

# Rectas

April 13, 2021

```
[1]: from math import sqrt, pi
```

```
[2]: class Point(object):  
    def __init__(self):  
        self.x = 0.0  
        self.y = 0.0
```

El identificador `Point` es el nombre que utilizaremos luego para definir objetos de este tipo. Las variables internas `x` e `y` almacenan el estado del objeto, en este caso un `Point` en dos dimensiones. Hemos decidido inicializar los `Point` por defecto a  $(0.0, 0.0)$ . Pero ¡OJO! NO podemos “ver” ese valor “conceptual” directamente, hemos de acceder a él con los “métodos” que nos hemos dado, en este caso el acceso a `x` e `y`, como veremos.

Pero, si estamos definiendo una clase de objetos es porque queremos utilizarlos ...

```
[3]: class Recta(object):  
    def __init__(self, p1, p2):  
        pass # Implementar con los atributos que prefiráis  
    def ecuacion(self):  
        return None # Devolver (a,b,c) siendo  $ax+by=c$  una ecuación de self  
    def pendiente(self):  
        return None # Devolver la pendiente p en  $y=px+c$  una ecuación de self  
        # No devuelva nada cuando es infinito  
    def punto(self):  
        return None # Devuelve un punto (x,y) de self con  $x=0$  o  $y=0$   
    def angulo(self):  
        return None # Devolver el ángulo (en radianes) que forma self con la  
        ↪ horizontal  
    def paralelas(self, r1):  
        return None # Nos dice si self y r1 son paralelas  
    def desp_paralela(self, p1):  
        pass # Modifica self desparazandola paralelamente hasta pasar por p1  
    def punto_corte(self, r1):  
        return None # Punto de corte de self y r1  
        # No devuelva nada si son paralelas  
        # Devuelve cualquier punto si son la misma recta  
    def dos_puntos(self):  
        return None # Devuelve dos puntos distintos (x,y) de self con  $x=0$  o  $y=0$ 
```

```
def perpendiculares(self,r1):  
    return none # Nos dice si self y r1 son perpendiculares  
def girar_noventa(self,p1):  
    pass # Modifica self convirtiendola en la perpendicular que pasa por p1
```

```
[ ]:
```