**Звіт**

**Лабораторна работа 7. Об'єктно-орієнтована декомпозиція**

**Мета роботи**: Використання об'єктно-орієнтованого підходу для розробки об'єкта предметної (прикладної) галузі.

**ВИМОГИ**

* Використовуючи об'єктно-орієнтований аналіз, реалізувати класи для представлення сутностей відповідно прикладної задачі - domain-об'єктів.
* Забезпечити та продемонструвати коректне введення та відображення кирилиці.
* Продемонструвати можливість управління масивом domain-об'єктів.

9. Квиткова каса

Дані про маршрут: маршрут - необмежений набір значень у вигляді "назва станції, час прибуття (для проміжних і кінцевої), час відправлення (для початкової та проміжних), кількість вільних місць"; загальна кількість місць; дні тижня; номер рейсу.

* 1. **Розробник**: Капелька Ярослав Іванович, КІТ-119а, варіант №9.

1. **ОПИС ПРОГРАМИ**
   1. **Засоби ООП**: клас, метод класу, поле класу.
   2. **Ієрархія та структура класів:** один публічний клас Main та публічний клас RouteClass, у якого є поля: назва маршруту, загальна кількість місць; дні тижня; номер рейсу, гетери, сетери, конструктор класу та метод виведення даних класу, класс StationClass, у якого є поля: назва станції, час прибуття , час відправлення, кількість вільних місць, статус станції"; гетери, сетери, конструктор класу та метод виведення даних класу, публічний класс RoutesClass, який зберігає масив об’єктів класу RouteClass.
   3. **Важливі фрагменти програми:**

**public** **class** RoutesClass

{

RouteClass[] route = **new** RouteClass[2];

}

**public** **class** RouteClass

{

**private** String name\_route;

**private** **int** total\_number\_of\_seats;

**private** GregorianCalendar days\_of\_the\_week;

**private** **int** flight\_number;

StationClass[] station = **new** StationClass[3];

RouteClass route;

**public** StationClass[] getStation()

{

**return** station;

}

**public** **void** setStation(StationClass[] station)

{

**this**.station = station;

}

**public** String getName\_route()

{

**return** name\_route;

}

**public** **void** setName\_route(String name\_route)

{

**this**.name\_route = name\_route;

}

**public** **int** getTotal\_number\_of\_seats()

{

**return** total\_number\_of\_seats;

}

**public** **void** setTotal\_number\_of\_seats(**int** total\_number\_of\_seats)

{

**this**.total\_number\_of\_seats = total\_number\_of\_seats;

}

**public** GregorianCalendar getDay\_of\_the\_week()

{

**return** days\_of\_the\_week;

}

**public** **void** setDays\_of\_the\_week(GregorianCalendar days\_of\_the\_week)

{

**this**.days\_of\_the\_week = days\_of\_the\_week;

}

**public** **int** getFlight\_number()

{

**return** flight\_number;

}

**public** **void** setFlight\_number(**int** flight\_number)

{

**this**.flight\_number = flight\_number;

}

**public** RouteClass() {};

**public** RouteClass(String name,StationClass[] station, **int** total\_number,GregorianCalendar days,**int** flight)

{

**this**.name\_route = name;

**this**.station = station;

total\_number\_of\_seats = total\_number;

**this**.days\_of\_the\_week = days;

flight\_number = flight;

}

**public** **void** outputData()

{

System.***out***.println("Название маршрута: " + name\_route);

System.***out***.println("Информация о станциях, присутствующих в даном маршруте: ");

**for** (**int** i = 0; i < station.length; i++)

{

System.***out***.println("Информация о станции номер: " + (i + 1) + ". ");

station[i].outputDataStation();

}

System.***out***.println("Общее количество мест: " + total\_number\_of\_seats);

System.***out***.println("День недели, когда доступен маршрут: " + days\_of\_the\_week.getTime());

System.***out***.println("Номер рейса: " + flight\_number);

}

}

**public** **class** StationClass

{

**private** String station\_name;

**private** String departure\_time;

**private** String arrival\_time;

**private** **int** number\_of\_free\_seats;

**private** String status\_station;

**public** String getStation\_name()

{

**return** station\_name;

}

**public** **void** setStation\_name(String station\_name)

{

**this**.station\_name = station\_name;

}

**public** String getDeparture\_time()

{

**return** departure\_time;

}

**public** **void** setDeparture\_time(String departure\_time)

{

**this**.departure\_time = departure\_time;

}

**public** String getArrival\_time()

{

**return** arrival\_time;

}

**public** **void** setArrival\_time(String arrival\_time)

{

**this**.arrival\_time = arrival\_time;

}

**public** **int** getNumber\_of\_free\_seats()

{

**return** number\_of\_free\_seats;

}

**public** **void** setNumber\_of\_free\_seats(**int** number\_of\_free\_seats)

{

**this**.number\_of\_free\_seats = number\_of\_free\_seats;

}

**public** String getStatus\_station()

{

**return** status\_station;

}

**public** **void** setStatus\_station(String status\_station)

{

**this**.status\_station = status\_station;

}

**public** StationClass() {};

**public** StationClass(String name, String time,String time1, **int** number, String status)

{

**this**.station\_name = name;

**this**.arrival\_time = time;

**this**.departure\_time = time1;

**this**.number\_of\_free\_seats = number;

**this**.status\_station = status;

}

**public** **void** outputDataStation()

{

System.***out***.println("Названия станции: " + station\_name);

System.***out***.println("Время прибытия на станцию: " + arrival\_time);

System.***out***.println("Время отправления с станции: " + departure\_time);

System.***out***.println("Количество пустых мест: " + number\_of\_free\_seats);

System.***out***.println("Статус станции: " + status\_station);

}

}

**public** **class** Main07

{

**public** **static** **void** main(String[] args)

{

String Name\_route1 = "Лозовая-Харьков";

String Name\_route2 = "Харьков-Изюм";

String Station\_name1 = "Лозовая";

String Station\_name2 = "Панютино";

String Station\_name3 = "Харьков";

String Station\_name4 = "Харьков";

String Station\_name5 = "Маяк";

String Station\_name6 = "Изюм";

String arrival\_time1 = "15:20";

String arrival\_time2 = "15:35";

String arrival\_time3 = "19:13";

String arrival\_time4 = "11:55";

String arrival\_time5 = "14:52";

String arrival\_time6 = "15:41";

String departure\_time1 = "15:24";

String departure\_time2 = "15:36";

String departure\_time3 = "Времени отправления нету, так как это конечная станция.";

String departure\_time4 = "12:00";

String departure\_time5 = "14:53";

String departure\_time6 = "Времени отправления нету, так как это конечная станция.";

**int** number\_of\_free\_seats1 = 55;

**int** number\_of\_free\_seats2 = 78;

**int** number\_of\_free\_seats3 = 63;

**int** number\_of\_free\_seats4 = 87;

**int** number\_of\_free\_seats5 = 61;

**int** number\_of\_free\_seats6 = 75;

String Status\_station1 = "Начальная";

String Status\_station2 = "Промежуточная";

String Status\_station3 = "Конечная";

String Status\_station4 = "Начальная";

String Status\_station5 = "Промежуточная";

String Status\_station6 = "Конечная";

GregorianCalendar days\_of\_the\_week1 = **new** GregorianCalendar(2020, 11, 7);

days\_of\_the\_week1.set(Calendar.***HOUR\_OF\_DAY***, 00);

days\_of\_the\_week1.set(Calendar.***MINUTE***, 00);

days\_of\_the\_week1.set(Calendar.***SECOND***, 00);

GregorianCalendar days\_of\_the\_week2 = **new** GregorianCalendar(2020, 11, 8);

days\_of\_the\_week2.set(Calendar.***HOUR\_OF\_DAY***, 00);

days\_of\_the\_week2.set(Calendar.***MINUTE***, 00);

days\_of\_the\_week2.set(Calendar.***SECOND***, 00);

StationClass Station\_list[] = **new** StationClass[3];

Station\_list[0] = **new** StationClass(Station\_name1, arrival\_time1, departure\_time1, number\_of\_free\_seats1, Status\_station1);

Station\_list[1] = **new** StationClass(Station\_name2, arrival\_time2, departure\_time2, number\_of\_free\_seats2, Status\_station2);

Station\_list[2] = **new** StationClass(Station\_name3, arrival\_time3, departure\_time3, number\_of\_free\_seats3, Status\_station3);

StationClass Station\_list1[] = **new** StationClass[3];

Station\_list1[0] = **new** StationClass(Station\_name4, arrival\_time4, departure\_time4, number\_of\_free\_seats4, Status\_station4);

Station\_list1[1] = **new** StationClass(Station\_name5, arrival\_time5, departure\_time5, number\_of\_free\_seats5, Status\_station5);

Station\_list1[2] = **new** StationClass(Station\_name6, arrival\_time6, departure\_time6, number\_of\_free\_seats6, Status\_station6);

RoutesClass Route\_list = **new** RoutesClass();

Route\_list.route[0] = **new** RouteClass(Name\_route1, Station\_list, 150, days\_of\_the\_week1, 1);

Route\_list.route[1] = **new** RouteClass(Name\_route2, Station\_list1, 200, days\_of\_the\_week2, 2);

**for**(RouteClass route : Route\_list.route)

{

route.outputData();

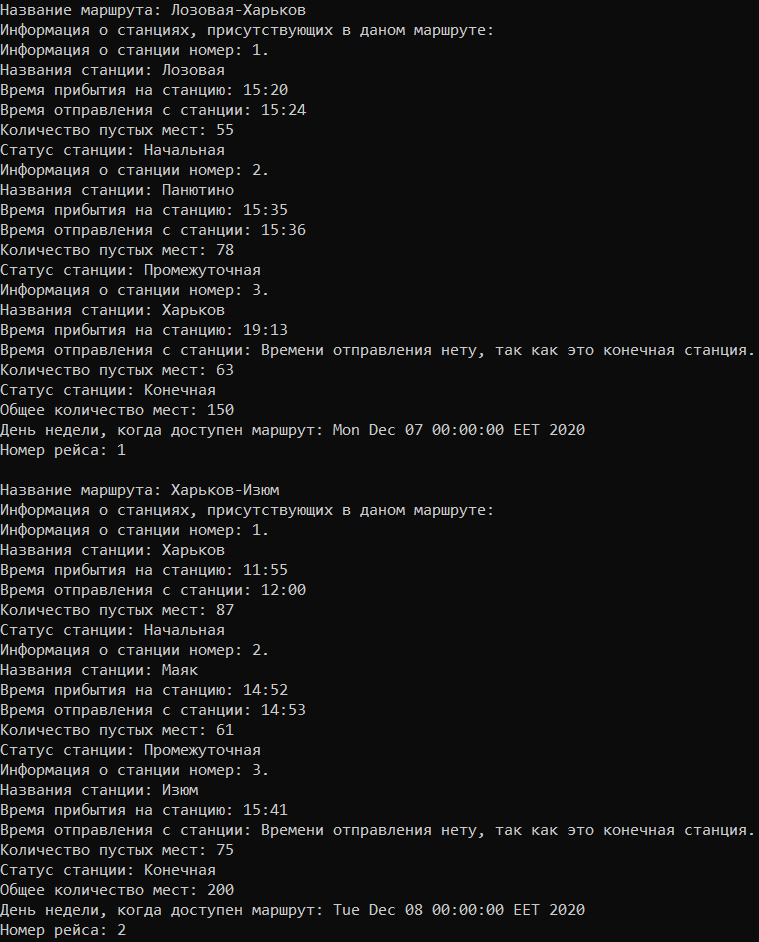
System.***out***.println();

}

}

}

**Результат роботи програми:**

****

**Висновки**

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи з об’єктно-орієнтованою декомпозицією.

Програма протестована, виконується без помилок.