

穴吹講義メモ 2016.12.16

XeenのC++コーディングルール

クラス名は CamelCase とする

備考

- ・キャメルケース CamelCase
- ・スネークケース snake_case

メソッド名は CamelCase とし
基本命名は DoThat とする

メンバーは一般的に使用される次の表現であればよい。
ただしクラスで統一する。

- lower_case_
- m_lower_case
- mCamelCase

変数名に一文字物は使わない。

ただしループ変数やテンポラリポイントにおいて
i, j, k, t, p, x, y, z, w
など広く使われているものは許可する

利用者に向けたコメントはヘッダー中に記述。

Doxygenコメントで。

```
/**  
 * @brief Javadoc 系ブロックコメント  
 */  
  
/// Javadoc 系コメント  
int value;  
  
int value; ///  
Javadoc 系左記説明コメント
```

```
/*!  
 * ¥brief Qt 系ブロックコメント  
 */  
  
//! Qt 系コメント  
int value;  
  
int value; //!< Qt 系左記説明コメント
```

ファイル先頭コメント

```
/**  
 * @file  
 * @brief ファイルの要約説明を書きます  
 * @author 実装者、引き継ぎ担当者を書きます  
 */
```


ヘッダーには常にインクルードガードを

```
#ifndef INCLUDE_HEADER_HPP_  
#define INCLUDE_HEADER_HPP_  
  
    ~snip~  
  
#endif
```

マジックナンバーを使わない

```
#define DEF_STOP_NUMBER 1234  
const int DEF_STOP_NUMBER = 1234; // もしくはこちら  
  
if(DEF_STOP_NUMBER == num){  
}
```

《CMの時間》

NEW
トップページ 最新情報 ギャラリー キャラクター&ボイス 遊び方 公式情報

ダウンロードしてくれよな!

AppStore, GooglePlayで
好評配信中!!!

App Store
からダウンロード

Google Play
で手に入れよう

assertとstatic_assertの話

- ・意味、目的
- ・実装例

```
#include "stdafx.h"
#include <assert.h>

enum ShotType {
    Fire,
    Valkan,
    Laser,

    COUNT
};
int shot_power_table[] = { 10, 3, 5, };
#define ELEMENTS_OF(a) (sizeof(a)/sizeof(a[0]))

int function( int a, int b )
{
    assert( a > 0 );
    return a-b;
}

int _tmain(int argc, _TCHAR* argv[])
{
    static_assert( ELEMENTS_OF(shot_power_table) == (int)ShotType::COUNT, "Data array size mismatch!" );

    int x=100;
    for ( int i=0; i < 100; ++i ) {
        x = function(x,i);
    }
    return 0;
}
```

VS デバッグテクニック

主に解説する内容

- ・ breakpoint、ステップ実行、ローカル変数、呼び出し履歴 は知っている？使っている？
- ・ 条件指定の breakpoint
- ・ 変数が書き変わった瞬間にブレーク。
- ・ ローカル変数の値を変えることで簡易的な調整作業もできるよ。