

Лабораторная работа №2. Знакомство с объектно-ориентированным программированием на Java

Цель работы: Научиться наследовать классы, добавлять новые методы, частично или полностью изменять методы наследованного класса

Ход выполнения работы:

1. Ознакомьтесь с текстом программы vagon.java (находится в полкаталоге лабораторной работы):

```
public class vagon {

    protected int ChisloMest=36;
    private int [ ] Mesta;
    public int NomerEtogoVagona;

    public void PustojVagon(int NomerVagona) {
        NomerEtogoVagona=NomerVagona;
        Mesta = new int[ChisloMest];
        for (int i=0;i<ChisloMest;i++) Mesta[i]=0;
        // 0 - признак того, что место свободно
    }

    public int MestoZanyato(int NomerMesta) {
        return Mesta[NomerMesta-1];
    }

    public void ZanyatxMesto(int NomerMesta) {
        Mesta[NomerMesta-1]=1;
        // 1 - признак того, что место занято
    }

    public void VagonInfo() {
        System.out.println("Информация о свободных местах:");
        for (int i=0;i<ChisloMest;i++)
            if (Mesta[i]==0) System.out.print((i+1)+" ");
        System.out.println("");
    }

}
```

Скопируйте его на свой компьютер и откомпилируйте.

Этот класс позволяет резервировать места в железнодорожном вагоне и проверять их занятость.

Создать пустой вагон со всеми незанятыми местами можно методом `PustojVagon(int NomerVagona)`, указав в качестве входного параметра номер вагона.

Зарезервировать (занять) место можно методом `ZanyatxMesto(int NomerMesta)`, указав в качестве параметра номер желаемого места.

Проверить, занято ли место, можно методом `int MestoZanyato(int NomerMesta)`, указав в качестве параметра номер места. Метод возвращает 0, если место не занято и 1, если место занято.

Вывести на экран информацию о еще не занятых местах можно методом `VagonInfo()`.

2. Создайте класс `vagon2`, который будет наследовать класс `vagon`.

Добавьте в него новый метод:

```
public void IzmenitxChisloMest(int NovoeChisloMest)
```

который будет изменять число мест в вагоне (поле `ChisloMest`).

Метод `ZanyatxMesto(int NomerMesta)` модифицируйте таким образом, чтобы он перед резервированием места вначале проверял, не занято ли оно и, если уже занято, то выводил на экран надпись: "Это место уже занято!" и номер этого места и вагона.

Метод `VagonInfo()` модифицируйте таким образом, чтобы вначале он выводил номер вагона и общее число свободных мест. А затем уже, как и раньше, перечень свободных мест.

3. Ознакомьтесь с текстом программы `poezd.java` (лежит в той же папке, что и лабораторная работа). Откомпилируйте ее. Эта программа использует созданный вами класс `vagon2`.

```

public class poezd {

    public static void main(String[] args) {

        vagon2 v1 = new vagon2();
        v1.PustojVagon(10);
        v1.ZanyatxMesto(5);
        v1.ZanyatxMesto(15);
        v1.VagonInfo();

        vagon2 v2 = new vagon2();
        v2.IzmenitxChisloMest(16);
        v2.PustojVagon(14);
        v2.ZanyatxMesto(5);
        v2.ZanyatxMesto(7);
        v2.ZanyatxMesto(5);
        v2.VagonInfo();

    }

}

```

в результате компиляции и запуска этой программы на экран должен быть выдан примерно такой текст:

```

Вагон №10
Число свободных мест: 34
Информация о свободных местах:
1 2 3 4 6 7 8 9 10 11 12 13 14 16 17 18 19 20 21 22 23
24 25 26 27 28 29 30 31 32 33 34 35 36
Это место уже занято: 5 (вагон №14)!
Вагон №14
Число свободных мест: 14
Информация о свободных местах:
1 2 3 4 6 8 9 10 11 12 13 14 15 16

```

4. Ответьте на вопросы преподавателя по тексту программы vagon2.java