

**Lecture**

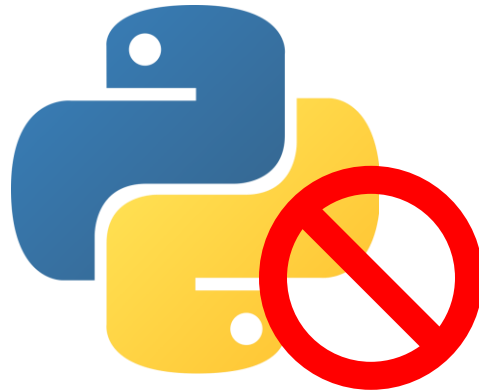
**Properties**





## Properties

# Getters + Setters





Properties



# Attributes + Getters + Setters



## Properties

```
class Dog:
    def __init__(self, age):
        self.age = age
```



## Properties

`my_obj.age`

`my_obj.age = <value>`

`my_obj.age`

```
class Dog:
    def __init__(self, age):
        self.age = age
```

`my_obj.age = <value>`

`my_obj.age`

`my_obj.age = <value>`





## Properties

**Validate  
the age**



Properties



# Protected + Getters + Setters



## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        return self._age

    def set_age(self, age):
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")
```

my\_obj.age

my\_obj.age = <value>

my\_obj.age

my\_obj.age = <value>

my\_obj.age

my\_obj.age = <value>





## Properties


```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        return self._age


    def set_age(self, age):
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")
```



my\_obj.age



my\_obj.age = <value>



my\_obj.age



my\_obj.age = <value>



my\_obj.age



my\_obj.age = <value>



## Properties


```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        return self._age

    def set_age(self, age):
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")
```



my\_obj.age



my\_obj.age = <value>



my\_obj.age



my\_obj.age = <value>



my\_obj.age



my\_obj.age = <value>



## Properties


```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        return self._age

    def set_age(self, age):
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")
```



my\_obj.age



my\_obj.age = <value>




my\_obj.age



my\_obj.age = <value>



my\_obj.age



my\_obj.age = <value>



## Properties





## Properties

# Properties



## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```



## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```



## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```





## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```



## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```



## Properties

```
age = property(get_age, set_age)
```

**Name  
of the  
property**





## Properties

```
age = property(get_age, set_age)
```

**Built-in  
Function**



## Properties

```
age = property(get_age, set_age)
```

The diagram consists of a blue square with an orange border containing the word 'Getter'. An orange arrow points upwards from the top center of this square to the 'get\_age' parameter in the code line above.

Getter



## Properties

```
age = property(get_age, set_age)
```



Setter



## Properties

```
age = property(get_age, set_age)
```



## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```





## Properties

```
class Dog:
    def __init__(self, age):
        self._age = age

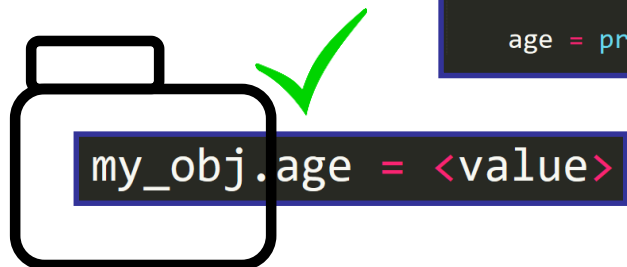
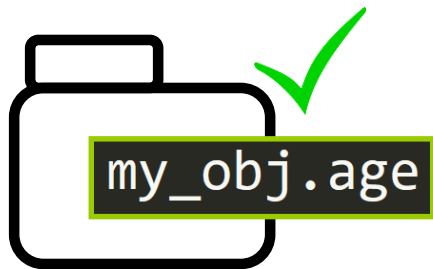
    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

age = property(get_age, set_age)
```



## Properties

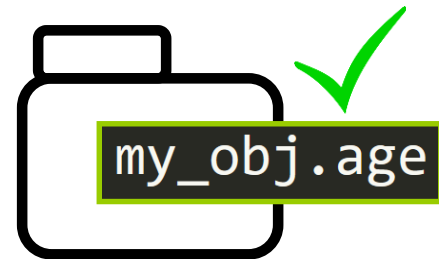
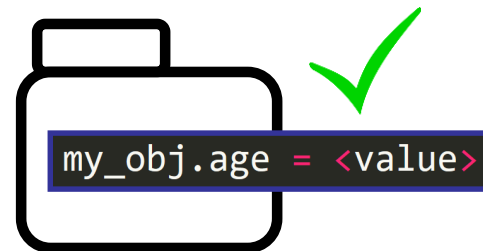
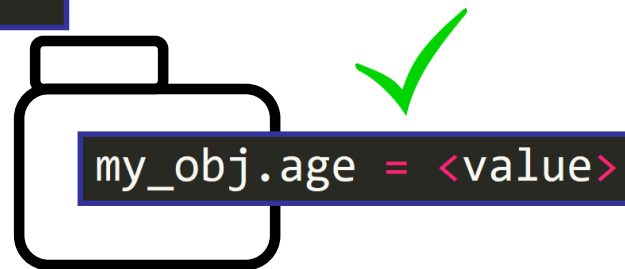
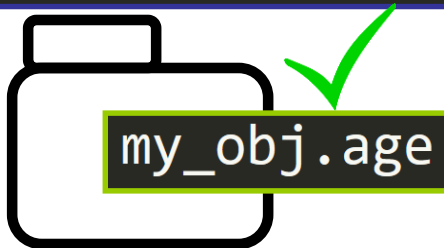


```
class Dog:
    def __init__(self, age):
        self._age = age

    def get_age(self):
        print("Running getter")
        return self._age

    def set_age(self, age):
        print("Running setter")
        if isinstance(age, int) and 0 < age < 30:
            self._age = age
        else:
            print("Please enter a valid age")

    age = property(get_age, set_age)
```





## Properties

```
my_obj.age
```



```
def get_age(self):
```



## Properties

```
my_obj.age = <value>
```



```
def set_age(self, age):
```



## Properties

```
my_obj.age = <value>
```



```
def set_age(self, age):
```



## Properties

```
age = property(get_age, set_age)
```



## Properties

# @property





## Now... An Example

