import java.util.Scanner;

import java.text.\*;

public class TicketMachine {

static Scanner input = new Scanner(System.in); //scanner to capture user input

static int fareIndex = 0; //variable that mathematically points to the correct price value in the array

static int totalTicketsSold = 0; //total number of tickets sold by the ticket machine across all transactions

static int ticketsThisTransaction = 1; //number of tickets for the current transaction

static double totalCurrentFare = 0; //the cash amount currently needing to be paid

static double totalCashThisTransaction = 0; //total amount of cash entered during current transaction

static double totalCash = 0; //total cash entered into ticket machine across all transactions

static int numberOfTransactions = 0; //total number of transactions

static double averageCashPerTransaction = 0; //the average cash entered per transaction

static int zone = 0; //how many zones the current customer has selected to travel, applies to all tickets purchased in same transaction

public static void main(String[] args) {

chooseZone(); //main method calls the first method which then cycles to keep the program running until terminated

}

public static void chooseZone(){ //method prompts the user to select how many zones they are traveling. This value is used to calculate the fare amount for each ticket.

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Please choose destination zone");

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Press 1 for travelling one zone");

System.out.println("Press 2 for travelling two zones"); //displays available options to the user

System.out.println("Press 3 for travelling three zones");

System.out.println("Press 4 for travelling four zones");

System.out.println("Press 5 for travelling five or more zones");

System.out.println("Press 6 to cancel transaction");

String zoneChoice = input.next(); //variable uses the scanner to record the users input

switch (zoneChoice){ //switch statement records the value based on the users travel distance that is needed to calculate the fare amount due. It also directs the program to the next method.

case"1":{

ticketType();

}

case"2":{

fareIndex += 1;

zone = 1;

ticketType();

}

case"3":{

fareIndex += 2;

zone = 2;

ticketType();

}

case"4":{

fareIndex += 3;

zone = 3;

ticketType();

}

case"5":{

fareIndex += 4;

zone = 4;

ticketType();

}

case"6":{

cancel(); //voids the transaction and returns to the first menu

}

case"98":{

admin(); //hidden option for accessing the admin statistics of the program

}

case"99":{

exit(); //hidden option to close the program

}

default:{ //when an invalid option is entered, will warn user and loop back to start of the method

System.out.println("Please choose a valid option");

chooseZone();

}

}

}

public static void ticketType(){ //method to prompt user to select the type of ticket required

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Please select type of ticket");

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Press 1 to choose an adult single ticket"); //displays options to user

System.out.println("Press 2 to choose a child single ticket");

System.out.println("Press 3 to choose an adult return ticket");

System.out.println("Press 4 to choose a child return ticket");

System.out.println("Press 5 to cancel transaction");

String ticketChoice = input.next(); //records user input

int num = Integer.parseInt(ticketChoice); //data conversion 1

String ticketCHoiceString = String.valueOf(num); //data conversion 2

switch (ticketCHoiceString){ //calculates fare amount due from it's array, based on the users selected option. Directs program to next method

case"1":{

multipleTickets();

}

case"2":{

fareIndex += 5;

multipleTickets();

}

case"3":{

fareIndex += 10;

multipleTickets();

}

case"4":{

fareIndex += 15;

multipleTickets();

}

case"5":{

cancel(); //cancels the transaction

}

default:{ //when an invalid option is entered, will warn user and loop back to start of the method

System.out.println("Please choose a valid option");

multipleTickets();

}

}

}

public static void multipleTickets(){ //method allows user to add more tickets or to proceed to payment

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Would you like to add more tickets?");

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"); //displays options to user

System.out.println("Press 1 to continue with current tickets");

System.out.println("Press 2 to add more tickets");

System.out.println("Press 3 to cancel transaction");

double[] fares = {1.9, 2.3, 2.7, 2.9, 3.1, 1, 1, 1, 1.2, 1.2, 3.5, 4.1, 4.9, 5.3, 5.6, 1.7, 1.7, 1.7, 2.1, 2.1}; //legend order: single adult, child, return adult, child

double thisFare = fares[fareIndex]; //array holds the fare prices values. Is calculated based on user choices which correspond to the index placement of the array.

totalCurrentFare += thisFare; //adds the price of the last selected ticket to the total amount due

String multipleTicketsChoice = input.next();

switch (multipleTicketsChoice){ //uses the users input to either redirect to payment method or to select another ticket to add

case"1":{

fareIndex = 0; //returns the selected zone value to 0 for next run of program

totalCashThisTransaction = totalCurrentFare; //adds the current total fare amount to the total cash amount of the current transaction

NumberFormat formatter = NumberFormat.getCurrencyInstance(); //formats the displayed amount due to have correct currency symbol and decimal places

String moneyString = formatter.format(totalCurrentFare);

System.out.println("Current amount due is: " + moneyString);

paymentOptions(); //directs program to next method in flow

}

case"2":{

fareIndex = zone; //sets the fareIndex value for additional tickets to carry forward for each added ticket so the user doesn't have to select it again every time

ticketsThisTransaction++; //increments the amount of tickets being purchased in the current transaction

NumberFormat formatter = NumberFormat.getCurrencyInstance(); //formats the displayed amount due to have correct currency symbol and decimal places

String moneyString = formatter.format(totalCurrentFare);

System.out.println("Current amount due is: " + moneyString);

ticketType(); //directs to previous method to add more tickets

}

case"3":{ //cancels the transaction

cancel();

}

default:{ //when an invalid option is entered, will warn user and loop back to start of the method

System.out.println("Please choose a valid option");

multipleTickets();

}

}

}

public static void paymentOptions(){ //method calculates amount still due and change amount required

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Please enter cash amount"); //prompts user to enter amount of cash entered

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

String cash = input.next(); //reads user input

double cashAmountEntered = Double.parseDouble(cash); //parsing the amount entered to a double in order to use for mathematical formula

if (cashAmountEntered < totalCurrentFare){ //checks if the amount of cash entered is less then the current amount due

totalCurrentFare -= cashAmountEntered; //subtracts the amount of cash entered from the amount due

NumberFormat formatter = NumberFormat.getCurrencyInstance(); //formats the displayed amount due to have correct currency symbol and decimal places

String moneyString = formatter.format(totalCurrentFare);

System.out.println("Current amount due is: " + moneyString); //displays new amount currently due

paymentOptions();

}

else{ //runs if amount of cash entered exceeds total amount due

double change = cashAmountEntered - totalCurrentFare; //subtracts total amount due from amount of cash entered to calculate change to dispense

NumberFormat formatter = NumberFormat.getCurrencyInstance(); //formats the displayed amount of change to have correct currency symbol and decimal places

String moneyString = formatter.format(change);

System.out.println("Your change is: " + moneyString); //displays amount of change dispensed

printTickets();

}

}

public static void printTickets(){ //dispenses tickets, sets per transaction variables back to base values and adds values to statistic variables for the ticket machine accessed with admin method

totalTicketsSold += ticketsThisTransaction;

totalCash += totalCashThisTransaction;

numberOfTransactions++;

averageCashPerTransaction = totalCash/numberOfTransactions;

fareIndex = 0;

ticketsThisTransaction = 1;

totalCurrentFare = 0;

totalCashThisTransaction = 0;

zone = 0;

System.out.println("Thank you for travelling with LUAS. Please take your tickets.");

chooseZone();

}

public static void cancel(){ //cancels an individual transactions while keeping ticket machine total statistics intact

fareIndex = 0;

ticketsThisTransaction = 1;

totalCurrentFare = 0;

totalCashThisTransaction = 0;

zone = 0;

chooseZone();

}

public static void exit(){ //method accessed with hidden option to close the program

System.exit(0);

}

public static void admin(){ //hidden method that displays statistics about ticket machine

System.out.println("Total transactions: " + numberOfTransactions); //displays total number of transactions

System.out.println("Total tickets sold: " + totalTicketsSold); //displays total number of tickets sold

NumberFormat formatter = NumberFormat.getCurrencyInstance(); //formats the displayed amount to have correct currency symbol and decimal places

String moneyString = formatter.format(totalCash);

System.out.println("Total cash amount: " + moneyString); //displays total cash amount lodged into machine

NumberFormat formatter1 = NumberFormat.getCurrencyInstance(); //formats the displayed amount to have correct currency symbol and decimal places

String moneyString1 = formatter1.format(averageCashPerTransaction);

System.out.println("Average cash per transaction: " + moneyString1); //displays the average cash lodged per transaction

chooseZone();

}

}