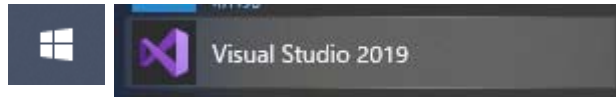


Visual Studioを使ってみよう!

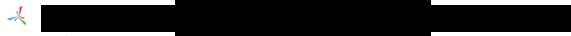
- Visual Studioを実行する



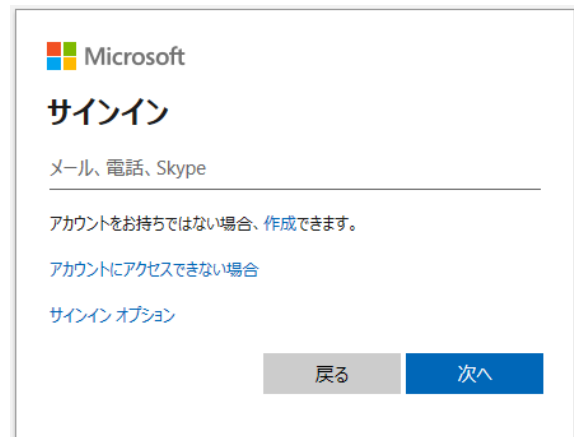
- [サインイン]というボタンがあるウィンドウが表示された場合(表示されない場合は次ページへ)
- [サインイン]というボタンをクリック



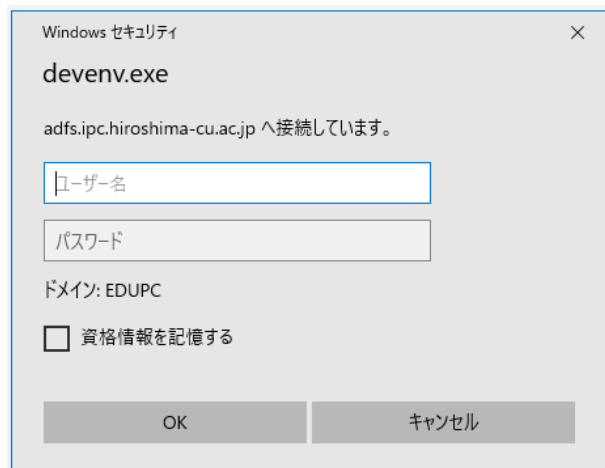
- 電子メール入力欄に [redacted] のメールアドレスを入力する



- [次へ]をクリック

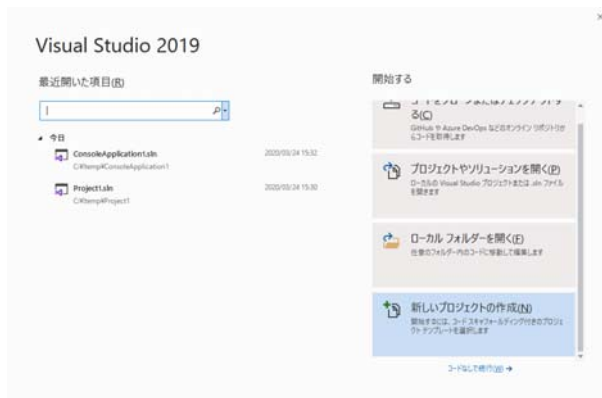


- ユーザー名とパスワードを入力する
(資格情報を記憶しておいたほうがいい)
- [OK]をクリック



Visual Studioを使ってみよう!

✖ [新しいプロジェクトの作成]をクリック



✖ [コンソール アプリ]を選ぶ

✖ [次へ]をクリック



✖ [プロジェクト名][ソリューション名]を入力(この例では[cg2])

✖ [場所]ボタンでフォルダを指定

✖ 分かりやすい場所にする

✖ ログアウトしても、そのファイルが消えてしまわないような場所にする (Fドライブ)

✖ [作成]をクリック

✖ そのフォルダに同じ名前のファイル名があるとうまくいかないので注意

ローカルフォルダに置いたファイルはシステムにより定期的に自動的に削除されます。個人フォルダに置けば削除されないので個人フォルダに置くようにしてください。

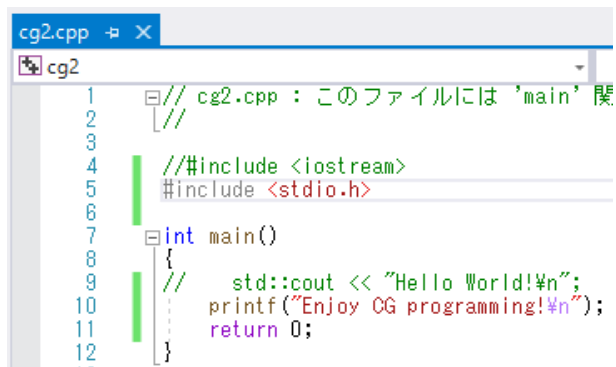
新しいプロジェクトを構成します



✖ 適当なプログラムを入力

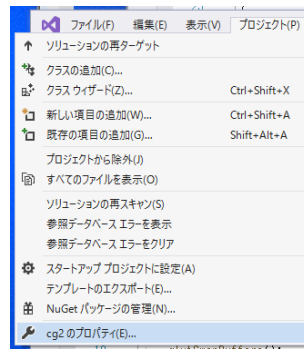
✖ [ビルド]-[ソリューションのビルド]をクリック

✖ [デバッグ]-[デバッグの開始]をクリック

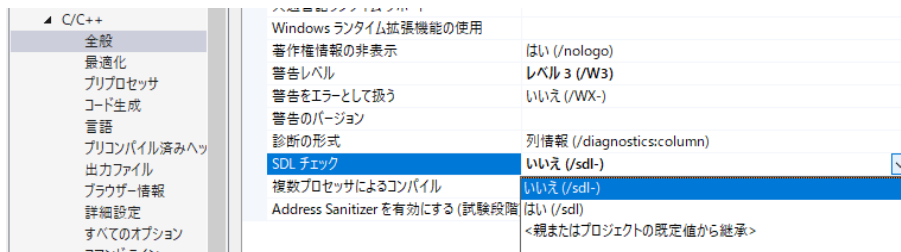


OpenGLのための準備

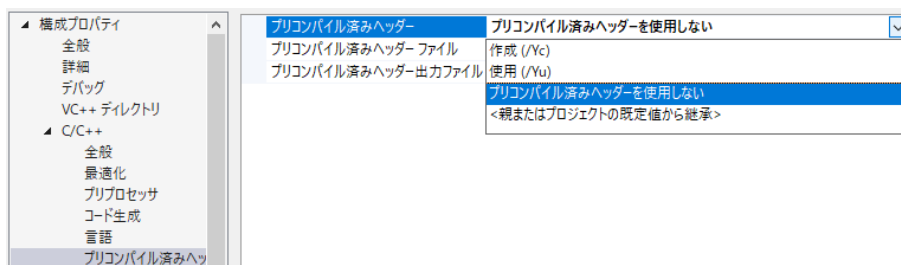
🌟 [プロジェクト]-[○○のプロパティ]をクリック



🌟 [構成プロパティ]-[C/C++]-[全般]-[SDLチェック]で[いいえ]を選ぶ



🌟 [構成プロパティ]-[C/C++]-[プリコンパイル済みヘッダー]-[プリコンパイル済みヘッダー]で[プリコンパイル済みヘッダーを使用しない]を選ぶ



サンプルプログラムをコンパイルして実行してみよう

```
#include <stdlib.h>
#include <stdio.h>
#include <gl/glut.h>

void myDisplay()
{
    glClearColor(1.0, 1.0, 0.0, 1.0);
    glClear(GL_COLOR_BUFFER_BIT);
    glColor3d(1.0, 0.0, 0.0);
    glBegin(GL_POLYGON);
    glVertex2d(-0.5, -0.5);
    glVertex2d(0.5, -0.5);
    glVertex2d(0.5, 0.5);
    glVertex2d(-0.5, 0.5);
    glEnd();
    glutSwapBuffers();
}

void myKeyboard(unsigned char key, int x, int y)
{
    if (key == 0x1B) exit(0);
}

void myInit(char* progame)
{
    glutInitDisplayMode(GLUT_RGBA | GLUT_DOUBLE);
    glutInitWindowSize(500, 400);
    glutInitWindowPosition(0, 0);
    glutCreateWindow(progame);
}

int main(int argc, char* argv[])
{
    glutInit(&argc, argv);
    myInit(argv[0]);
    glutKeyboardFunc(myKeyboard);
    glutDisplayFunc(myDisplay);
    glutMainLoop();
    return 0;
}
```

