

**Design a model showing process interaction where you are waiting for another process to get finished**

```
pip install simpy
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting simpy
  Downloading simpy-4.0.1-py2.py3-none-any.whl (29 kB)
Installing collected packages: simpy
Successfully installed simpy-4.0.1
```

```
import simpy
class Car(object):
    def __init__(self,env):
        self.env=env
        self.action=env.process(self.run())

    def run(self):
        while True:
            print("Start parking and charging at %d" %self.env.now)
            charge_duration=5
            yield self.env.process(self.charge(charge_duration))

            print("Start driving at %d" %self.env.now)
            trip_duration=2
            yield self.env.timeout(trip_duration)

    def charge(self,duration):
        yield self.env.timeout(duration)

env=simpy.Environment()
car=Car(env)
env.run(until=15)
```

```
☐ Start parking and charging at 0
Start driving at 5
Start parking and charging at 7
Start driving at 12
Start parking and charging at 14
```

```
import simpy
class Airport(object):
    def __init__(self, env):
        self.env = env
        self.action = env.process(self.checkin())

    def checkin(self):
        while True:
            print('Start checkin and closing at %d' % self.env.now)
            duration = 6
            yield self.env.process(self.opening(duration))
            print('Opening at %d' % self.env.now)
            opening_duration = 2
            yield self.env.timeout(opening_duration)

    def opening(self, duration):
        yield self.env.timeout(duration)

import simpy
env = simpy.Environment()
aitport = Airport(env)
env.run(until=31)
```

```
Start checkin and closing at 0
Opening at 6
Start checkin and closing at 8
Opening at 14
Start checkin and closing at 16
Opening at 22
Start checkin and closing at 24
Opening at 30
```

