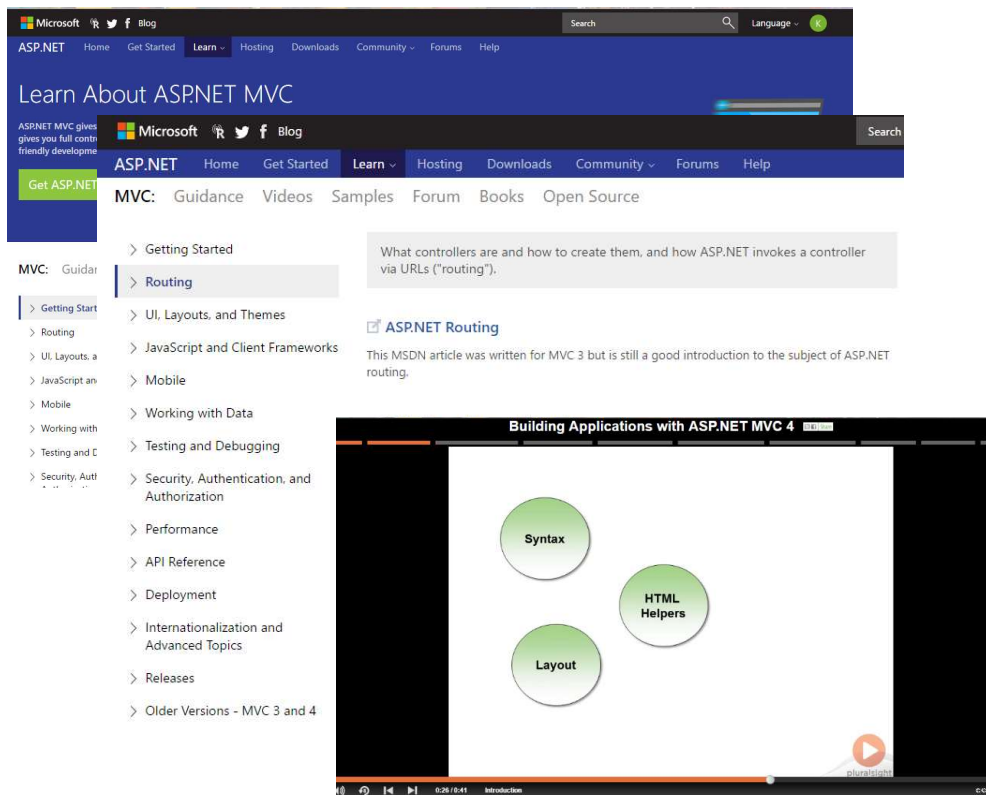
さらに学ぶには

公式サイトの活用、コミュニティの紹介

Microsoft Student Partners

公式サイトの見方

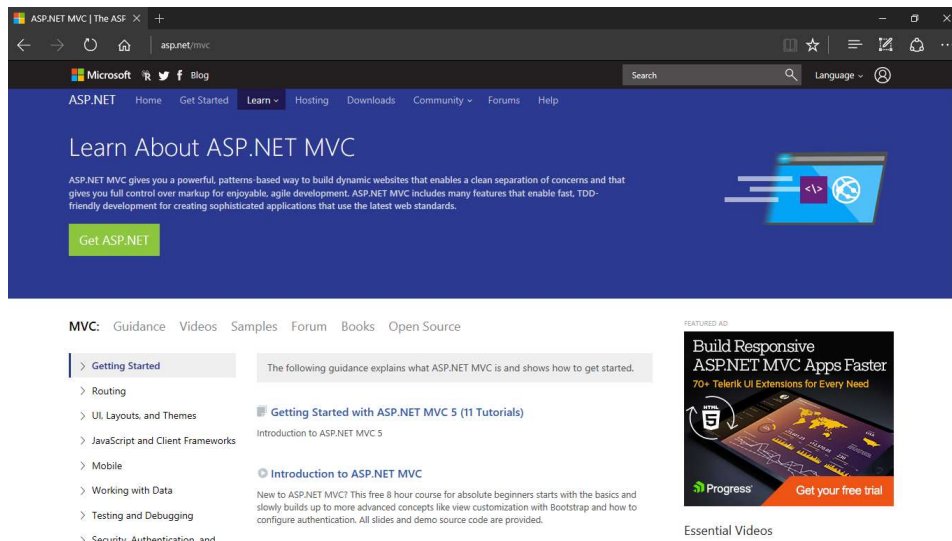
<http://www.asp.net/mvc> (公式サイト, 英語)



ビデオもかなりある！

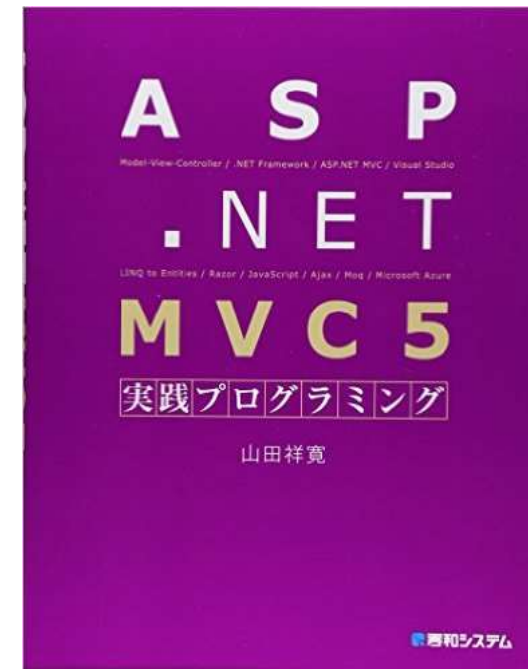
- プロジェクトの新規作成、基本的なHTML FormによるHttp Requestの処理ができるようになる
- 公式サイトのチュートリアルを一通りやって、MVCパターンでSQL Serverとのデータのやり取りができるようになる (<http://www.asp.net/mvc/overview/getting-started/introduction/getting-started>)
- 公式サイトでその他のパターンを学ぶ
 - Web API (<http://www.asp.net/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api>)
 - Single Page Application (<http://www.asp.net/single-page-application/overview/introduction/knockoutjs-template>)

見るべき入門サイト、情報源



<http://www.asp.net/mvc/overview/getting-started/introduction/getting-started>

公式サイトチュートリアル(英語)



ASP.NET MVC5 実践プログラミング
(定価 3500円)

ユーザーコミュニティ

- <https://forums.asp.net/> (フォーラム)
- <https://www.asp.net/community> (コミュニティ)
- <https://live.asp.net/> (Live Streaming, 1ヵ月に1回程度はやってる)

A promotional banner for the ASP.NET Community Standup. The background is a collage of images showing various community members. The text is centered and reads: "ASP.NET Community Standup" in large white letters. Below it, in smaller white text, is "Next Episode Streaming Live on", followed by "7 Jun, 2016 at 7:45 AM (東京 (標準時))" and "1w 7h 30m 12s". At the bottom, there is a button that says "Add to calendar" with a dropdown arrow, and a line of text that says "Check back here to watch & ask questions then!".

Microsoft

ASP.NET Community Standup

Next Episode Streaming Live on
7 Jun, 2016 at 7:45 AM (東京 (標準時))
1w 7h 30m 12s

Add to calendar ▾

Check back here to watch & ask questions then!

まとめ

- これからの ASP.NET は Web Forms ではなく、MVCが主役
- Visual Studio や Azure のサポートが強力だから、高速に高機能なWebアプリを開発することができる！
- 残念ながら、日本語のサポートはそれほど充実していない…

ご清聴ありがとうございました

Microsoft Student Partners