package kr.yarmalovich.dish.project;

import java.lang.System.Logger;

import java.util.ArrayList;

import kr.yarmalovich.dish.project.AddSweet;

import kr.yarmalovich.dish.project.DishException;

import java.util.Collections;

public final class SearchByPrice {

private static Logger LOG =

Logger.getLogger(SearchByPrice.class);

public static final double MINIMAL\_RANGE\_VALUE = 100;

public static final double MAXIMAL\_RANGE\_VALUE = 250;

private SearchByPrice() {

}

public ArrayList<AddDish > SearchByPrice(double min,

double max, ArrayList<AddDish > Dish) throws DishException

{

if (Dish.isEmpty()){

throw new DishException("A dish cannot be without food.");

}

ArrayList<AddDish> searchResultList = new

ArrayList<AddDish>();

for (AddDish cf : Dish) {

if (cf.getPrice() >= min && cf.getPrice() <= max)

{

LOG.debug("foreach " + cf);

searchResultList.add(cf);

}

}

return searchResultList;

}

}