Ну посмотри как все клево – если константная сущность, то для нее важно имя и значение, возможно задание типа. Ну а если переменная (mutable) сущность, то будь добр поставить слово var. Т.е. если пользоваться инициализацией c конструктором то сначала пишешь без var … Как удобно … Только надо принять правила определения констант в английском языке с is … Ну ничего полезного слово const не дает … Большинство локальных объектов константы … Т.е. понятие константы – это не то понятие что в Паскале или Си … Это – сущность, которой в ее области видимости делается только инициализация и более нельзя сделать присваивания …

Т.е. каждый раз при входе в рутину ее локальные константы инициализируются – т.е. это не локальные статики …

Отсюда кстати встает вопрос – если ты так красиво предложил делать use для классов может сделать use для рутин – это как раз будут локальные статики …

var variable: Type [.ctorName] [(<expressions>)]

var variable:[Type] = <expression>

var variable: ?Type

constant[: Type] is <expression>

constant: Type[.ctorName] [([<expressions>])]

<EntityDeclaration> ::= <VariableDeclaration>| <ConstantDeclaration>

<VariableDeclaration> ::= ‘var’ <Identifier> “:” <NonInitializedVar>|

<ConstructorInitializedVar>|<ExpressionInitializedVar>

<NonInitializedVar> ::= “?” <Type>

<ConstructorInitializedVar> ::= <Type> [“.” <Identifier>] [(<ExpressionsList>)]

<ExpressionsList> ::= <Expression> {“,” <Expression>}

<ExpressionInitializedVar> ::= [<Type>] “=”<Expression>

<ConstantDeclaration> ::= <Identifier> <ConstructorInitializedConst>|<ExpressionInitializedConst>

<ConstructorInitializedConst> ::= <Type> [“.” <Identifier>] [(<ExpressionsList>)]

<ExpressionInitializedConst> ::= [: <Type>] “is”<Expression>