



Занятие №30

Программы с интерфейсом (GUI) - tkinter и pyqt6





1. Какие отличия процедурной и ООП парадигм?

Какие отличия процедурной и ООП парадигм?

ООП или Объектно-ориентированное программирование хорошо применяется в практике программирования для более лёгкого создания управляемых проектов.

Процедурный подход подразумевает написание программного кода без использования объектов. Процедурное программирование заключается в написании кода с или без подпрограмм.

ООП обучает любой язык программирования более хорошему программному коду и используется, для получения более высокой производительности и написания больших проектов, не боясь запутаться в их управлении. ООП даёт вам возможность создавать объекты которые можно будет использовать многократно, для того что бы вы или другие разработчики могли использовать их в своих проектах не переделывая их снова и снова. ООП убирает барьеры и сложности в написании и управлении большими приложениями.



```
In [12]: class Point():
        def __init__(self, x, y):
            self.x = x
            self.y = y

        def __str__(self):
            return '({}, {})'.format(self.x, self.y)

        def __eq__(self, other):
            if self.x == other.x and self.y == other.y:
                return True
            else:
                return False

        def dist(self, other):
            return ((self.x - other.x)**2 + (self.y - other.y)**2)**0.5
```

```
In [13]: A = Point(3, 4)
        B = Point(3, 4)
```

```
In [14]: A == B
```

```
Out[14]: True
```

