// Basic code that I quickly hashed out based on our meeting on Sept-21

package pds;

import java.util.ArrayList;

import java.util.HashMap;

enum OrderStage

{

PREP\_WAITING,

PREP,

COOK\_WAITING,

COOK,

DELIVERY\_WAITING,

DELIVERY

}

class Food

{

public double getPrice() {

return 0.0;

}

}

interface FoodItemOwner

{

public void cancel(Food food);

}

class OrderItem

{

private Food food;

private FoodItemOwner owner; // change type to interface that contains cancelFood(food) method

public Food getFood() {

return food;

}

public void cancel() {

owner.cancel(food);

}

}

class TimeRange

{

private double start, end;

public double getStart() {

return start;

}

public double getEnd() {

return end;

}

}

public class Order {

private ArrayList< OrderItem > orderItems = new ArrayList< OrderItem >();

private String destination;

private String referencePhoneNumber;

private String customerName;

private double timeCreated;

private HashMap< OrderStage, TimeRange > stageTimes = new HashMap< OrderStage, TimeRange >();

private double estimatedTime;

public void cancelOrder() {

for (OrderItem oi : orderItems) {

oi.cancel();

}

}

public double getPrice() {

double total = 0.0;

for(OrderItem oi : orderItems) {

total += oi.getFood().getPrice();

}

return total;

}

public double calculateTime() {

return TimeSystem.convertRealTimeToSimulationTime(stageTimes.get(OrderStage.DELIVERY).getEnd() - timeCreated);

}

}