Current Skill Variables and identifiers

What is a variable?

- Variables are used to associate a name with a value, which can change during the algorithm .
- In other words, a variable is a storage location for a single type of data, which can be used to store a value and used to retrieve a value.
- Variable declaration :

Variable declaration can be done in two ways :

1. Name_of_variable : type
2. Name_of_variable, name_of_other_varibale : type

```
age : integer // declare a variable with the name “age” of type “integer”
a,b : boolean // declare two variables with the name “a” and “b” of type “boolean”
```

To declare a constant (A variable that will not change its value throughout the algorithm) we use the keyword CONST instead of VAR

Identifier

- Variable name or identifier has precise rules for creation.
- Luckily, the same rules for identifiers apply to anything you are free to name, including variables, program name, procedures, functions and structures (we will see all of them in this course).
- There are only three rules to remember for legal identifiers:

1. The name must begin with a letter (uppercase or lowercase) or the underscore symbol ’_’.
2. Subsequent characters may also be numbers.
3. You cannot use the same name as an algorithm reserved word, such as var, begin ,end...

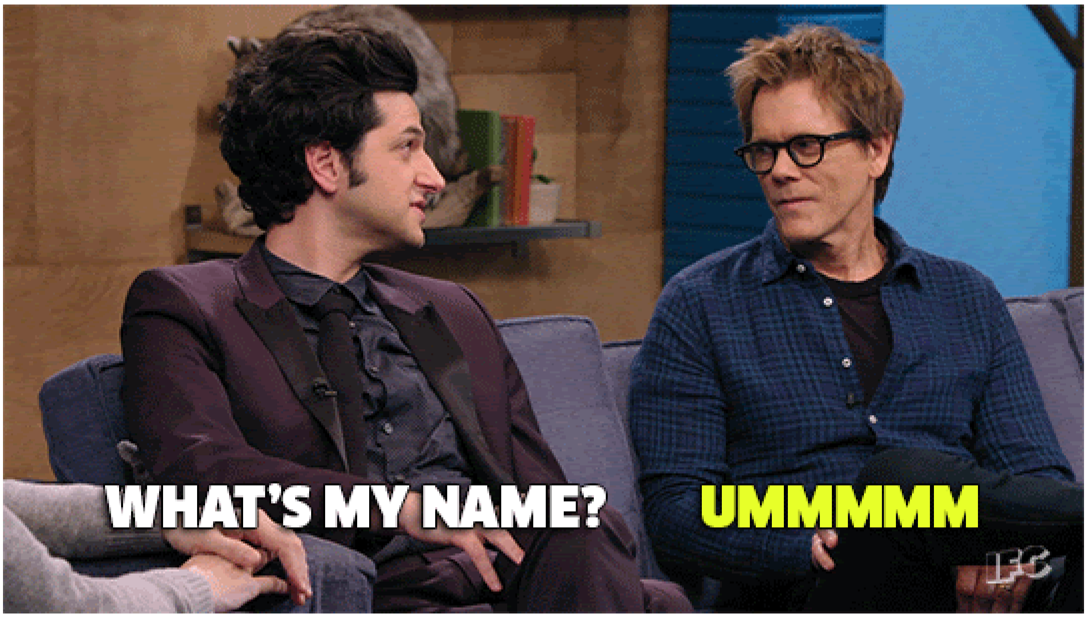
The following examples are legal:

- okidentifier
- OK2Identifier
- _alsoOK1d3ntifi3r

- __SStillOkbutKnotsonice

These examples are not legal:

- 3DPointClass // identifiers cannot begin with a number
- hollywood@vine // @ is not a letter, digit or _
- *_coffee // * is not a letter, digit or _
- var // var is a reserved word



Assignment :

Assignment a value to a variable can be done with four way :

- Name_of_variable := value
- Name_of_variable := name_of_other_varibale
- Name_of_variable := expression
- Name_of_variable := name_of_function

```
age := 28     // assign (i.e. store) the value 28 in the variable “age”

b := false    // assign (i.e. store) the value false in the variable “b”

a := b        // assign (i.e. store) the value of “b” (false) in the variable “a”
```

‹ Previous

next ›