Coding Guideline Projekt „Sisyphus“

* 1. Klassennamen

Ein Klassenname beginnt mit einem Großbuchstaben. Die einzelnen Wörter im Klassennamen beginnen ebenfalls mit einem Großbuchstaben.

Bsp.:

* TestClass
* XmlTreeBuilder
  1. Variablennamen

Ein Variablenname beginnt mit einer abgekürzten Typbezeichnung (für Standardtypen sollte die ungarische Notation (die vereinfachte Variante unter <http://de.wikipedia.org/wiki/Ungarische_Notation#Systems_Hungarian>) verwendet werden) und wird fortgesetzt durch die Aufgabenbezeichnung, beginnend mit einem Großbuchstaben.

Bsp.:

* int nCounter = 0;
* uint unCounter = 0;
* XmlTree xtTree = new XmlTree();
  1. Methodennamen

Ein Methodenname beginnt mit einem Großbuchstaben.

* 1. Instanzvariablen

Instanzvariaben (Membervariablen) sind schlicht Variablen, die mit einem Unterstrich „\_“ beginnen.

Bsp.:

* string \_sFilename;
* XmlTree \_xtTree;
  1. Properties

Properties lauten wie die Instanzvariablen. Es werden lediglich der Unterstrich und der Typbezeichner weggelassen.

Bsp.:

* String \_sFilename; 🡪 Property: string Filename {get{return \_sFilename;}}
  1. Static – Klassenvariablen

Static Klassenvariablen verhalten sich wie Instanzvariablen, es wird lediglich noch ein Unterstrich vorangestellt. (Wahrscheinlich nur beim Singleton – Pattern benötigt.)

Bsp.:

* static string \_\_sFilename;
  1. Zugriff auf Instanzvariablen

Beim Zugriff auf Instanzvariablen ist this nicht voranzustellen.

Bsp.:

* \_sFilename = „Test.xml“;
  1. Methodenzugriff

Beim Methodenzugriff ist this voranzustellen.

Bsp.:

* this.PrivateInit();