场景介绍:
已有飞机和供货地点，当有用户下单时，需要将所选产品放入一个或多个包裹中。每个包裹限重1.8KG。有时候会因为天气原因或其他原因无法成功送货，这种状态叫做’任务失败’。 客户并不在意他们的包裹如何到达，他们只关心订单的状态和预期到达的包裹和服务。

问题:
根据已给的web-service 文件nest_sim.py 补充并完善API。 目的是提供的API可以允许用户查询可选的地址和物品存货，派遣飞机并查询飞机状态。API属于内部API无法直接暴露于客户。
任务是实施一个用户端的网站并拥有以下功能:
1.允许用户下单，可选择产品和目的地
2.允许用户查询订单状态
3.允许监管者审查所有的订单(哪些产品已经送到哪些地址)

订单的状态为:
‘CONFIRMED’:订单收到并在处理中
‘SHIPPED’:订单货物已安排航班发送
‘DELIVERED’:订单已运送完毕
‘DELAYED’:订单因为不可抗因素推迟
‘CANCELED’:订单被取消因为无法满足运送条件

首先建立一个后台，用于支持服务，数据库的资料用products.csv和hospitals.csv中的数据。
顾客只在意订单而不是飞机航班，所以在后台中使用内置的API用来把货物分散至一个或者多个航班中(因为航班限重原因)，并且跟踪他们的情况。当一个订单被分散时，’Shipped’或’Delayed’状态在用户查询时应有百分比 如80%货物已到达。

当后台建立完成后，建立一个前端网页 拥有三个功能页:
1. 注册，保存用户至数据库中，并有User_ID为primary key去锁定用户
2. 订单页：允许用户选择目的地，允许用户选择产品以及数量，确认订单成功下单，不能允许顾客选择没有库存的产品，不能要求顾客主动将订单分散至多次航班(后台应该自动为顾客确定是否需要分单以及帮助顾客分单，顾客只需要下单产品以及数量并确认)
3. 查询页：提供订单状态查询功能(分别为 ‘CONFIRMED’，‘SHIPPED’，‘DELIVERED’，‘DELAYED’，‘CANCELED’)，查询页会显示订单预计到达时间，当订单还有五分钟到达时 查询页需要提示用户订单马上到达

使用如下配置 Flask 1.0.2
Flask-SQLAlchemy 2.3.2
enum34 1.1.6
products.csv以及hospitals.csv 与程序在同一目录下，如果需要在不同目录下 请声明路径
在localhost:12345下运行

当创建一个新的航班时，会有如下状态：
‘PENDING’: 在使用API确认前，航班会一直处于该状态，该状态可以保持5分钟，5分钟内不确认该航班消除
‘CONFIRMED’:航班被客户确认并且等待装货
‘SHIPPED’:航班在路上
‘DELIVERED’:航班成功抵达
‘MISSION_FAILURE’:航班需要返回并且无法运送货物
‘COMPLETE’:所有关于该航班的操作结束

目前已提供如下 API:
Get /time:
返回从开始到现在的时间(以秒为单位)
举例:
$ curl localhost:12345/time
1122

POST /step_time <INT>
在测试的时候，可以使用此API 让进程的时间加快，如此就可以不必等待
举例：
$ curl -H "Content-Type: application/json" -X POST \
localhost:12345/step_time -d "1000"
{"success":true}

Get /inventory
返回一个JSON 列表，包含当时所有可选的产品，每条返回包含:
1. id:产品的ID
2. product:产品的名称
3. quantity:产品的剩余数量
4. mass_g:产品的重量
举例：

$ curl localhost:12345/inventory
[{"id":1,"mass_g":700.0,"product":"RBC A+ Adult","quantity":30}, ...]

Get /hospotals
返回一个JSON列表，包含医院以及需要时间，每条返回包含：
1. id：目的地的ID
2. name: 目的地的名字
3. fight_time_s：到达目的地的预期时间
举例：
Example:
$ curl localhost:12345/hospitals
[{"flight_time_s":2134,"id":1,"name":"Bigogwe"}, ...]

Post /flight {hospital:<ID>,products:[<ID1>,…,<IDn>]}
使用一个JSON创建一个新的航班，包含 目的地的ID 以及产品ID(如果产品的数量大于1，可多次添加产品ID至JSON中)。只有当 医院以及产品 都可选时 才可以创建航班，当航班创建成功后，必须由客户确认航班才可以进行下一步操作。
返回的结果为 该航班的信息介绍以及当前状态，与Get /flight/<id> 相同
举例：
$ curl -H "Content-Type: application/json" -X POST \
localhost:12345/flight -d '{"hospital": 1, "products": [1,9,9]}'
{"delivered":false,"hospital":1,"id":12,"products":[1,9,9],"state":"PENDING"}

POST /flight/<id>/confirm
通过航班ID确认航班，必须要通过这一步骤才能使航班状态脱离PENDDING
在这次模拟中，设置为有10%的可能性 航班无法创建成功(天气或不可抗拒因素)。 为测试目的，可以通过添加参数?fail=0 o或?fail=1 设置航班状态(localhost:12345/flight/12/confirm?fail=1)。
举例

Example:
$ curl -X POST localhost:12345/flight/12/confirm
{"success":true}

POST /flight/<id>/cancel
根据航班ID取消航班
举例
$ curl -X POST localhost:12345/flight/13/cancel
{"success":true}

Get /flight/<id>
返回一个JSON 包含航班状态

1. id:航班的ID
2. hospital：目的地医院的ID
3. products：产品的列表
4. state:目前航班的状态
5. delivered:是否成功送达
6. （可选） delivery_eta_s:剩余到达时间
7. （可选） returan_eta_s:剩余返回时间
8. （可选） note: 当航班状态为’COMPLETE’时，对于该航班的决议

举例：

```
$ curl localhost:12345/flight/17
{"delivered":false,"delivery_eta_s":1742,"hospital":1,"id":17,"products":[3,3],"
state":"SHIPPED"}
```