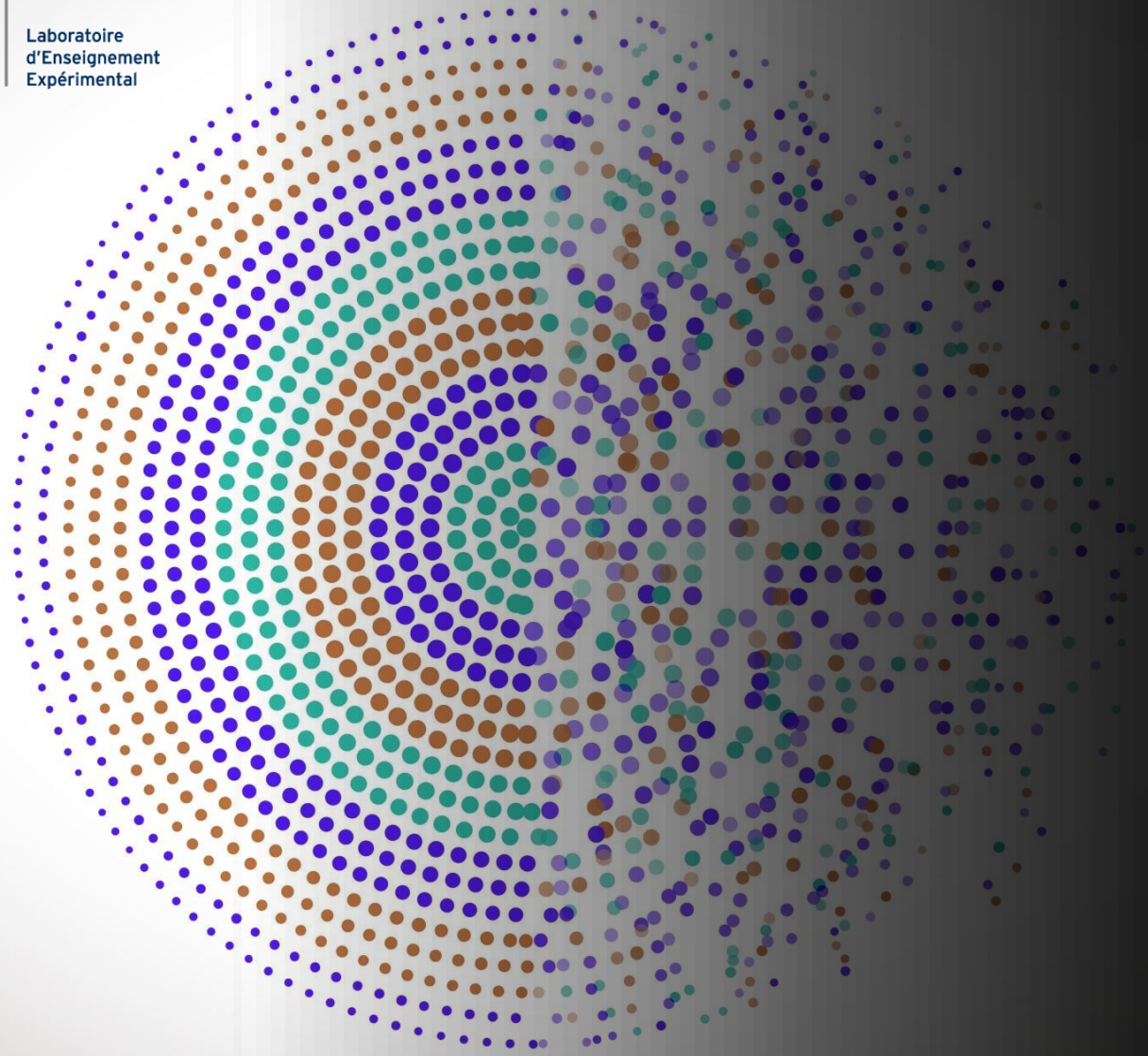
INSTITUT d'OPTIQUE GRADUATE SCHOOL ParisTech | LEnsE Laboratoire d'Enseignement Expérimental

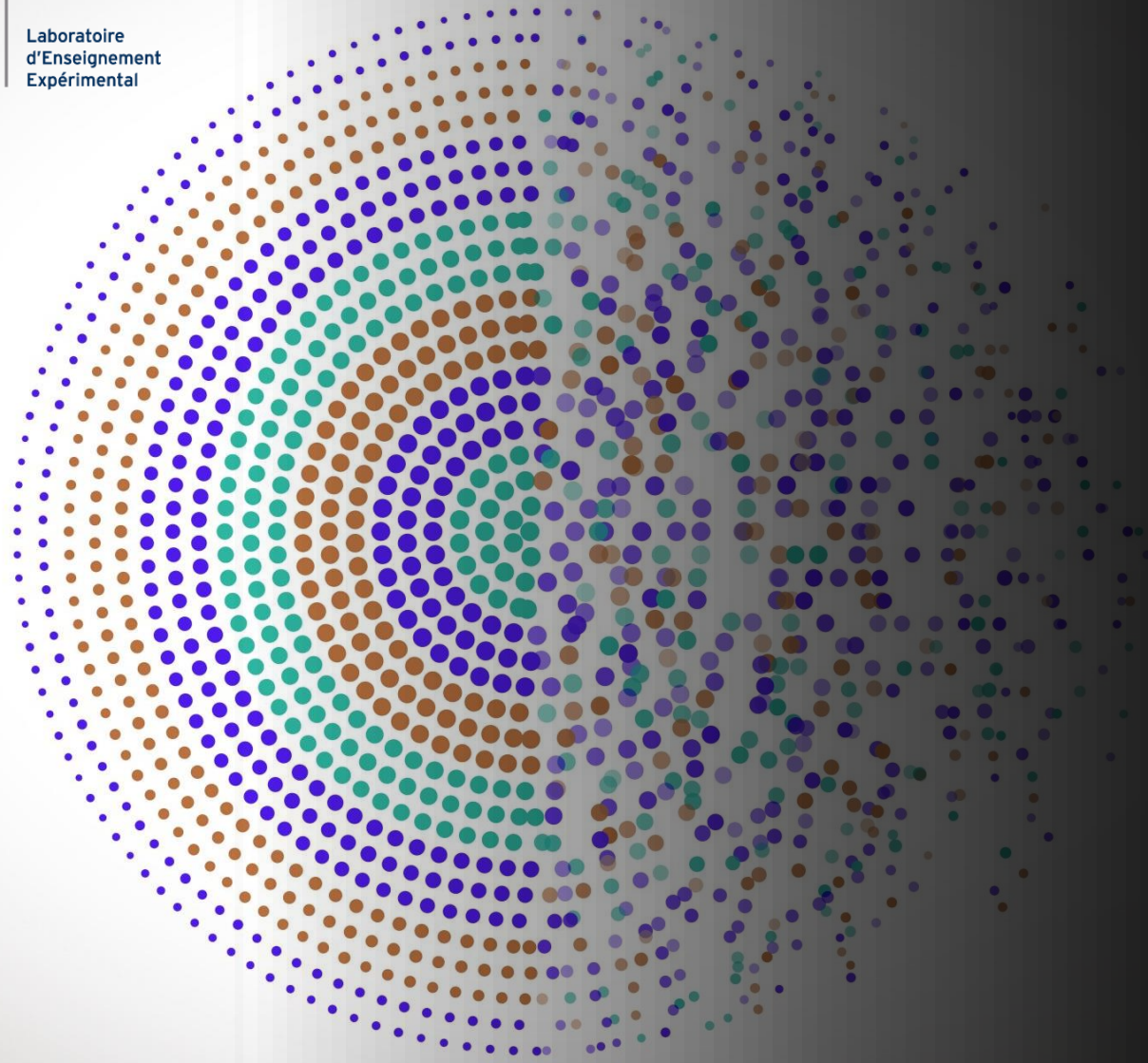


ONIP-2 / FISA

Programmation Orientée Objet

Outils Numériques / Semestre 6 / Institut d'Optique / ONIP-2

INSTITUT d'OPTIQUE GRADUATE SCHOOL ParisTech | LEnsE Laboratoire d'Enseignement Expérimental



POO
S'entrainer

Outils Numériques / Semestre 6 / Institut d'Optique / ONIP-2

S’entrainer à la POO

A travers les exemples proposés, vous serez capables de :

- Créer des **classes** incluant des **méthodes** et des **attributs**
- **Instancier des objets** et les faire interagir
- Définir et **documenter** les méthodes et attributs de chaque classe

Point

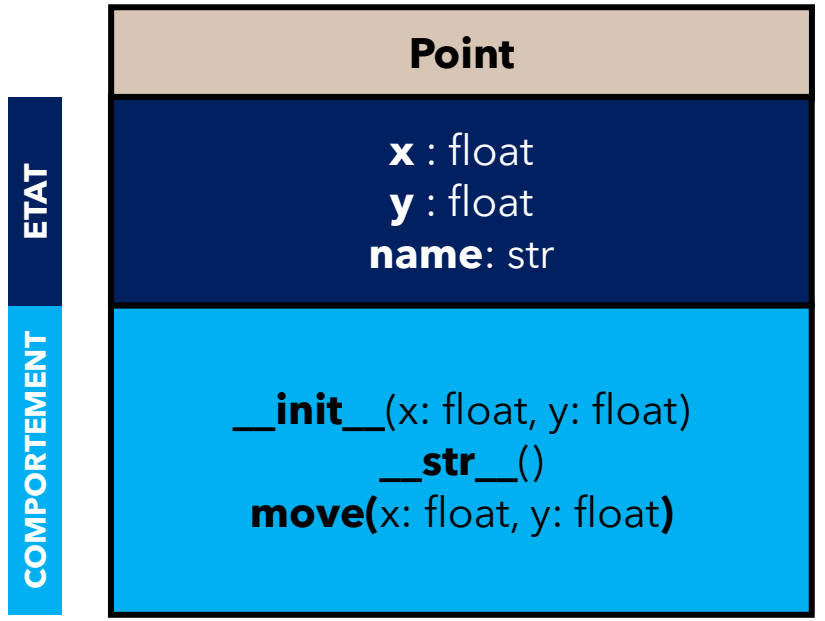
Rectangle

Cercle

Définir les classes

	ETAT	COMPORTEMENT
Point	??	??
Rectangle	??	??
Cercle	??	??

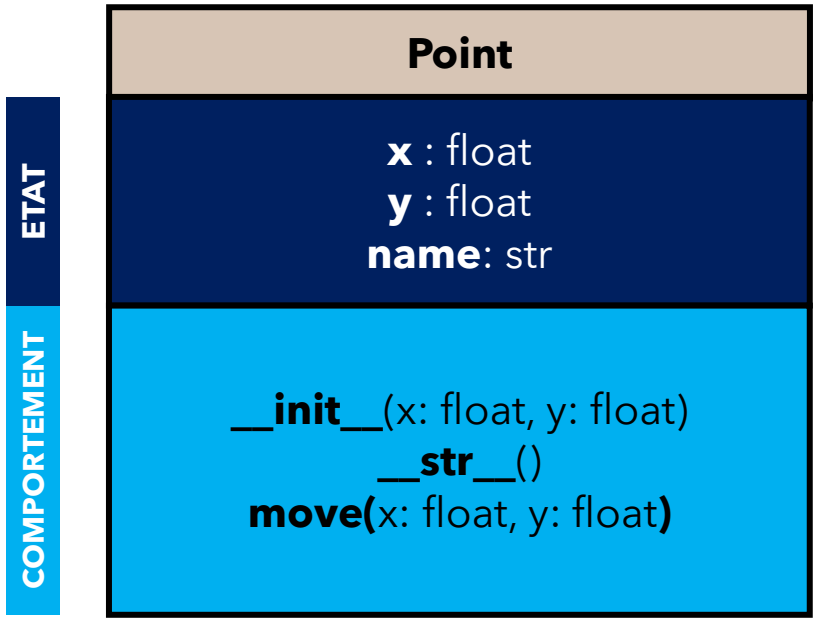
S’entrainer à la POO



S'entrainer à la POO

INSTITUT d'OPTIQUE GRADUATE SCHOOL ParisTech | LEnsE Laboratoire d'Enseignement Expérimental

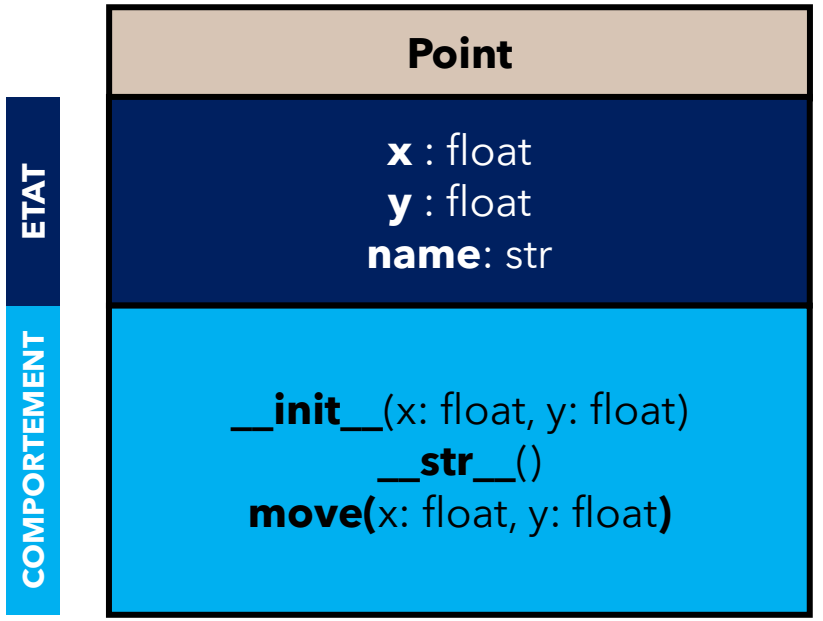
```
class Point:
    def __init__(self, x:float, y:float, name:str):
        self.x = x
        self.y = y
        self.name = name
```



S'entrainer à la POO

```
class Point:
    def __init__(self, x:float, y:float, name:str):
        self.x = x
        self.y = y
        self.name = name
```

```
pointA = Point(-0.5, 5.5, 'A')
```



S'entrainer à la POO

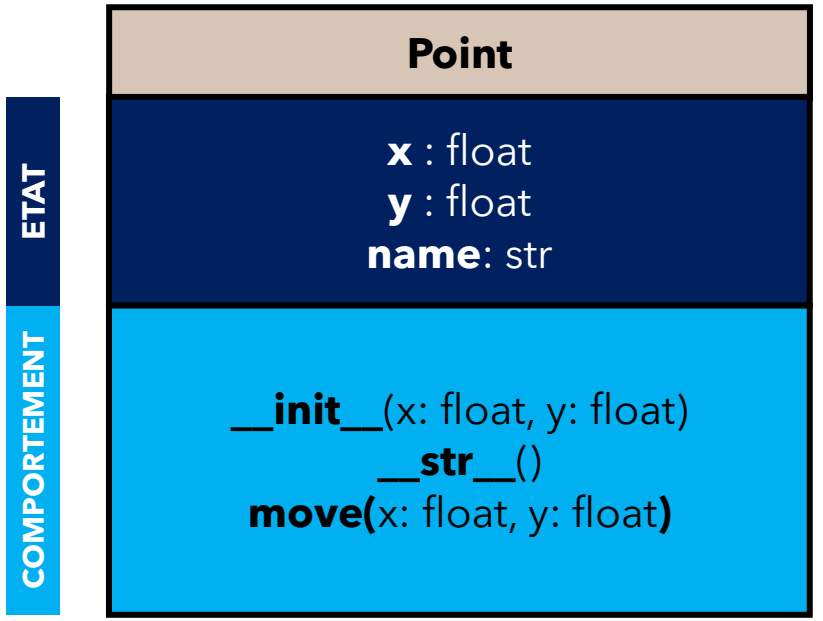
```
class Point:
    def __init__(self, x:float, y:float, name:str):
        self.x = x
        self.y = y
        self.name = name
```

```
pointA = Point(-0.5, 5.5, 'A')
```

```
    def __str__(self):
        str = f'p_{self.name} ( {self.x}, {self.y} ) '
        return str
```

```
print(pointA)
```

```
>>>  p_A ( -0.5, 5.5 )
```



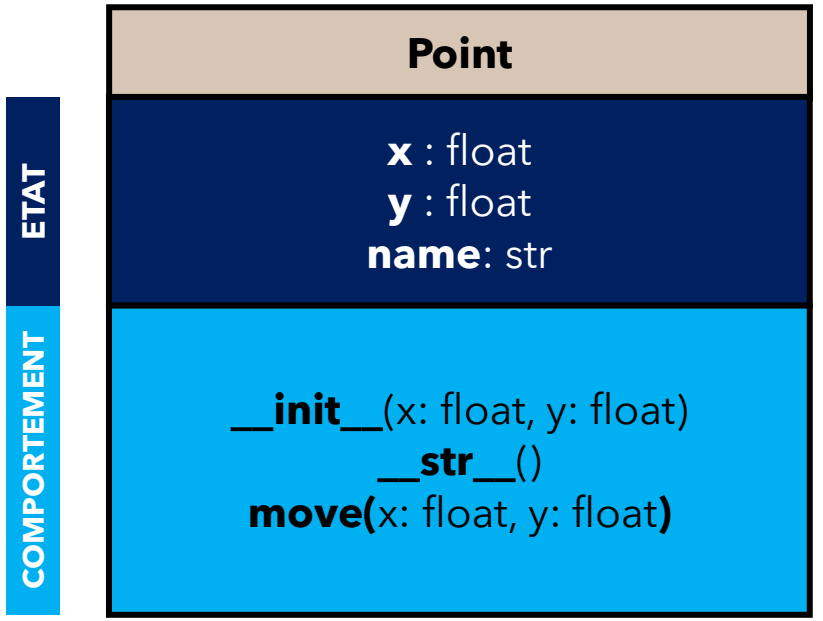
S'entrainer à la POO

```
class Point:
    def __init__(self, x:float, y:float, name:str):
        self.x = x
        self.y = y
        self.name = name
```

```
pointA = Point(-0.5, 5.5, 'A')
```

```
    def move(self, x:float, y:float):
        self.x = x
        self.y = y
```

```
pointA.move(1.0, -2.3)
```

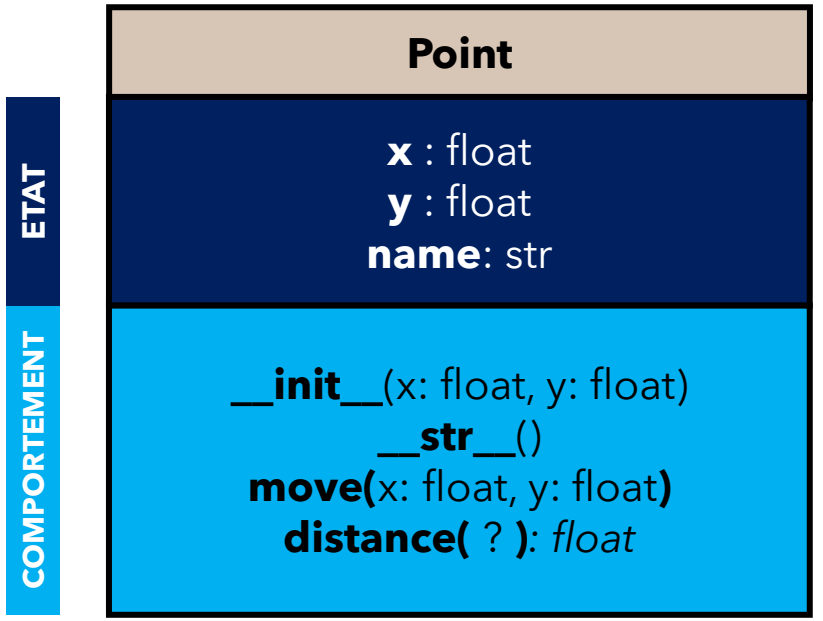


S'entrainer à la POO

```
class Point:
    def __init__(self, x:float, y:float, name:str):
        self.x = x
        self.y = y
        self.name = name

    pointA = Point(-0.5, 5.5, 'A')

    def distance(self, ??):
        ?
```



S'entrainer à la POO

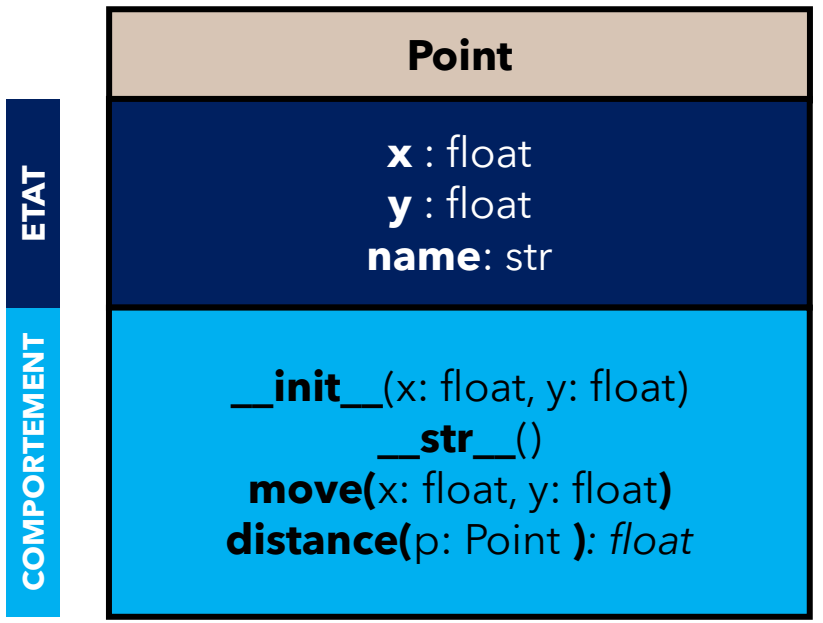
```
class Point:
    def __init__(self, x:float, y:float, name:str):
        self.x = x
        self.y = y
        self.name = name
```

```
pointA = Point(3, 6, 'A')
```

```
    def distance(self, p: Point):
        dx = self.x - p.x
        dy = self.y - p.y
        return np.sqrt( dx**2 + dy**2 )
```

```
pointB = Point(0, 10, 'B')
print( pointA.distance(pointB) )
```

```
>>> 5.0
```



S'entrainer à la POO

	ETAT	COMPORTEMENT
Point	**x** : float, **y** : float, **name**: str	**__init__**(x, y) , **__str__**() **move(**x, y**)**, **distance(Point p)**: float
Rectangle	??	??
Cercle	??	??

S'entrainer à la POO

INSTITUT d'OPTIQUE GRADUATE SCHOOL ParisTech | LEnsE Laboratoire d'Enseignement Expérimental

	ETAT	COMPORTEMENT
Point	**x** : float, **y** : float, **name**: str	**__init__**(x, y) , **__str__**() **move(**x, y**)**, **distance(Point p)**: float
Rectangle	**p1**: Point, **p2**: Point, **name**: str	**__init__**(x, y) , **__str__**() **perimetre()**: float, **surface()**: float
Cercle	**p1**: Point, **radius**: float	**__init__**(x, y) , **__str__**() **perimetre()**: float, **surface()**: float