

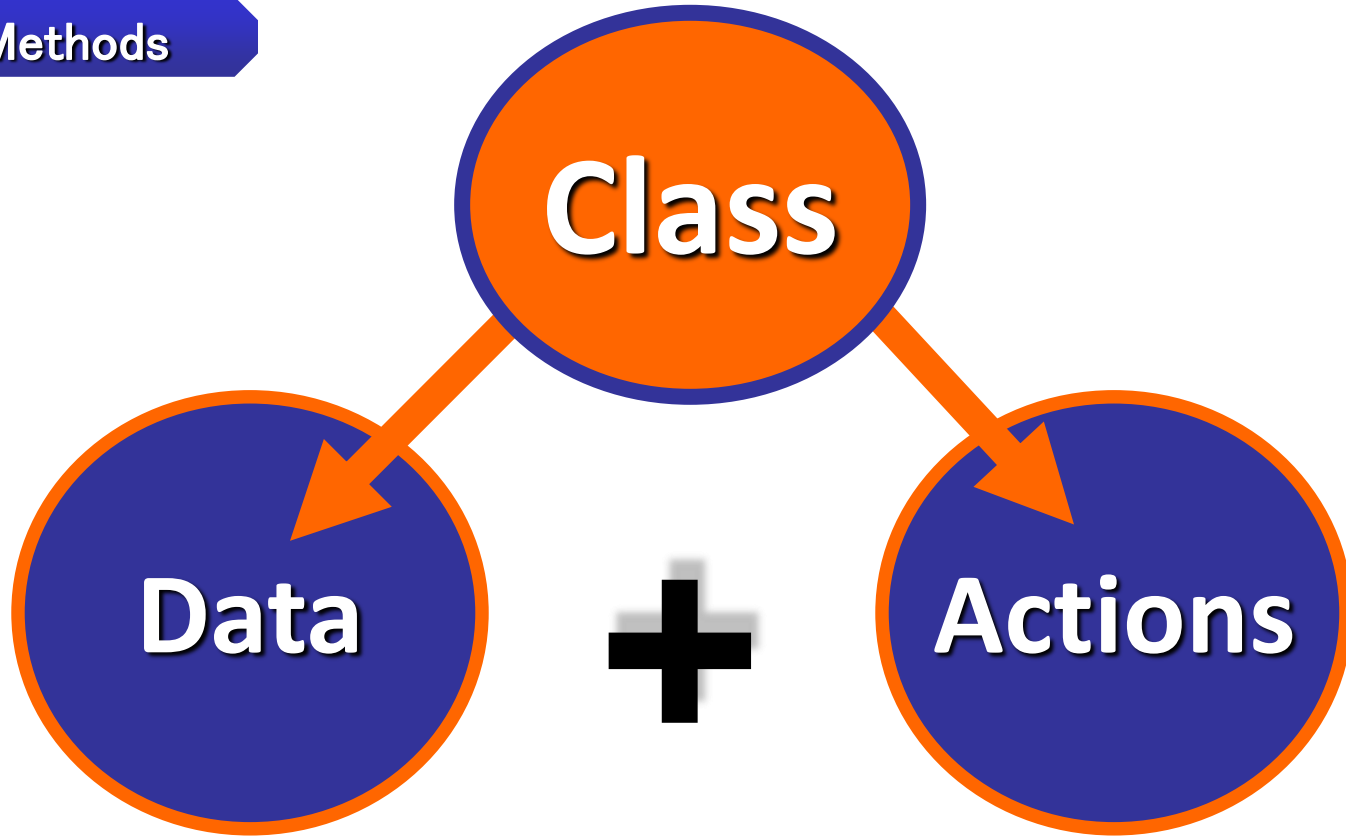
Lecture

Methods



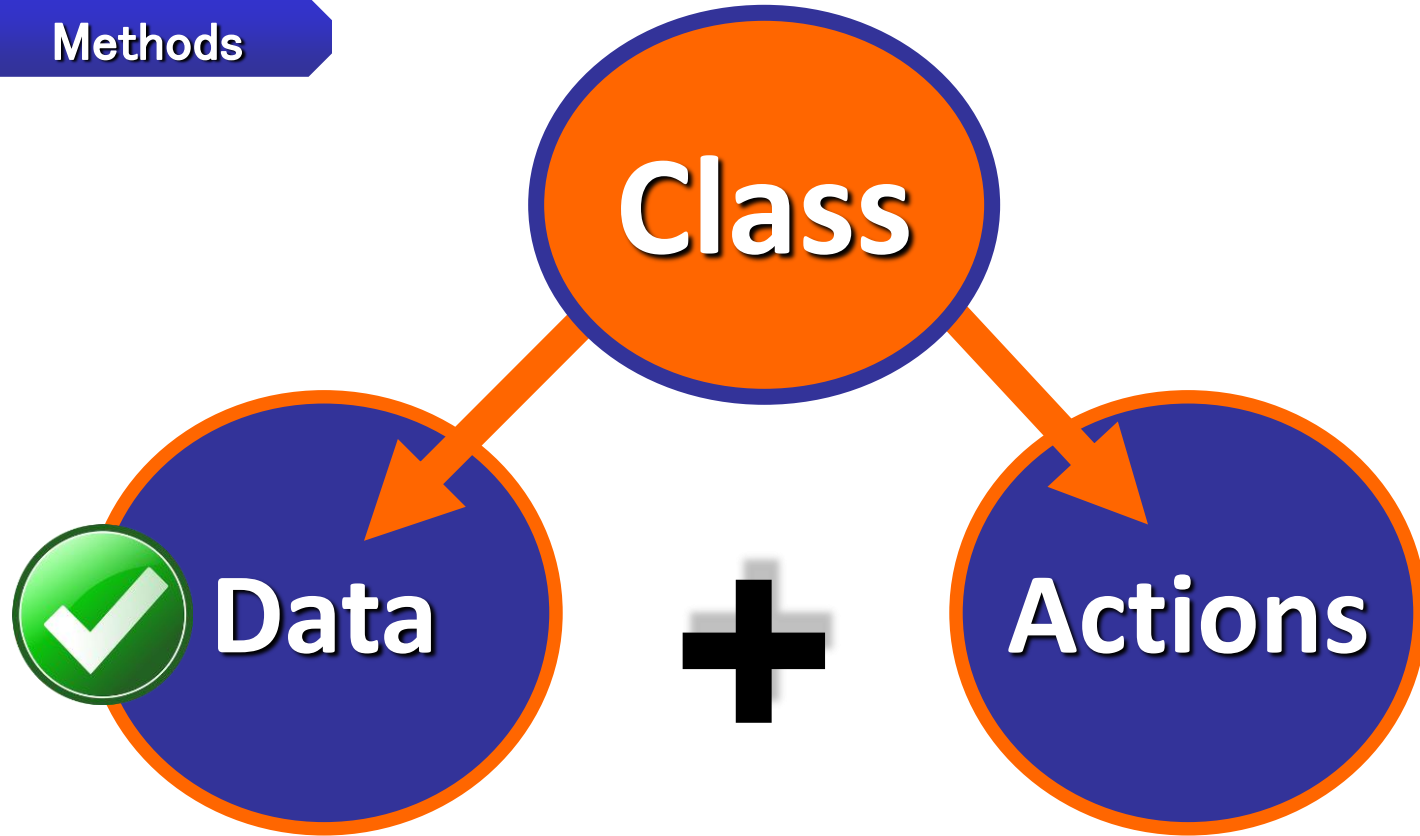


Methods



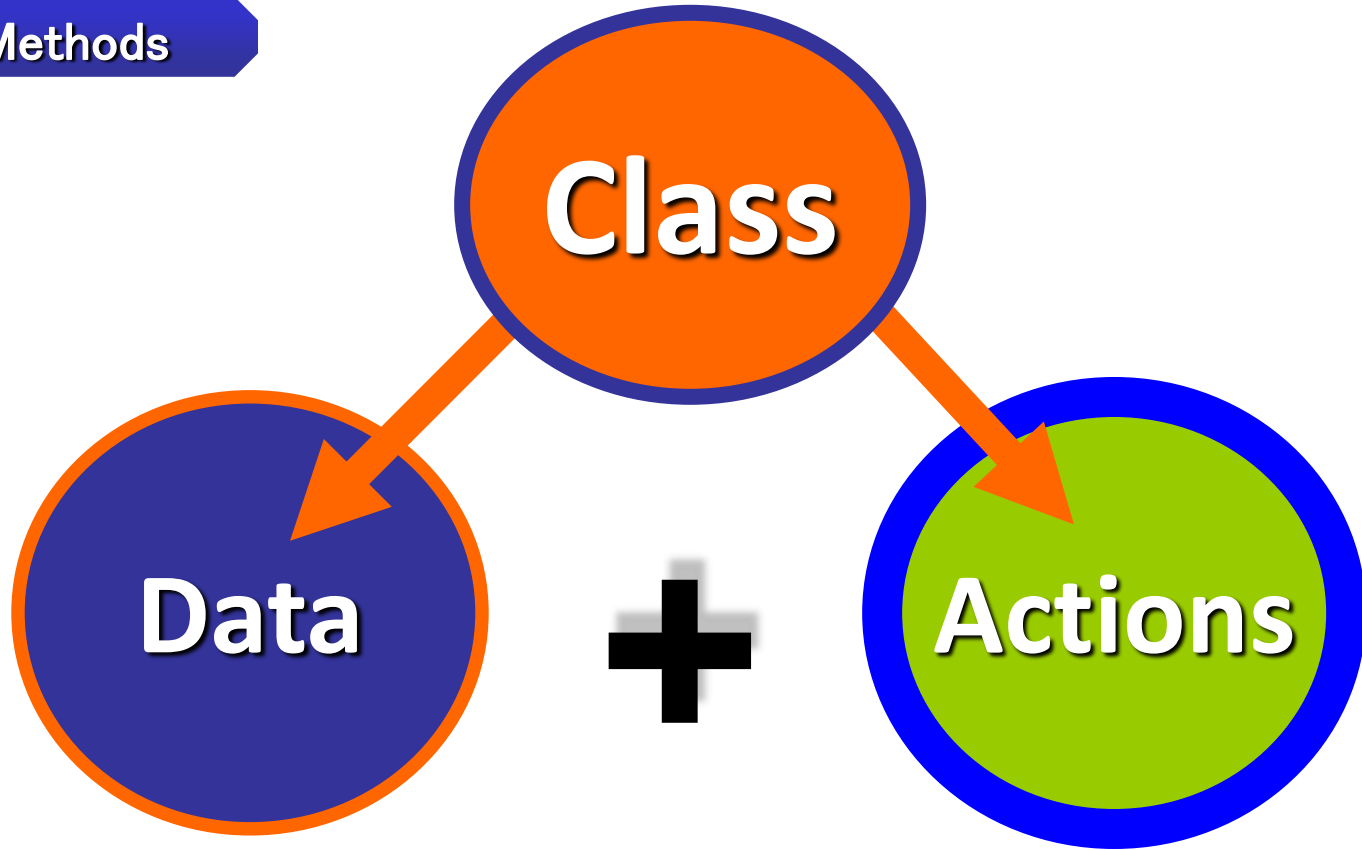


Methods





Methods





Methods

Methods = Actions



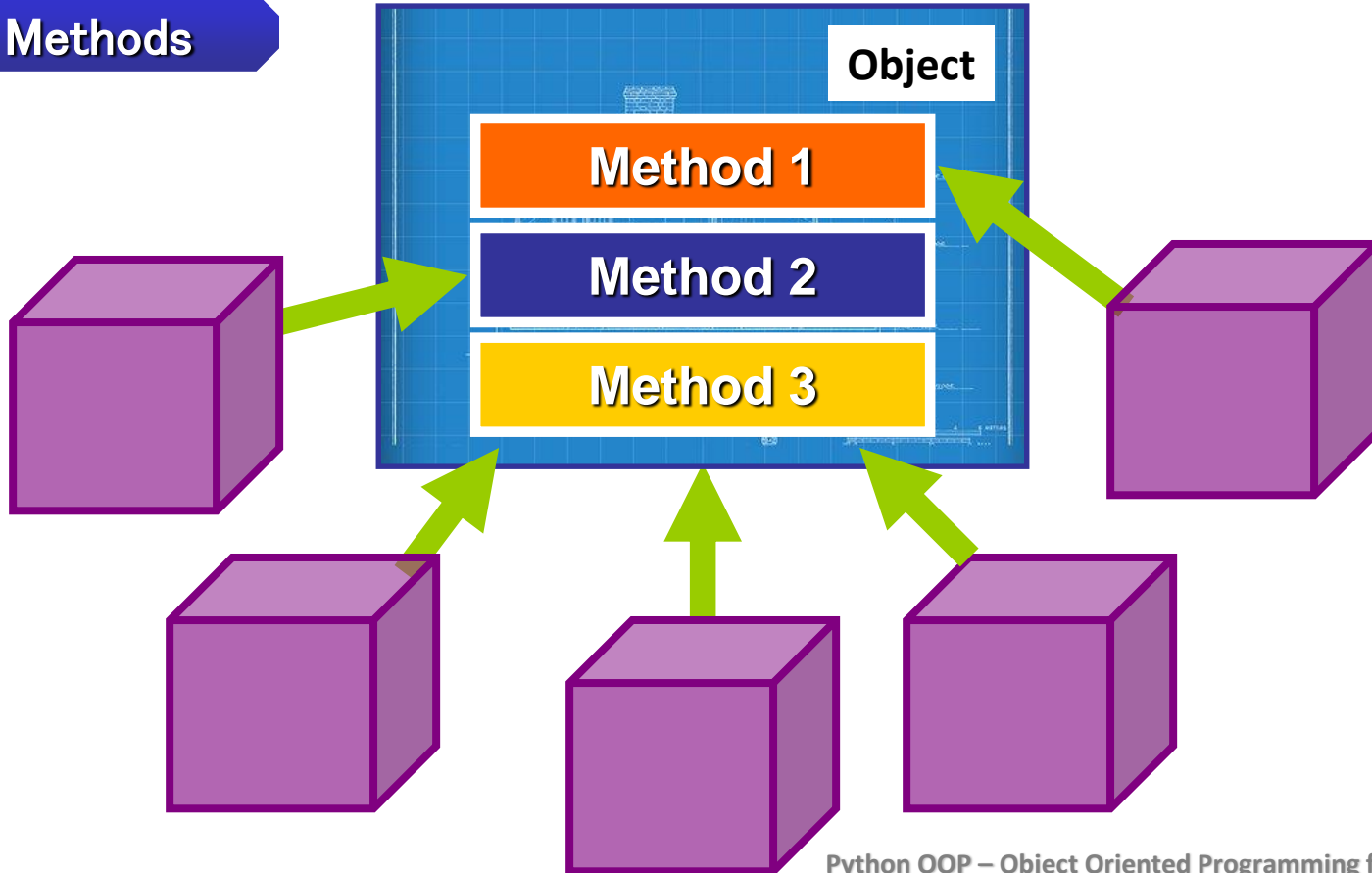
Methods

Methods = Actions

How an object can execute operations and interact with its data

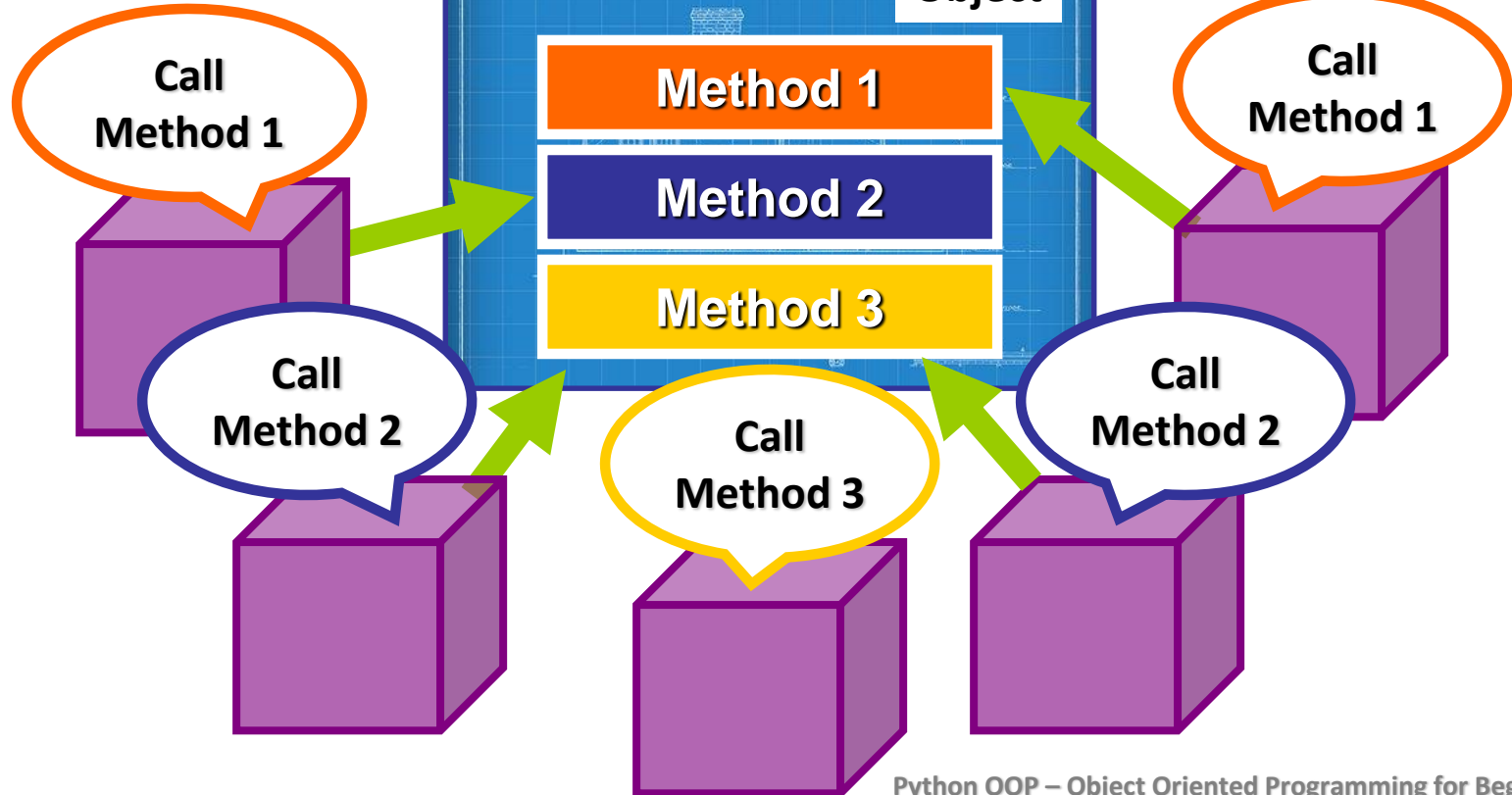


Methods





Methods





Methods

Methods have access to the data of the instance that calls them.



Methods

self

Methods have access to the data of the instance that calls them.



Methods

**Method names
include verbs**



Methods

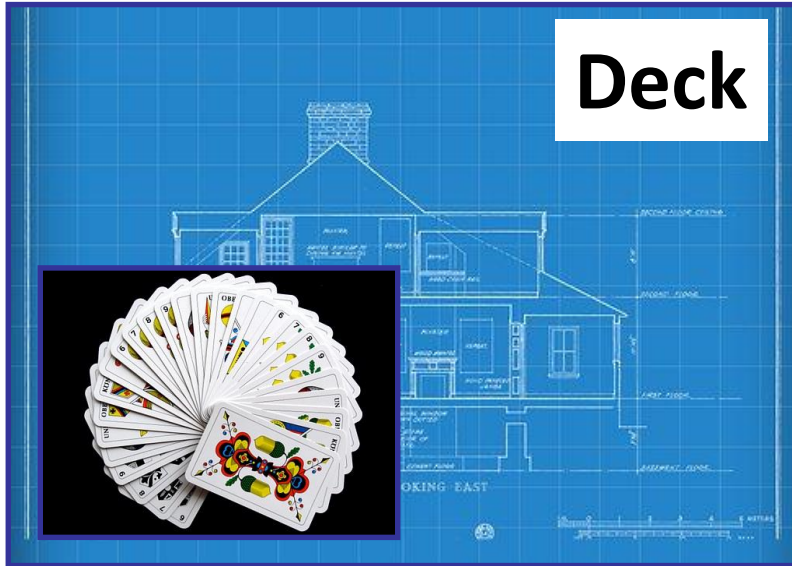
Calculator



- Add
- Subtract
- Multiply
- Divide
- More...



Methods



- Build
- Show
- Shuffle
- More...



Methods

Method Names and Instance Variables

Use the function naming rules: lowercase with words separated by underscores as necessary to improve readability.

Use one leading underscore only for non-public methods and instance variables.



Methods

Method Names and Instance Variables

Use the function naming rules: lowercase with words separated by underscores as necessary to improve readability.

Use one leading underscore only for non-public methods and instance variables.



Methods

Method Names and Instance Variables

Use the function naming rules: lowercase with words separated by underscores as necessary to improve readability.

Use one leading underscore only for non-public methods and instance variables.



Methods

```
def <method_name>(self, <params>):  
    # Body
```



Methods

Keyword

```
def <method_name>(self, <params>):  
    # Body
```



Methods

Name

```
def <method_name>(self, <params>):  
    # Body
```



Methods


Self and parameters

```
def <method_name>(self, <params>):  
    # Body
```



Methods

```
def <method_name>(self, <params>):  
    # Body
```





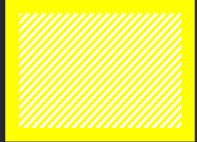
Methods

```
def <method_name>(self, <params>):  
    # Body  
    Actions
```



Methods

```
def <method_name>(self, <params>):
```



Body

Actions



Methods

```
class Calculator:
```

```
    def __init__(self, model, year, serial_num):
```

```
        self.model = model
```

```
        self.year = year
```

```
        self.__serial_num = serial_num
```

```
    def add(self, a, b):
```

```
        return a + b
```




Methods

```
class Calculator:
```

```
    def __init__(self, model, year, serial_num):  
        self.model = model  
        self.year = year  
        self.__serial_num = serial_num
```

```
    def add(self, a, b):  
        return a + b
```



Methods

```
class Calculator:
```

```
    def __init__(self, model, year, serial_num):  
        self.model = model  
        self.year = year  
        self.__serial_num = serial_num
```

```
    def add(self, a, b):  
        return a + b
```



Methods

```
class Calculator:

    def __init__(self, model, year, serial_num):
        self.model = model
        self.year = year
        self.__serial_num = serial_num

    def add(self, a, b):
        return a + b
```



Now... An Example

