

~~apt method anything is object in py~~  
 after access (attr - read + attr - write)

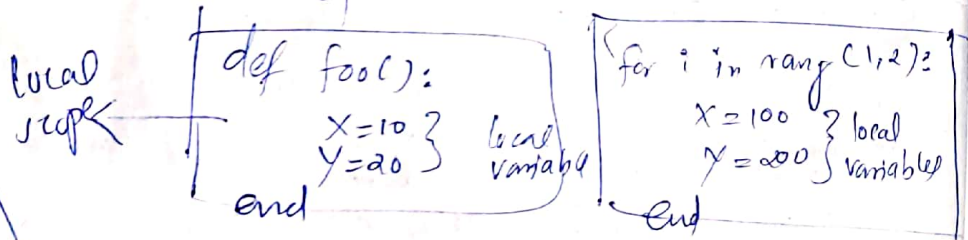
combination of attr - read + attr - write

## Types of variables

- 1) local variable
- 2) class variable
- 3) instance variable
- 4) Global variable
- 5) Constants

### 1) Local variable

- declared in ~~declared in~~ method or loop or block
- starts with lowercase letter or underscore [a-z]\_
- A variable that is declared inside function or loop (block) of a computer program.
- Accessible only within the function it is declared



Local var have its scope within a block of its initialization. Once the code block completes its execution, no scope.

⇒ if i try to access x & y from outside the loop or function (method) it prints undefined local variable error  
 Local variables are not available outside the function.

```
def foo():
```

```

a=1
b=2
print a+b
end
  
```

⇒ Access a & b from outside function

```

print a
print b
foo()
  
```

⇒ ~~Created~~ local variable is created when the function starts executing and is destroyed when the execution is complete.

⇒ more reliable & secure since the value cannot be changed by other functions.

⇒ Accessed within a block or function.

⇒ default value of local variable is garbage.

## 2) class variable

- starts with @@.
- declared in a class.
- They need to be initialized before use otherwise prints error.
- A class variable belongs to the whole class and can be accessible from anywhere inside the class.
- A class variable is shared by all the descendants of the class.
- class variable must be initialized before method definition

```

class class_name:
    @@class variable
    def method_name:
        {block}
    end
end
  
```

you can't just put like this

~~Class class\_name~~  
 def method\_name: @@ class variable {block} end

Initialize before function creation

## Class variable

imp ⇒ if the value is changed at one instance, it will be changed at every instance.

Eg of this statement

```
class Leia
  @@ boyfriend = "Hansolo"
  def self.boyfriend
    @@ boyfriend
  end
  def self.lando_fier_hu_smooth_move
    @@ boyfriend = "Lando"
  end
end
```

puts Leia.boyfriend ⇒ Hansolo

puts Leia.lando\_fier\_hu\_smooth\_move ⇒ Lando

puts Leia.boyfriend ⇒ Lando

bcz if the value is changed at one instance, it will be changed at every instance.

⇒ An uninitialized class variable will result in an error.

```
class Student
  @@ var = 0
  def fine
    @@ var += 1
    end
  puts @@ var → fine
  puts a → error
end
```

## 3) Instance variable

- Starts with @ sign.
- It belongs to one instance of the class and can be accessed from any instance of the class within a method.
- They only have limited access to a particular instance of class.
- They don't need to be initialized.
- An uninitialized instance variable will have nil value.

## 4) Global variable

- Start with \$.

In program

> \$ default\_country = "India"

⇒ "India"

> \$ default\_country = "England"

⇒ "England"

puts \$ default\_country = England

## 5) Constants

ABC = 1

Variable never change value  
variable which has constant value

not preferred