

✓ Воркшоп

```
class User:
    name = ''
    email = ''
    approved_by = ''

    def __init__(self, name, email):
        self.name = name
        self.email = email

    def show(self):
        print('Name: ', self.name)
        print('Email: ', self.email)
        print('Approved by: ', self.approved_by)

class AdminUser(User):
    def __init__(self, name, email):
        # super возвращает все поля и методы класса-родителя, от которого мы наследуем новый класс
        super().__init__(name, email)
        self.approved_by = 'admin'

    def approve(self, user:User):
        user.approved_by = self.name

class RegularUser(User):
    def __init__(self, name, email):
        super().__init__(name, email)
        self.approved_by = 'Not approve'

    def status(self):
        print(f'{self.name}, you status approve: ', self.approved_by)

    def change_email(self, new_email):
        self.email = new_email
        print(f'{self.name}, you email was change on {self.email}')

user = User('aleks', 'mail@alex.com')
admin = AdminUser('ivan', 'mail@ivan.com')

admin.approve(user)

user.show()
→ Name: aleks
  Email: mail@alex.com
  Approved by: ivan

admin.show()
→ Name: ivan
  Email: mail@ivan.com
  Approved by: admin

admin2 = AdminUser('petr', 'mail@petr.com')
admin2.approve(admin)

admin.show()
→ Name: ivan
  Email: mail@ivan.com
  Approved by: petr

user2 = RegularUser('Oleg', 'oleg@mail.ru')

user2.show()
→ Name: Oleg
  Email: oleg@mail.ru
```

Approved by: Not approve

```
user2.status()  
admin2.approve(user2)  
user2.status()
```

```
↔ Oleg, you status approve: Not approve  
↔ Oleg, you status approve: petr
```

```
user2.change_email('1')
```

```
↔ Oleg, you email was change on 1
```