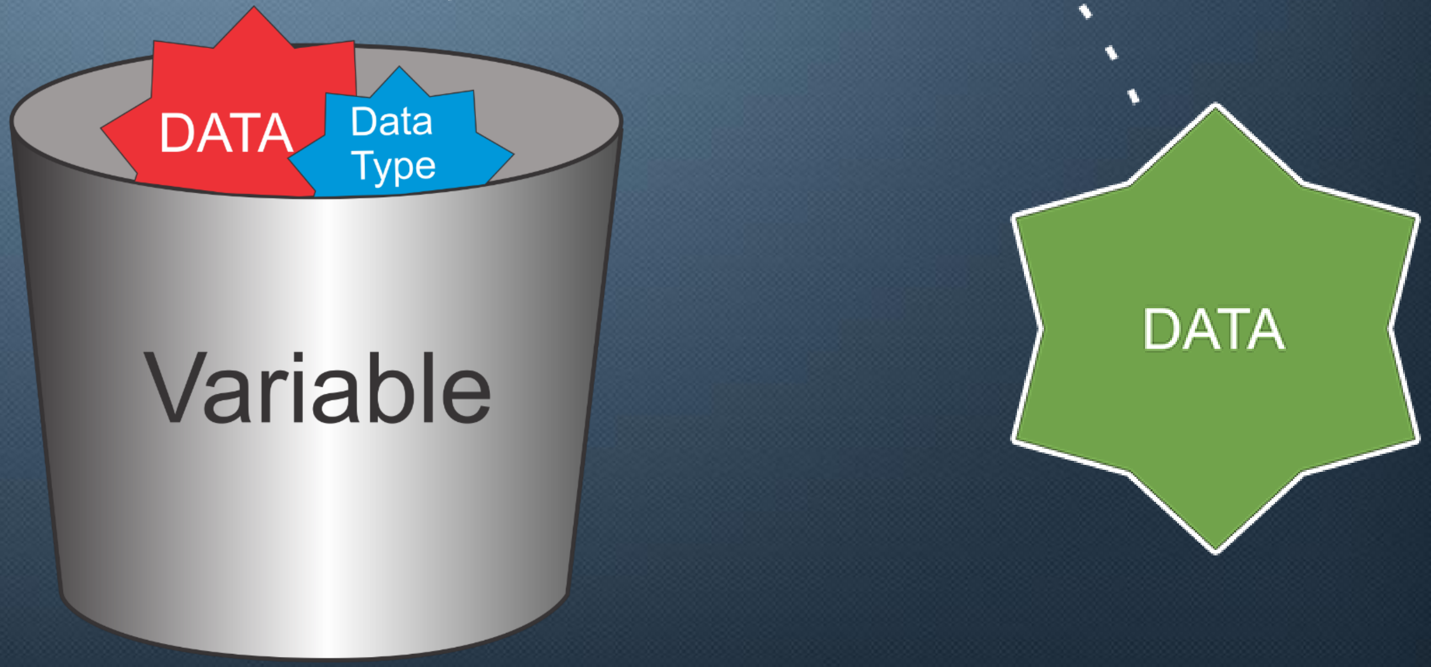
Variables & Assignments

Variables

Variables are placeholders for values. Variables are used to avoid re-computing of values and to give names that help the reader to understand code.



Properties of Variables

❑ Variable names must start with an underscore or letter, which can be followed by any number of letters, digits, or underscores.

❑ Reserved words can not be used as variable names.

❑ Variables cannot start with a number.

Legal	Illegal
_name	@name
true	True
Age22	22age

Variables are case sensitive i.e. 'name' and 'Name' are two different variables

Dynamic Typing

Dynamic typing means that there is no need to declare variables before using them

Variable creation

A variable is created when your code first assigns it a value

Variable types

A variable never has any type information associated with it. The notion of type lives with objects, not names.

Variable use

- When a variable appears in an expression, it is immediately replaced with the object that it currently refers to.
- All variables must be assigned before they can be used.

Hands-On

Assignment

- ***Variables*** are entries in a system table, with spaces for links to objects. Variables always link to objects and never to other variables, but larger objects may link to other objects.

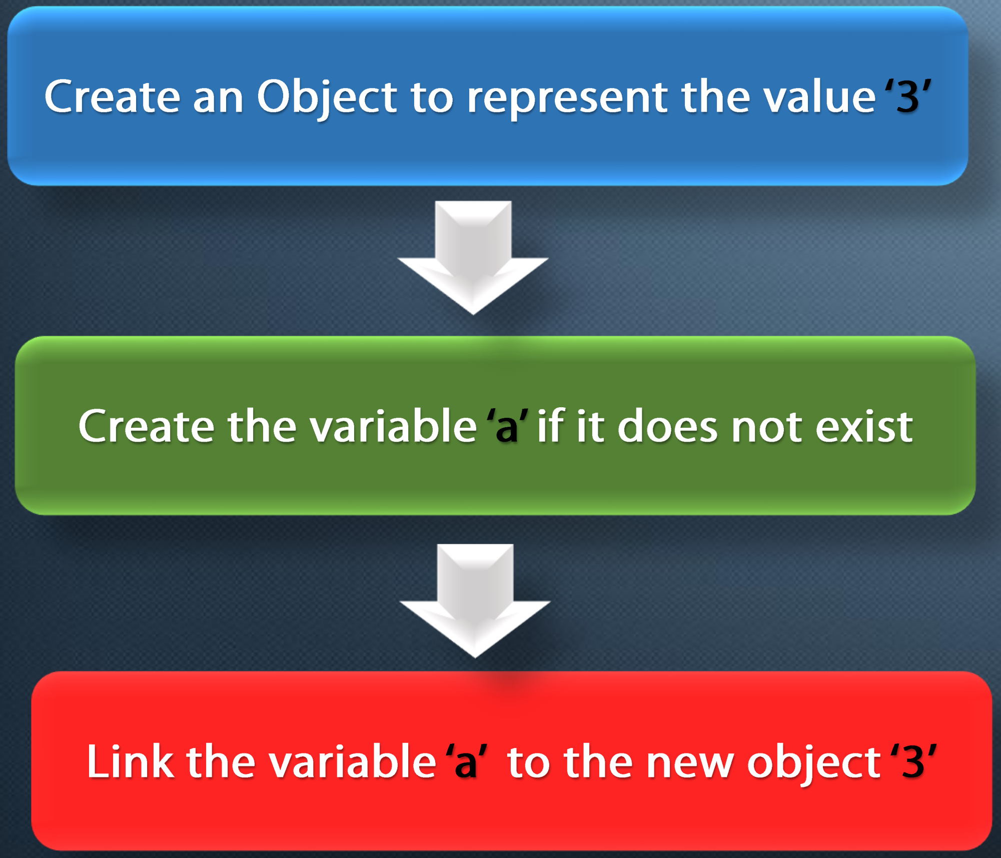
- ***Objects*** are pieces of allocated memory, with enough space to represent the values for which they stand.

- ***Reference*** is a kind of association, implemented as a pointer in memory

Assigning a name to an object

3 steps will be performed to carry out this request.

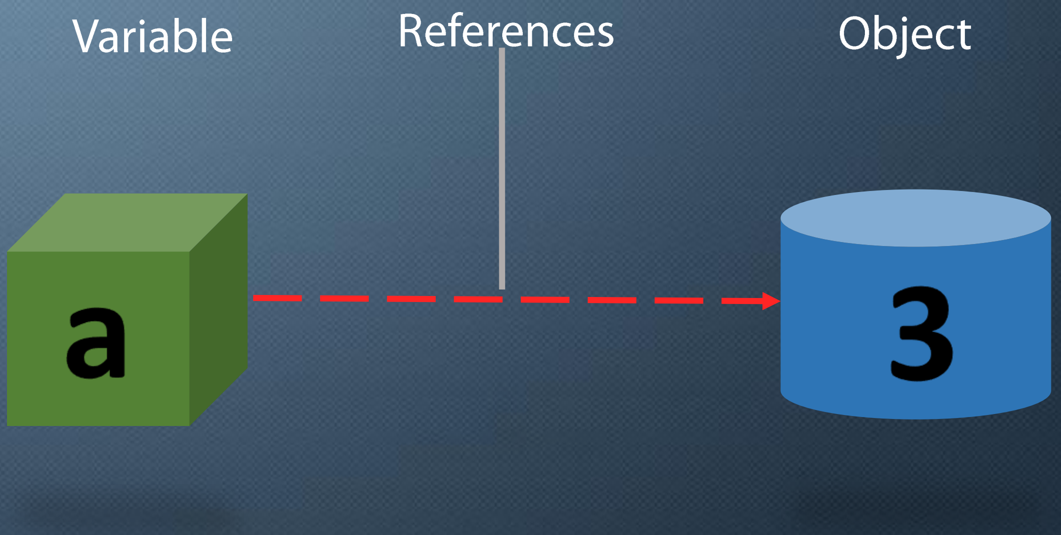
Suppose we want to assign a variable 'a' to value '3':



Create an Object to represent the value '3'

Create the variable 'a' if it does not exist

Link the variable 'a' to the new object '3'



Types live with objects, not variables

```
a=5       # it is an integer
a='sue'   # it is a string now
a=3.72    # now it's a floating point
```

In the above example, we haven't actually changed the type of the ***variable*** 'a'; we have simply made the variable reference to a different type of object.

Objects, on the other hand know what type they are - each object contains a header field that tags the object with its type.

Objects are garbage collected

Whenever a name is assigned to a new object, the space held by the prior object is reclaimed if it is not referenced by any other name or object. This automatic reclamation of objects' space is known as garbage collection.

Hands-On