

1. Visual Studio の開発環境構築

以下のリンク先にある使用許諾契約「Microsoft Visual Studio Community 2026」を確認してください。

比較的規模の大きな法人・機関等の場合で、製品開発を行う場合には、有償の「Professional 版」または「Enterprise版」が必要です。

<https://visualstudio.microsoft.com/ja/license-terms/>

1-1 Microsoft Visual Studio Community のライセンス条項（概略）

Visual Studio Community が使えない場合の多くは、「エンタープライズ組織」で使うまたは使う予定の製品開発を請け負っているかどうかです。IT企業が開発作業を行う場合には、「Professional 版」または「Enterprise版」を購入する前提で考えてください。

1) Visual Studio Community について

Visual Studio Community は、無償ですべての機能を搭載したあらゆるプラットフォームやデバイスのエンタープライズ（※）向けではないアプリケーション向けのIDEです。Visual Studio Community はエンタープライズ向けではないアプリケーションを作成するためのパワフルで生産性の高い機能や、Windows や iOS や Android 向けのモバイル アプリケーション開発ツールや、多数の拡張機能にアクセスすることを含んだすべての機能を保有しています。

※ 大企業や中堅企業、公的機関など、複数の部門で構成されるような比較的規模の大きな法人・機関という意味で「エンタープライズ」という語が用いられています。またその組織とは、PC 台数 250 台以上または年商 100 万ドル（約 1 億円）以上を有する場合を指します。

2) Visual Studio Community が利用可能な者

Visual Studio Community の製品使用権は以下に説明するような顧客セグメントと利用シナリオに依存します。

個人開発者

あらゆる個人開発者は Visual Studio Community を利用でき、無償または有償のアプリケーションを作成できます。

組織

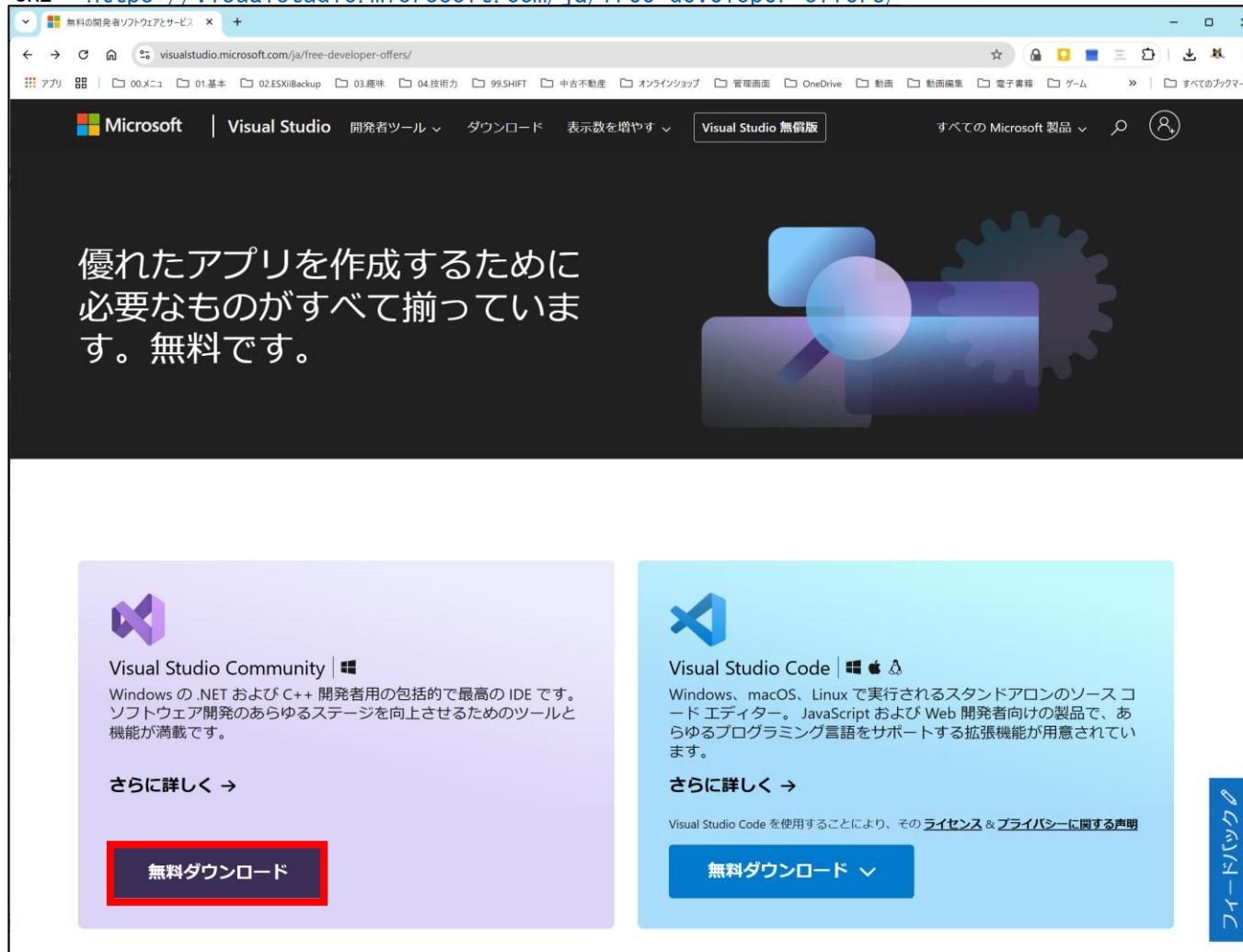
- 以下のシナリオでは Visual Studio Community を利用者数に制限なく利用することができます:
教室の研修環境、学術的調査、オープン ソース プロジェクトへの貢献
- その他のすべての利用シナリオ:
エンタープライズではない組織において 5 ユーザーまでは Visual Studio Community を利用できます。エンタープライズ組織は上記の教室の研修環境、学術的調査、オープン ソース プロジェクトへの貢献以外では従業員や請負契約者ともに利用を認められません。

1-2 インストール

1) ダウンロード

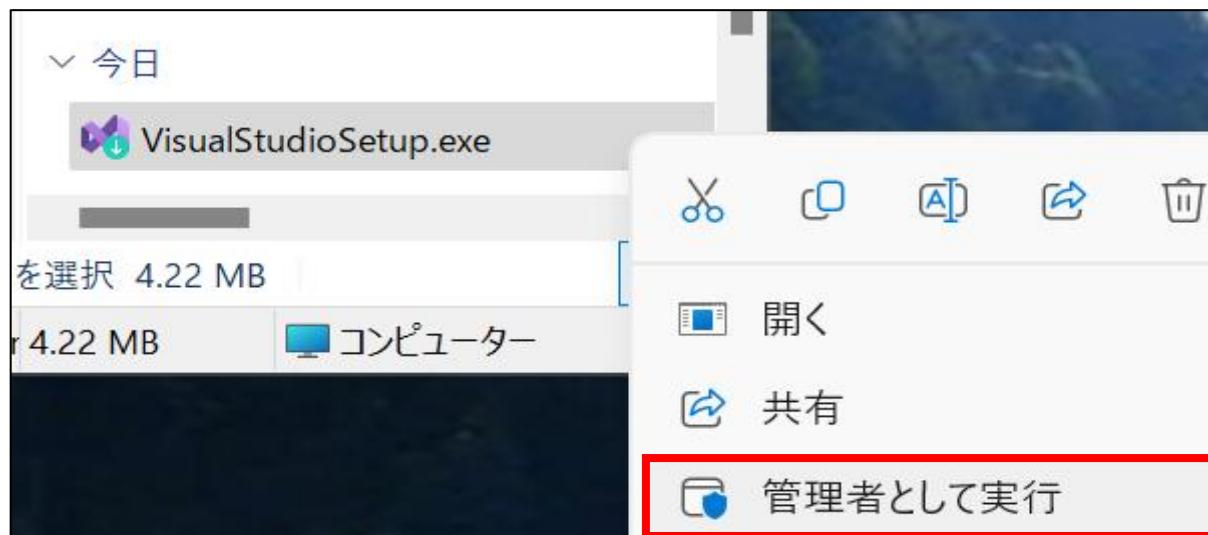
以下からVisual Studio Community をダウンロードしてください。

URL: <https://visualstudio.microsoft.com/ja/free-developer-offers/>

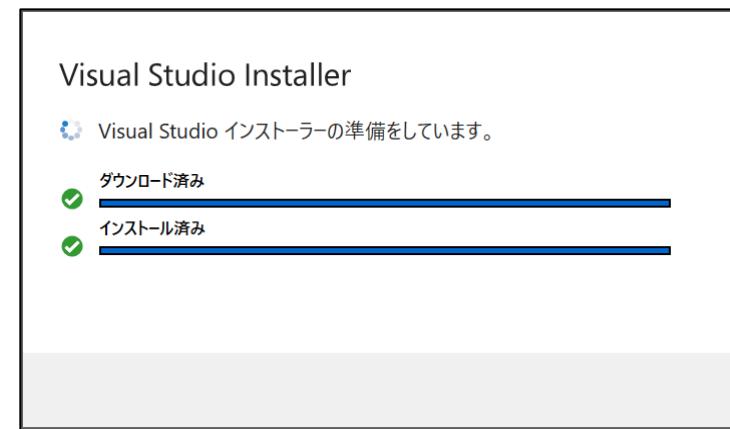
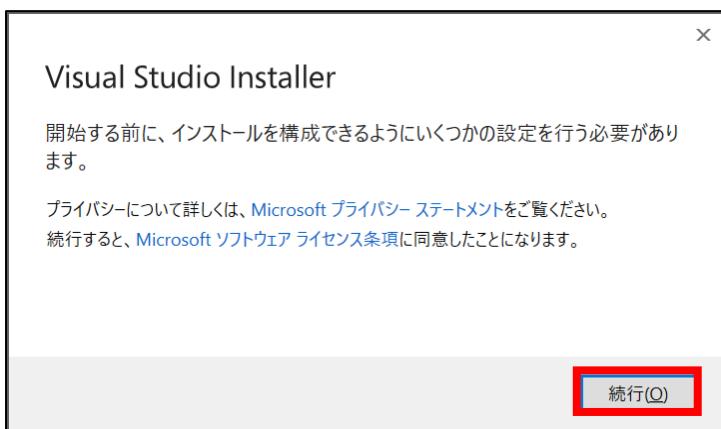


2) インストール

インストーラを管理者として実行します。



管理者として実行

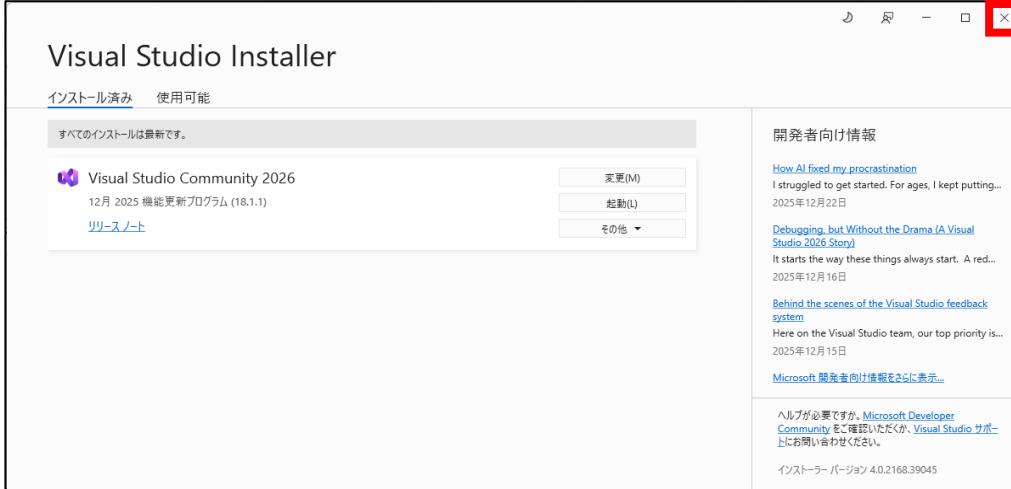


以下が表示されます。

「ワークロード」タブで、「.NET マルチプラットフォーム アプリのUI開発」にチェックをつける。それ以外にはチェックしない。チェックをつけ終わったら「インストール」を押下します。



インストールが完了すると、以下が表示されます。「×」を押して終了します。
また、Visual Studioが起動するので終了します。



3) Prism .NET MAUIテンプレートのインストール

Windows Power Shellを管理者権限で起動します。



.NET環境へ「Prism .NET MAUIテンプレート」をインストールし、Visual Studioのメニューから選択できるようにします。

```
dotnet new install Prism.Templates@8.1.97
```

実行結果

```
選択管理者: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

新機能と改善のために最新の PowerShell をインストールしてください!https://aka.ms/PSWindows

PS C:\WINDOWS\system32> dotnet new install Prism.Templates@8.1.97

.NET 9.0 へようこそ!
SDK バージョン: 9.0.100

テレメトリ
.NET ツールは、エクスペリエンスの向上のために利用状況データを収集します。データは Microsoft によって収集され、コミュニティと共有されます。テレメトリをオプトアウトするには、好みのシェルを使用して、DOTNET_CLI_TELEMETRY_OPTOUT 環境変数を '1' または 'true' に設定できます。

.NET CLI ツールのテレメトリの詳細をご覧ください: https://aka.ms/dotnet-cli-telemetry

ASP.NET Core HTTPS 開発証明書をインストールしました。
証明書を信頼するには、「dotnet dev-certs https --trust」を実行します
HTTPS の詳細情報: https://aka.ms/dotnet-https

最初のアプリを作成するには、https://aka.ms/dotnet-hello-world を参照してください
最新情報については、https://aka.ms/dotnet-whats-new を参照してください
ドキュメントを探すには、https://aka.ms/dotnet-docs を参照してください
Github で問題の報告とソースの検索を行うには、https://github.com/dotnet/core を参照してください
'dotnet --help' を使用して使用可能なコマンドを確認するか、https://aka.ms/dotnet-cli にアクセスしてください

次のパッケージがインストールされます:
Prism.Templates@8.1.97

成功: Prism.Templates@8.1.97により次のテンプレートがインストールされました。
テンプレート名 短い名前 言語 タグ
-----
```

テンプレート名	短い名前	言語	タグ
Prism .NET MAUI App	prism-maui	[C#]	MAUI/Android/iOS/macOS/Mac Catalyst/Windows/Tizen
Prism Blank App (Uno Platform)	uno-blank	[C#]	Prism/Xamarin.Uno Platform/WebAssembly/iOS/Android/WinUI/UWP
Prism Blank App (WPF)	wpf-core-blank	[C#]	Desktop
Prism Blank App (Xamarin.Forms)	xf-blank	[C#]	Prism/Xamarin/Xamarin.Forms
Prism Full App (WPF)	wpf-core-full	[C#]	Desktop
Prism Module (WPF)	wpf-module-core	[C#]	Desktop
Prism Module (Xamarin)	xf-module	[C#]	Prism/Xamarin/Xamarin.Forms

```
PS C:\WINDOWS\system32>
```

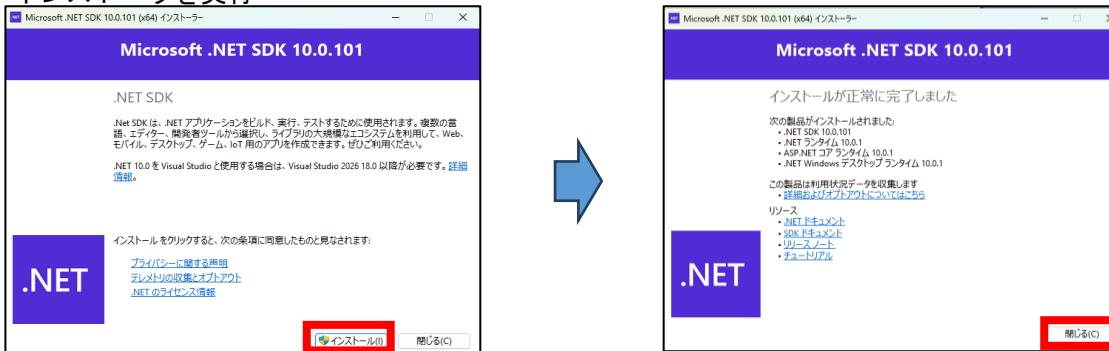
4) .net 10.0 のインストール

以下から、.net 10.0 のインストーラをダウンロードします。通常はWindowsのx64版になります。
「x64」を選択するとインストーラがダウンロードされるので、インストールします。

<https://dotnet.microsoft.com/ja-jp/download/dotnet/10.0>

The screenshot shows the Microsoft .NET download page for version 10.0.1. It displays two main sections: 'SDK 10.0.101' and 'ASP.NET Core ランタイム 10.0.1'. The 'SDK 10.0.101' section includes a table for Windows where the 'x64' link is highlighted with a red box. A large blue arrow points from this table to the 'Microsoft .NET SDK 10.0.101 (x64) インストーラー' window.

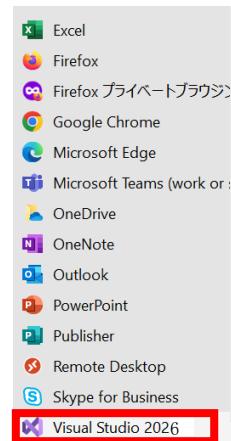
インストーラを実行



5) Prism .NET MAUI テンプレートを使ったサンプルアプリの起動

①プロジェクト作成

Visual Studio を起動します。



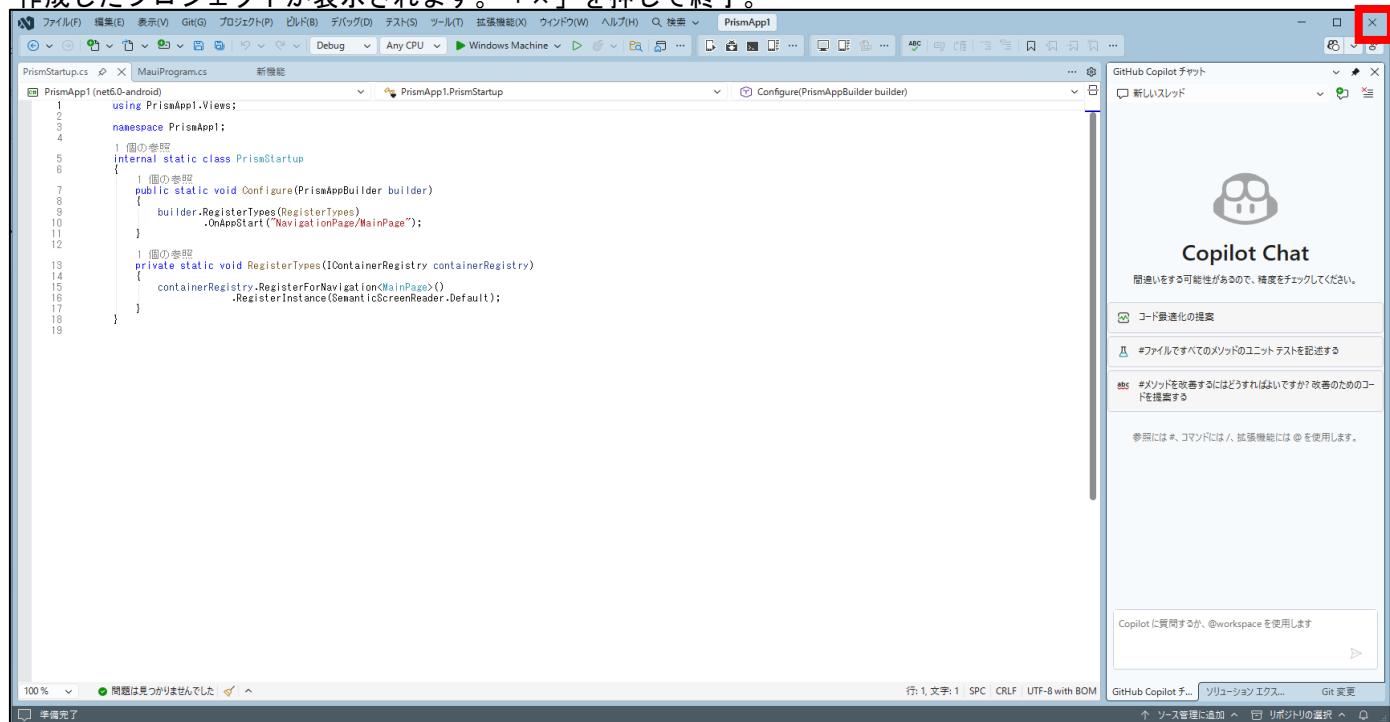
「Prism .NET MAUI App」を選択し、次へを押下



プロジェクト名と場所を任意に設定後、「作成 (C) 」を押下する。



作成したプロジェクトが表示されます。「×



②プロジェクトの修正

このプロジェクトは、.net6.0をターゲットにしていますが、.net10.0をターゲットにしないと動きません。

プロジェクトを作成したフォルダーにあるプロジェクトファイル（拡張子が.csproj）をテキストエディターで開いて書き替えます。

■修正前

```
<TargetFrameworks>net6.0-android;net6.0-ios;net6.0-maccatalyst</TargetFrameworks>
```

```
<TargetFrameworks Condition="$(MSBuild)::IsOSPlatform('windows'))">$(TargetFrameworks);net6.0-windows10.0.19041.0</TargetFrameworks>
```

■修正後

```
<TargetFrameworks>net10.0-android;net10.0-ios</TargetFrameworks>
```

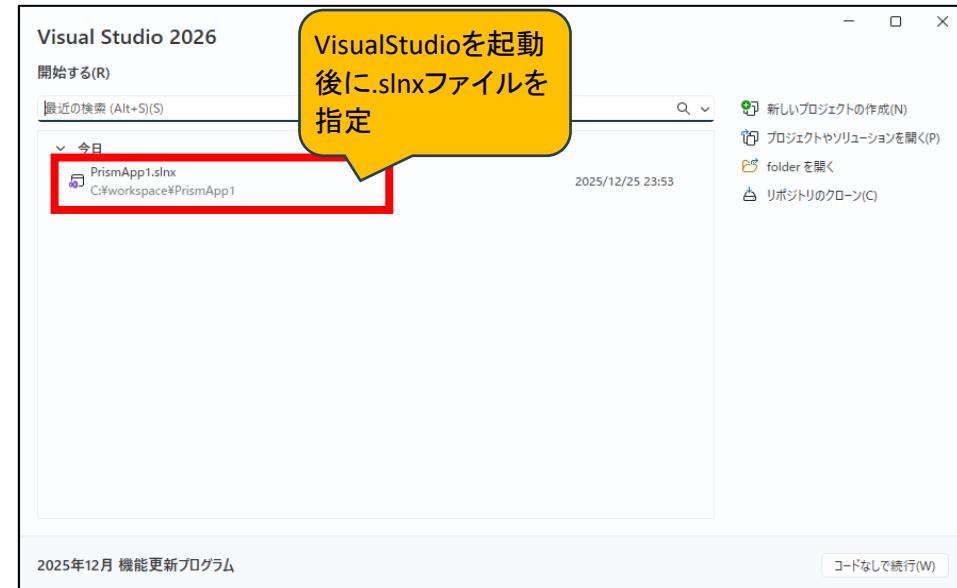
```
<TargetFrameworks Condition="$(MSBuild)::IsOSPlatform('windows'))">$(TargetFrameworks);net10.0-windows10.0.19041.0</TargetFrameworks>
```

③実行

.slnx形式のソリューションファイル「PrismApp1.App.slnx」をエクスプローラ上でダブルクリックするか、VisualStudioを起動後に選択します。



または

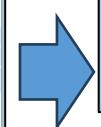


起動直後

A screenshot of Visual Studio showing the PrismApp.cs code in the editor. The code defines a static class PrismStartup with methods Configure and RegisterTypes. It also contains a private static void RegisterTypes method. A GitHub Copilot Chat window is open, providing suggestions for the code.

```
1  namespace PrismApp1
2  {
3      internal static class PrismStartup
4      {
5          // 開発者用
6          public static void Configure(PrismBuilder builder)
7          {
8              builder.RegisterTypes(RegisterTypes);
9              .OnActivate("NavigationPage/MainPage");
10         }
11     }
12
13     // 開発者用
14     private static void RegisterTypes(IContainerRegistry containerRegistry)
15     {
16         containerRegistry.RegisterForNavigation<MainPage>()
17             .RegisterInstance(BindingContext.Default);
18     }
19 }
```

表示⇒ソリューションエクスプローラを選択



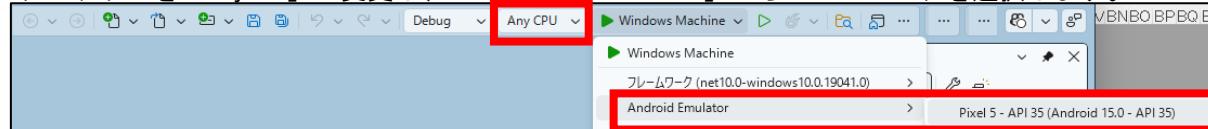
A screenshot of Visual Studio showing the Solution Explorer window open, with the 'Solution Explorer' tab selected. The 'PrismApp1' project is listed. The code editor shows the same PrismApp.cs code as the first screenshot. The entire Solution Explorer window is highlighted with a red box.

```
1  using PrismApp1.Views;
2  namespace PrismApp1
3  {
4      internal static class PrismStartup
5      {
6          // 開発者用
7          public static void Configure(PrismBuilder builder)
8          {
9              builder.RegisterTypes(RegisterTypes);
10             .OnActivate("NavigationPage/MainPage");
11         }
12
13         // 開発者用
14         private static void RegisterTypes(IContainerRegistry containerRegistry)
15         {
16             containerRegistry.RegisterForNavigation<MainPage>()
17                 .RegisterInstance(BindingContext.Default);
18         }
19 }
```

以下は、Androidの例です。

ターゲットの変更

ターゲットを「AnyCPU」に変更し、「Android Emulator」からエミュレータを選択します。



また、以下のようなSDKのインストール画面が出たら、「同意」を押下してインストールします。



ビルド

「ビルド」→「ソリューションのクリーン」をしてから「ビルド」→「ソリューションのビルド」を実行します。

エラーがないことを確認します。

実行

▶ を押下して、デバッグ実行します。

実行結果

PCのスペックによっては、Androidシミュレータの起動まで、かなり待たされます。

以上

実行結果

