ステートメントを構成する「式ｏｒ式じゃないもの」
の構成要素（変数、定数、リテラル値、キーワード（＝全・命令語句））

ExcelVBA「初級者」の僕が、ヘルプを読んでやれるところまで整理した、
ステートメントを構成する「式 or 式じゃないもの」の 構成要素（変数、定数、リテラル値、キーワード（＝全ての・命令語句））

（※「プロシージャ」は、「ステートメント」が複数集まってできている、開始と終了の区切りが決定づけられたものです。なので下記のものたちは、そのまま、「プロシージャを構成する要素」＝「学習対象」、にもなります。）

「式」か？否か？			Exc 中（というか、「各種の ” ライブラリ ” 」の中）に既に組み込まれているモノたち			ユーザー側が自作するモノたち		
「式」 **「式」とは、** **「値と型が確定したもの** **（＝値と型の ″ 評価 ″ が完了したもの）」** **のことを言います。** **1つの語句か、複数の語句でできています 。** 基本、単語1語であっても、ウォッチウィンドウで調べたときに、「エラーメッセージが表示されておらず」「値と型」が表示されていれば（＝特に「型」が表示されていれば）、「値と型が確定した」という意味になり、「式」と呼びます（呼べます）。 **※「値」とは、文字系の一般データ（数値、文字列、日付、論理値など）か、または、オブジェクトの、いずれかのことを指しています。** **その2種類しかありません。** **なので、裏を返せば、「式」も、** **一般データかオブジェクトかのいずれかしか」、** **返しません。あるいは、評価されません。** また、「式」には、 ● 数値式、● 文字列式、 ● 日付式、● ブール式、● オブジェクト式、 などがあります。 繰り返しになりますが、、「式」は、「変数や定数単独、リテラル単独」でできているか、あるいは、プラス、それらと、「演算子、関数、ステートメント名、プロパティ、メソッド、等々」といった、「キーワード（＝すべての命令単語）」との「組み合わせ」で、できています。 そして、一般データかオブジェクトのいずれかを返します。（ or 評価されます。）	※長い「式」の中には、「複数の短い式」が混在・点在することもあります。 **また、「式」は何らかの命令単語の「引数」になることも少なくありません。** **逆に言うと、「引数」には「式」の形になったものしか使えません。もう少し細かく言うと、基本、「引数」には、一般データのリテラル値か、オブジェクト式のいずれか、もしくはそれらを代入した変数や配列、定数、（全部「式」のかたちになっています）のどれかしか使えません（例外があるかもですが）。** **※「エラー解決」は、「式」単位で、「式のくくりで 」、ウォッチウィンドウで各種チェックをすると、早く解決できることが少なくありません。** **逆に言うと、「式」という単位の考え方が自分の中に根付いていないと、エラー解決や「コードを書く技術の向上」が遠のいてしまいます。**	⇨ 「式」を構成する要素 ⇨	操作対象		基本、「なし」。 強いて言うと「Me」キーワードくらい？ 例外？番外？：「オブジェクトモデル」 （＝「単一のオブジェクト」と「コレクションオブジェクト」） ※コレは、ここ、「ステートメントの直接の構成要素」に挙げてはいけない気がしますが、ただ、コードの中では、関数やプロパティやメソッド、引数、などを使って書いた「オブジェクト式」が、「生のオブジェクト」に相当します。 ウラで、ライブラリ単位で、各ライブラリの中にそれぞれの「オブジェクトモデル」が事前に組み込まれているので、それのおかげで、**もうひとつの操作対象の、「オブジェクト式」「オブジェクト変数」**が作れます。	作対象 値と型が「評 確定されるも	ここ、パープルの ろだけは、「式を構 する要素」ではあ れども、でも、「キ ワード（＝命令）じ あないモノ」たち す。 **逆に、ほかのグリ ンとイエローの部 は全部、「キーワ ド」となります。**	リテラル値（文字系の「一般データ」にしかない「生の値」） 変数01・一般データの単一変数 変数02・要素が一般データの配列変数 変数03・オブジェクト変数（オブジェクト式＝オブジェクトを代入するためのもの） 変数04・要素がオブジェクトの配列変数？ 変数05・要素がオブジェクトや一般データごちゃまぜの配列？ など。 **※リテラル値や変数が「たったの1語」だけでも、「値と型が確定してさえいれば」、それは「式」となります。 よって「短い式」とは、そのような「1語」の場合もあります。もちろん、複数語句で1つの式、というケースも多いです。**
			操作対象を動かすための「命令単語（＝キーワード）」のうち、 値を返すタイプのモノ	主に一般データを かすことがほとん のキーワード	演算子 Excel専用のVBA関数 Office共通のVBA関数 WorksheetFunctionオブジェクトのメソッド Excel専用の組込定数 Office専用の組込定数 など。	作対象を動かす の「自作の命 単語」のうち、 を返すタイプの 自作キーワー	一般データもオブ クトも動かす自作 キーワード	自作関数。（＝ユーザーのオリジナルの関数） （＝汎用呼び出し可能な自作のプロシージャ。そのうち、値を返すタイプのみ。） （※「値」＝文字系の一般データ、または、オブジェクトのいずれかのこと。） **※この部分、特に、「オブジェクトを返す色んなタイプの自作関数」を作れることが、VBAの「真のスタートライン」なんですが、そのことを詳しく書いた本やサイトは、日本には1つもありません。** **なので、多くの人が、「真のスタートライン」にすら立てず、大抵は初級の書籍かテキストをやった直後で挫折します。（お金を払って習っていても）**
				主にオブジェクトを かすキーワード	プロパティ（基本、値を返さないタイプのモノはありません。） メソッド（値を返すことが可能なタイプのみ） 「オブジェクトがらみ」の、「列挙（Enum）」に含まれる組込定数		主にオブジェクトを かす自作キーワ	自作のプロパティ（値を返さないタイプのモノはありません。） 自作のメソッド（値を返すことが可能なタイプのみ）
				上記兼用	引数名（名前付き引数で使う）		上記兼用	自作の定数 自作の引数名（名前付き引数で使う）
					など。 ※上記のモノたちの中で、例外的に、「値を返さないモノ」があったとしても、 恐らく、その数はほんの少しですので、学習しはじめの最初のウチは、 上記のモノたちは、基本、「ほぼ値を返す」と思っていていいです。			など。 ※上記のモノたちは、「値を返したいからこそ」「自作する」ので、 「値が返されないタイプのモノ」は「無い」、とお考え下さい。
「式じゃないもの」	※長い「式じゃないモノ」の中には、「複数の短い式」が混在・点在することもあります。 特に、「引数」として混在・点在しますし、ほかにもあるかもです。 （「引数」には、リテラル値やオブジェクト式、また、それらを代入した「変数（配列）」「定数」しか使えませんが、それらは全部、「式」のかたちをなしています。）	「式じゃないもの」を構成する要素	操作対象を動かすための「命令単語（＝キーワード）」のうち、 値を返さないタイプのモノ	一般データ、オブ クト、その他、を動 すキーワード	データ型（オブジェクト名含む） ※ 変数宣言のときに使います。	作対象を動かす の「命令単語 キーワード）」 ち、 を返さないタイプ モノ 自作キーワー	一般データやオブ ジェクトを動かす ワード	自作のデータ型（自作のオブジェクト名含む） ※ 変数宣言のときに使います。
				主にオブジェクトを かすキーワード	メソッド（値を何も返さないタイプのみ）			自作の関数 （＝汎用呼び出し可能な自作のプロシージャのうち、値を何も返さないタイプのみ） など。
				一般データ、オブ クト、その他、を動 すキーワード	ステートメント名 ※「ステートメント名」とは、 If、Then、Else、For、Next、Do、Loop、On Error、Sub、Function、End、Call、Open、Wirite、Beep、などの、ステートメント用のキーワードたちのことと思ってください。ヘルプではこの「ステートメント名」という用語が使われています。 フロー制御系、定義系、テキストファイル制御系、フォルダ操作系、ハードウェア操作系、など、各種あります。 基本、ステートメント用のキーワードは、何も返しません。・・・と思います。たぶん。（例外もあるかもですが）			
					引数名（名前付き引数で使う） イベント名 など。			自作の引数名（名前付き引数で使う） 自作のイベント名

←全部、ヘルプを読むと、こんな感じのことが書いてあります。

自分は学者じゃないし、初心者の方の「目線・学習しやすさ・目的」のために、そこを考えて、参考になるようにあえて意訳(？)もしてあるので、１００％正しいわけではないと思いますが、ただ、プロの書籍の著者にしても、ここはちゃんと明確には整理・説明が「できない・無理」だと思います。**現状、彼らは、「ステートメントやオブジェクト式の意味」自体すら、まともに説明できていないので。20年も経ったのに、未だに業界内での**

日本のVBAの市販書籍やサイトには、「式」の概念、オブジェクトモデルの詳細（特にコレクションオブジェクトの階層構造内の位置や詳細）、ウォッチウィンドウやオブジェクトブラウザの意義、「真のスタートラインとしてのオブジェクトを返す自作関数の作成」、「ＳＱＬやその詳細」、のことなどを網羅的に１か所で教えてくれるモノは「１つもありません」（方向性としてすらでも）。
結果として、上記の表の1/2以下の項目の中の、さらに1/3程度のことしか教えてくれない感じです。
それでは、「挫折するべく挫折させられて、結局、" わざと "、核心を教えずカモっているのでは？」と疑われても、反論できないような気がします。

※「キーワード」とは？

「キーワード」とは・・・、
(a) 基本、VBEにての、F1キーの押下でヘルプページが出てくるモノ（＝F1キー押下に反応するモノ）、
(b) F1キーの押下に反応しなくても、ステートメントの中に実際に書く単語で、かつ、ヘルプにも載っているモノ、
(c) （F1キー押下には反応しないけど）以上のキーワードと性格が同じ、自作の命令や設定値、データ型名、イベント名、など、
・・・のこと、と捉えてよい（＝キーワードに含めてよい）」と思います。 **つまり「キーワード＝変数やリテラル値以外の、すべての命令や設定値ととらえてよい」、という感じです。**

ヘルプを読むと、「何が ”キーワード ” なのか？」や、「どこからどこまでがキーワードなのか？」が分かりにくいのですが、逆に、ヘルプやウォッチウィンドウやオブジェクトブラウザなどを読みやすくするためには、
「あえて」で、**「キーワードとは、VBAの中において、ステートメント内に実際に記述する、 ” 命令単語 ” ・あるいは ” 設定値単語 ” という意味合いをもつ、すべての語句・単語」**と考えてよいと思います。
（※また、「命令単語」とは、「＝何かを動かせる命令単語のこと＝何かを加工できる・処理できる・ゲットできるetc、の、そういう命令単語のこと」、というイメージで捉えて頂いて大丈夫だと思います。）

つまり、VBAの場合は、世間一般常識的な、「キーワード＝重要語句」という意味ではなくて、**「キーワード＝VBAで使える、すべての命令単語や設定値単語」** という感じになります。 **特に「ヘルプ」の中では。**
（そう明記はされていませんが、でも、そういうニュアンスで書かれていますし、実際、そう考えたほうが、「 迷いが ・大幅に減り 」、ヘルプが読みやすくなります。）

なので**そこには、 演算子、全ての関数（Excel固有、Office共通、API、その他、等々、すべて）、オブジェクト名、プロパティ、メソッド、ステートメント名（各種ステートメント用の単語）、各種組み込み定数、引数名、イベント名、**などが含まれます。（ムリヤリにでもそう含めて考えていいです。）
なので、更に **「意訳」して、自作の関数、オブジェクト名、プロパティ、メソッド、プロシージャ（Callキーワードやドットで呼び出せるモノ）、定数、引数名、なども、「自作キーワード」として位置づけてもよい**と思います。

将来、クラスモジュールをバンバン作るようになって、「そのような考え方が間違いだと思えるようになるまで」、上記のような考え方で通して問題ありません。
もちろん、クラスモジュールをバンバン作るようになっても、「そのような考え方で大体はイケル！とも思いますので」、もしそうなら、なおさら、それで通して問題ありません。もちろん、「それは違う」と明確にできるなら、そう直せばさらに問題なくなります。
このあたりのことは、「VBAのレジェンドですら」、「かなりいい加減な説明に終始しています」。例えば「VBAエキスパート試験」の古いほうのテキストなんて、「超・いい加減」です。ここでのこ説明が「良い」とは言いませんが、少なくともそれと比較しても「劣ってはいない」と思いますので、以上のような考え方・捉え方で問題ありません。
Excelが出て、これまでの20年以上、「この部分について」 自著や自サイトで「超」いい加減な説明しかしてこなかった彼らが、今更、何か言えた義理ではないですし、ここで書いている「意訳的なこと ・こじつけ」、「ダサさ」、「説明ミス」、など、彼らのいい加減さに比べたら全然「可愛いもの」です。

あと、とっても大事なことですが、「キーワード（＝すべての命令単語）」は・・・・、
つまり、すべての、演算子、全ての関数（Excel固有、Office共通、API、その他、等々、すべて）、プロパティ、メソッド、ステートメント名（各種ステートメントを用の単語）、各種組み込み定数、引数名、イベント名、自作のそれら、などは、どれもこれも、『 全部をひっくるめて考えた場合 』、
そもそも、大別をすると、
①文字系の一般データを返す動きしかしないタイプのモノ、 ②何らかの動き（処理）をしつつ、同時に、文字系の一般データを返す動きもするタイプのモノ、
③オブジェクトを返す動きしかしないタイプのモノ、 ④何らかの動き（処理）をしつつ、同時に、オブジェクトを返す動きもするタイプのモノ、
⑤何も返さずに、何らかの動きだけをするタイプのモノ、
の、この5つのタイプの「どれか」に「あてはまります」。（引数や既定値が「ある・なし」、あるいは、その他の性格も含めて。）
それぞれ「動かす対象となる ” モノ ”や、使う ” 場面 ” 、 ” 書き方（構文の構造） ” などが異なる」ので、そのため、それぞれの呼び名が異なるだけで、本質的な「動き」の部分を見ると、ほぼ、この5つのタイプしかなく、このうちの「どれか」に「あてはまります」。
この考え方、くくり方、は非常に重要です。ヘルプやオブジェクトブラウザやウォッチウィンドウなどが読みやすくなるためです。

なお、「何らかの動き」とは、計算・条件別集計したり、比較したり、変換、ソート（並べ替え）、結合や分離、転記、切り取り、外部データ接続、吸込み、メッセージ表示、開く/閉じる、コピー、削除、検索、置換、フィルタ（抽出、絞込み）、グラフ作成する、・・・などのことを指します、

というわけで、上記の表の「キーワード」の部分（パープルの部分を除いた、グリーンとイエローの部分）に限ってですが、そのうち、『 式じゃないもの （イエロー部分）』には、⑤だけがあてはまります。 一方、『「式」を構成する要素（グリーン部分）』の方には、①～④があてはまります。