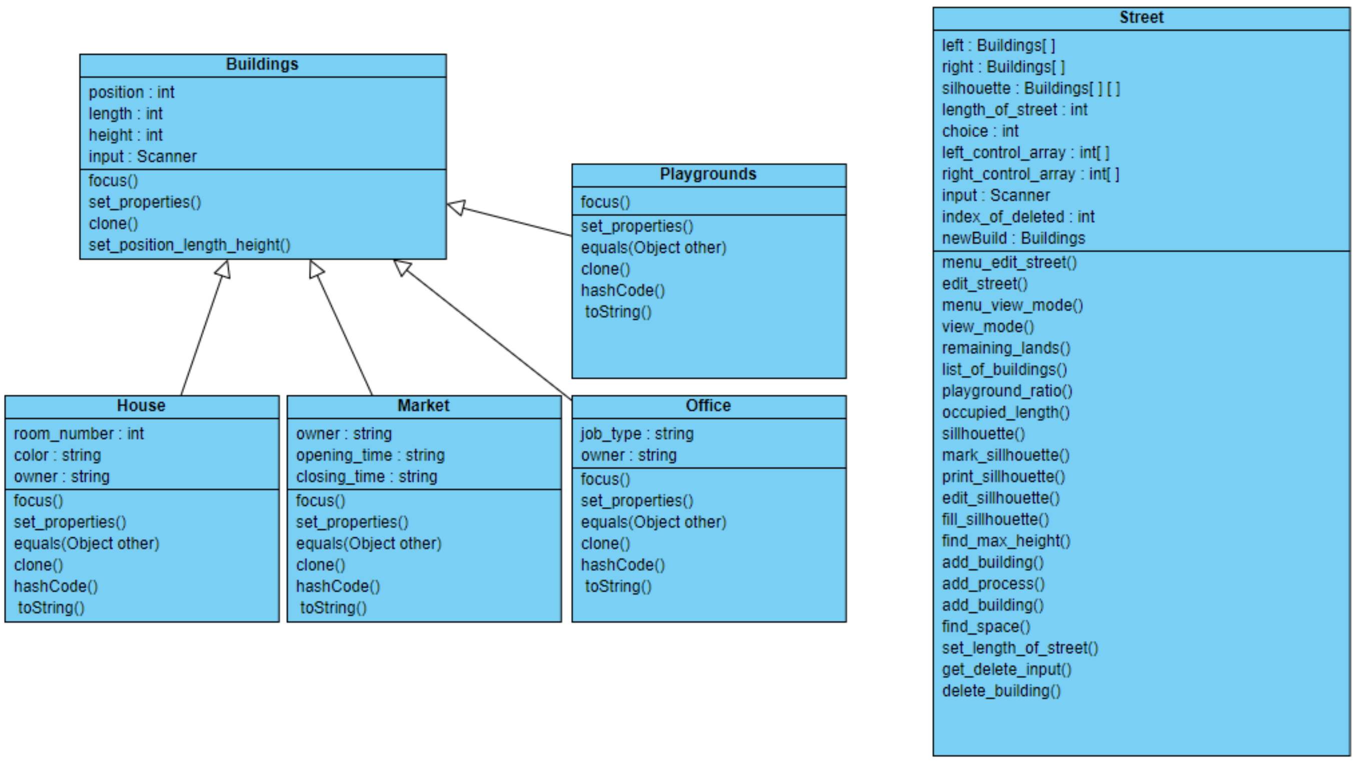
1 – System Requirements

In the homework, we are asked to make program that presents a two sided street. Then we need a Street class that operates required functions like adding and deleting. Since Street has two side, it has 2 array to represent both side. As we add and delete buildings, we need building classes (house, office, market, playgrounds). And one superclass for them(“Buildings“ class).

We have 4 building type which has some common and not common properties. For common properties such as position, length, height fields and functions like focus(), we agreed that these 4 building have to be subclass of a superclass so execute methods polymorphically. As we are not allowed to use data structures, we used basic arrays which is hard to copy,delete and add item.

2- Use Case and Class Diagrams



4- Problem Solution Approach

The task is creating an program that is able to add/delete building which is created by user given inputs and then drawing silhouette of street.

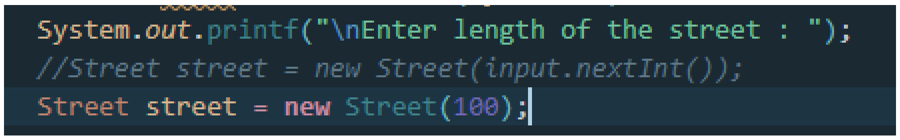
I created a Street class that presents street. I assumed that when we are looking silhouette, near side is right side, further side is left side. Class has 2 Buildings array , one for ‘left’ and other one for ‘right’. It stores Building objects that exist at that time. Adding and deleting methods works on these arrays. There are two 1D int helper array for both side which is used in adding and deleting. It marks with 1 the places that occupied by buildings. And there is a 2D int array which is used in printing silhouette. In printing, at first part I combine both side of street in one 2D int array which is actually 2D view of street. Some of the buildings will use the same indexes of that 2D int array but it’s not problem because it will be silhouette . After that, I cleared the cells which are not outline walls of buildings. So, we will have an 2D array that represents silhouette.

In adding function, user enters type of the building. After creating building, user gives property inputs(position, length and uncommon variables) for that building. After that, I look into helper control array and try to find space in chosen side of street. Deleting function takes 2 two parameter (position and side). Position doesn't have to be beginning position of building. It can be any point of a building. Street class has Building reference field. This reference provides convenience in adding and deleting by passing into methods.

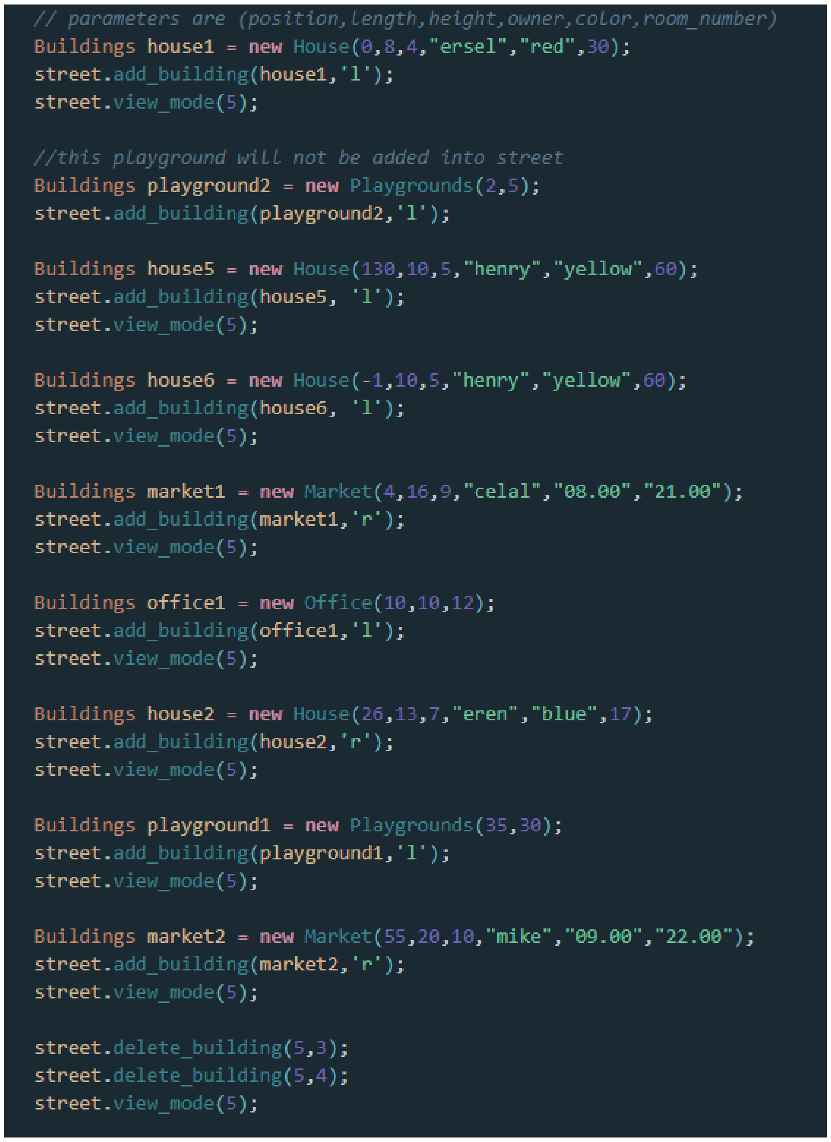
"Buildings" is superclass of House, Office, Market and Playgrounds. These classes have overridden toString, equals, clone and hashcode methods. They have some polymorphic methods like focus() that prints particular informations of all buildings in street. Position, length and height variables are superclass fields. They are protected, so they can be set by user input from subclasses.

5 – Test Cases

Create Street. Size can be set by user input but for test cases, it is comment line.



```java
System.out.printf("\nEnter length of the street : ");
//Street street = new Street(input.nextInt());
Street street = new Street(100);
```



```java
// parameters are (position,length,height,owner,color,room_number)
```

Create house, add it to left side and Print silhoeutte

```java
Buildings house1 = new House(0,8,4,"ersel","red",30);
street.add_building(house1,'l');
street.view_mode(5);
```

Create playground, but its side and position Is intersecting with house1. So it won't be added

```java
//this playground will not be added into street
Buildings playground2 = new Playgrounds(2,5);
street.add_building(playground2,'l');
```

Create house5. It exceeds bound of street

```java
Buildings house5 = new House(130,10,5,"henry","yellow",60);
street.add_building(house5, 'l');
street.view_mode(5);
```

Create house6. Position is -1. It is invalid

```java
Buildings house6 = new House(-1,10,5,"henry","yellow",60);
street.add_building(house6, 'l');
street.view_mode(5);
```

Create market1. It is valid

```java
Buildings market1 = new Market(4,16,9,"celal","08.00","21.00");
street.add_building(market1,'r');
street.view_mode(5);
```

Office1 valid

```java
Buildings office1 = new Office(10,10,12);
street.add_building(office1,'l');
street.view_mode(5);
```

House2 valid

```java
Buildings house2 = new House(26,13,7,"eren","blue",17);
street.add_building(house2,'r');
street.view_mode(5);
```

Playground2 valid

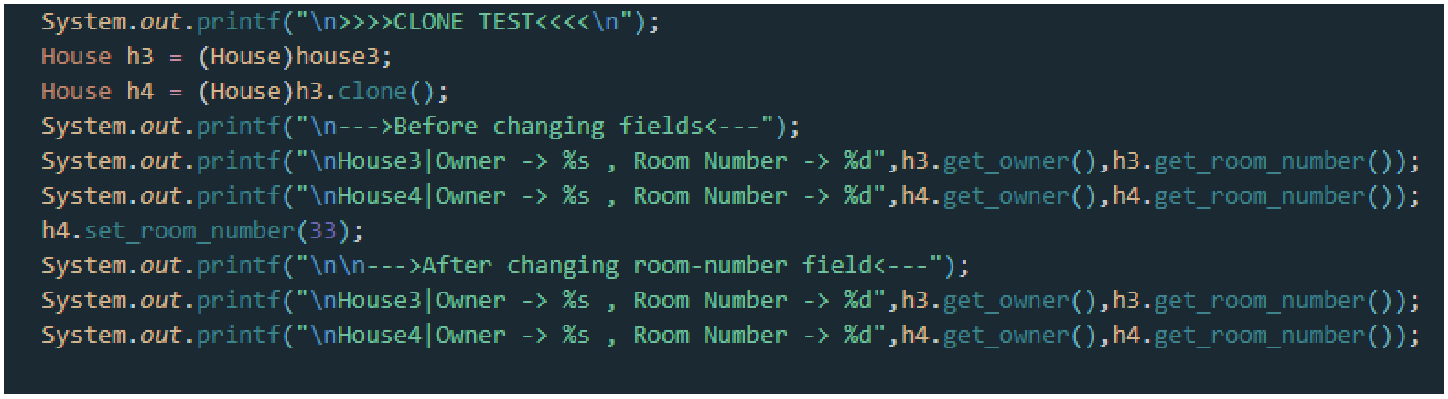
```java
Buildings playground1 = new Playgrounds(35,30);
street.add_building(playground1,'l');
street.view_mode(5);
```

Market2 valid

```java
Buildings market2 = new Market(55,20,10,"mike","09.00","22.00");
street.add_building(market2,'r');
street.view_mode(5);

street.delete_building(5,3);
street.delete_building(5,4);
street.view_mode(5);
```

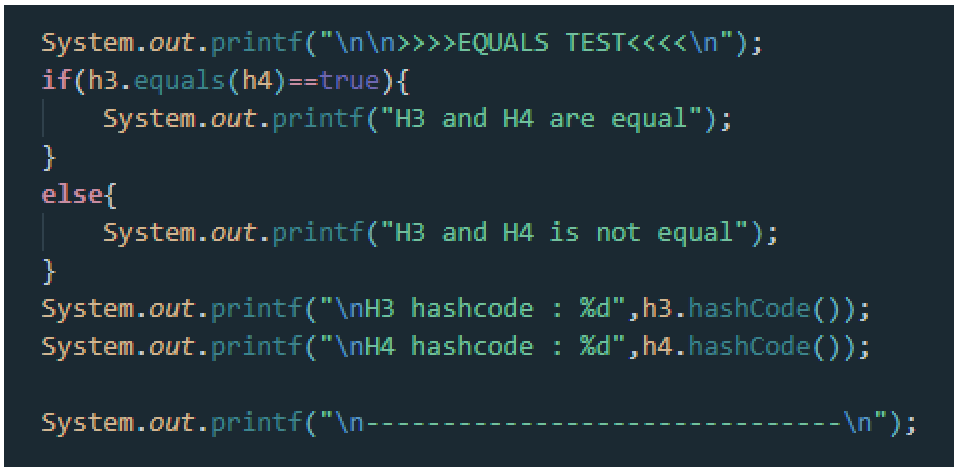
Deleting 2 buildings

Overriden Cloning House objects



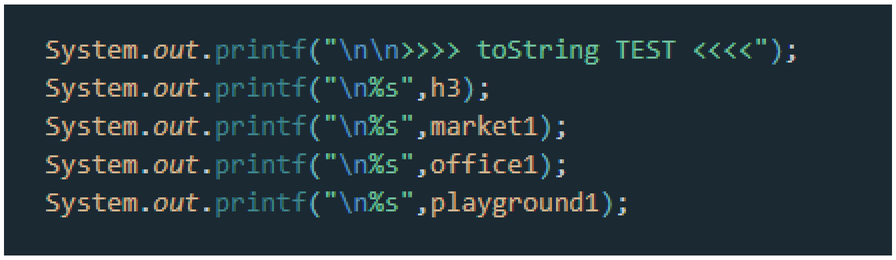
```
System.out.printf("\n>>>>CLONE TEST<<<<\n");
House h3 = (House)house3;
House h4 = (House)h3.clone();
System.out.printf("\n--->Before changing fields<---");
System.out.printf("\nHouse3|Owner -> %s , Room Number -> %d",h3.get_owner(),h3.get_room_number());
System.out.printf("\nHouse4|Owner -> %s , Room Number -> %d",h4.get_owner(),h4.get_room_number());
h4.set_room_number(33);
System.out.printf("\n\n--->After changing room-number field<---");
System.out.printf("\nHouse3|Owner -> %s , Room Number -> %d",h3.get_owner(),h3.get_room_number());
System.out.printf("\nHouse4|Owner -> %s , Room Number -> %d",h4.get_owner(),h4.get_room_number());
```

Overriden equals and hashcode method



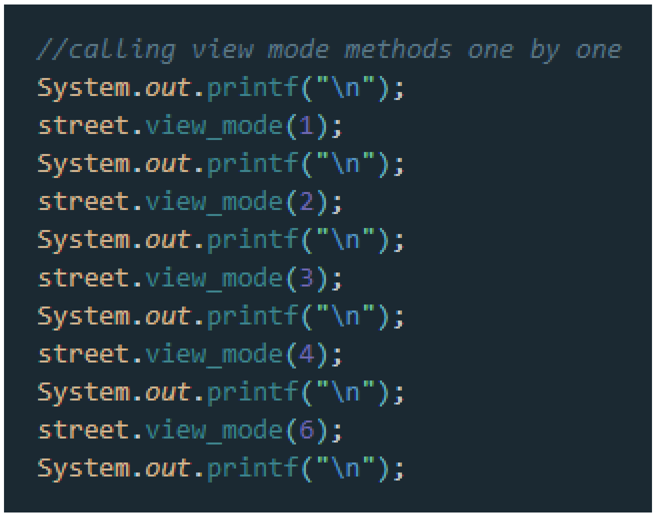
```
System.out.printf("\n\n>>>>EQUALS TEST<<<<\n");
if(h3.equals(h4)==true){
    System.out.printf("H3 and H4 are equal");
}
else{
    System.out.printf("H3 and H4 is not equal");
}
System.out.printf("\nH3 hashcode : %d",h3.hashCode());
System.out.printf("\nH4 hashcode : %d",h4.hashCode());

System.out.printf("\n-------------------------------\n");
```

Overriden toString method



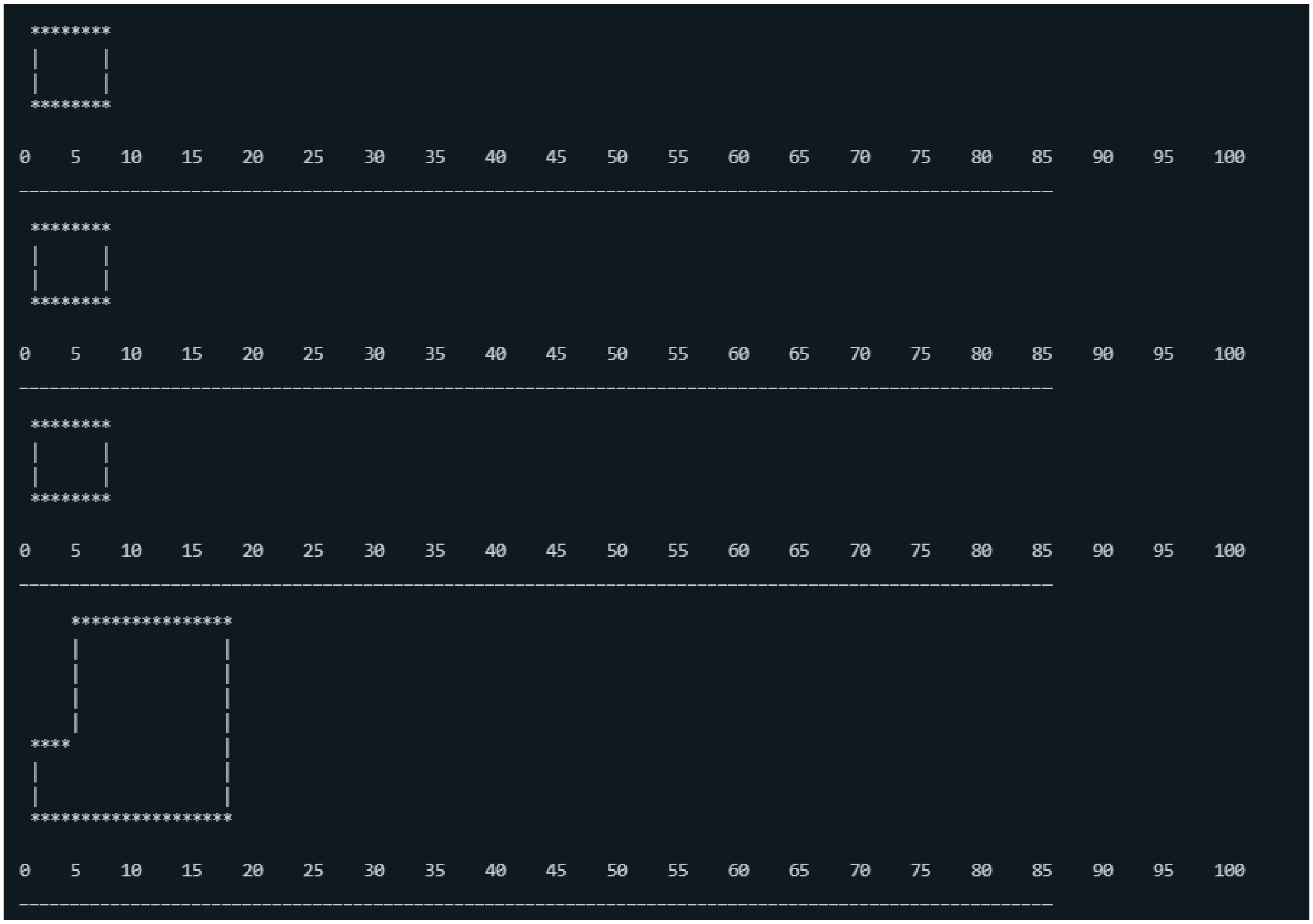
```
System.out.printf("\n\n>>>> toString TEST <<<<");
System.out.printf("\n%s",h3);
System.out.printf("\n%s",market1);
System.out.printf("\n%s",office1);
System.out.printf("\n%s",playground1);
```

Testing view mode methods

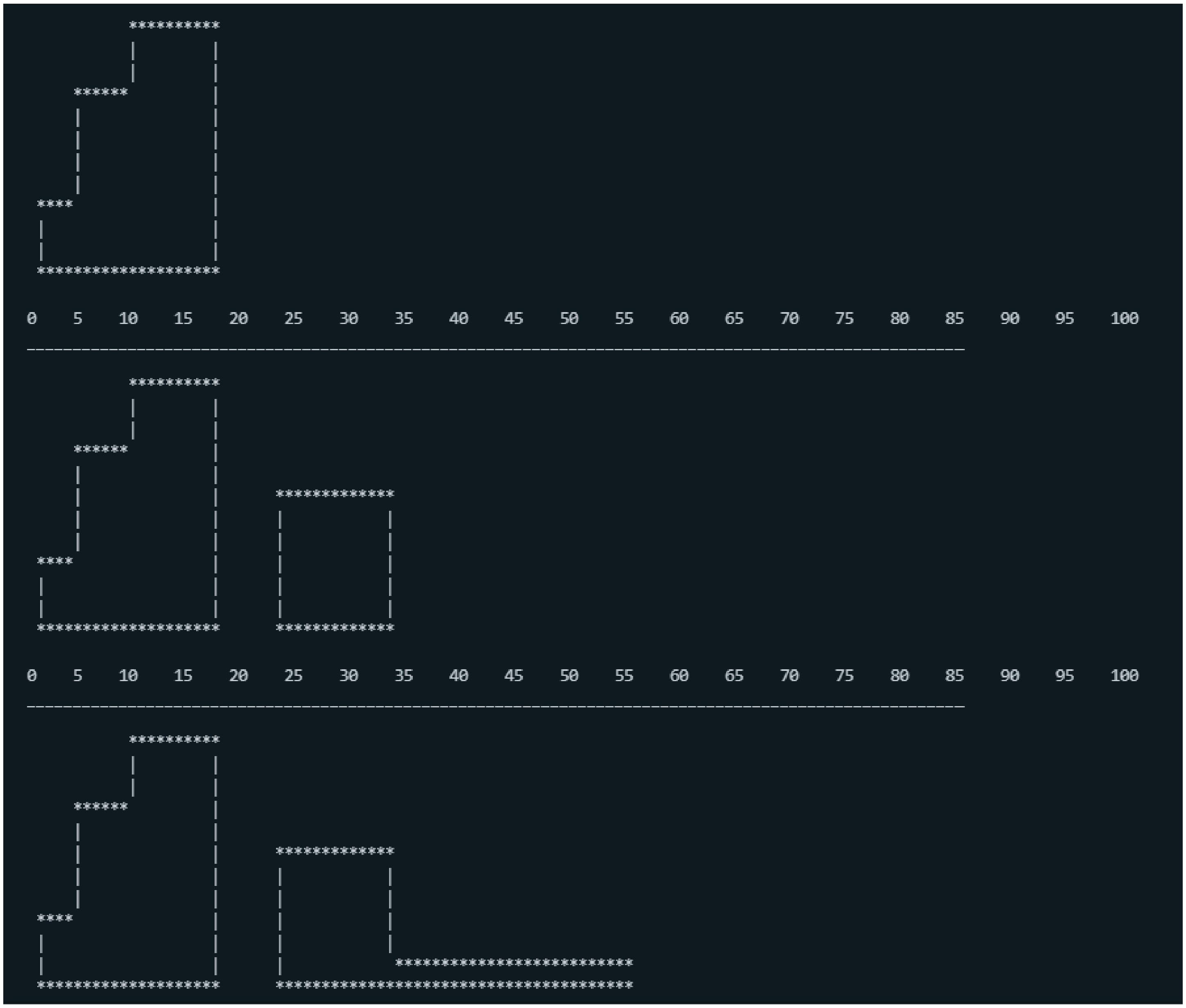


```java
//calling view mode methods one by one
System.out.printf("\n");
street.view_mode(1);
System.out.printf("\n");
street.view_mode(2);
System.out.printf("\n");
street.view_mode(3);
System.out.printf("\n");
street.view_mode(4);
System.out.printf("\n");
street.view_mode(6);
System.out.printf("\n");
```

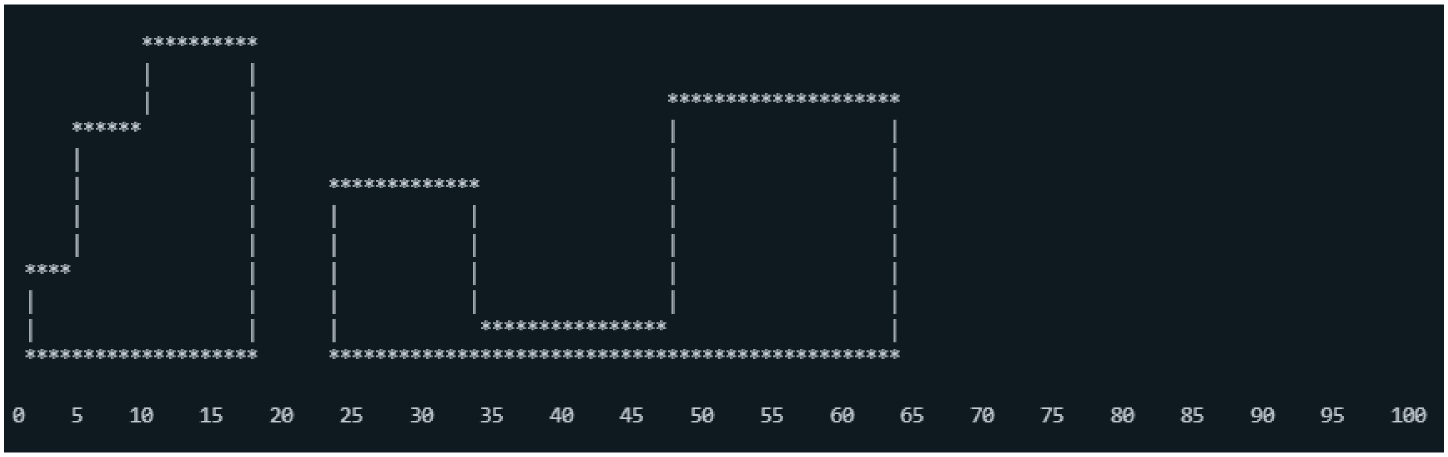
6 – Running and Results



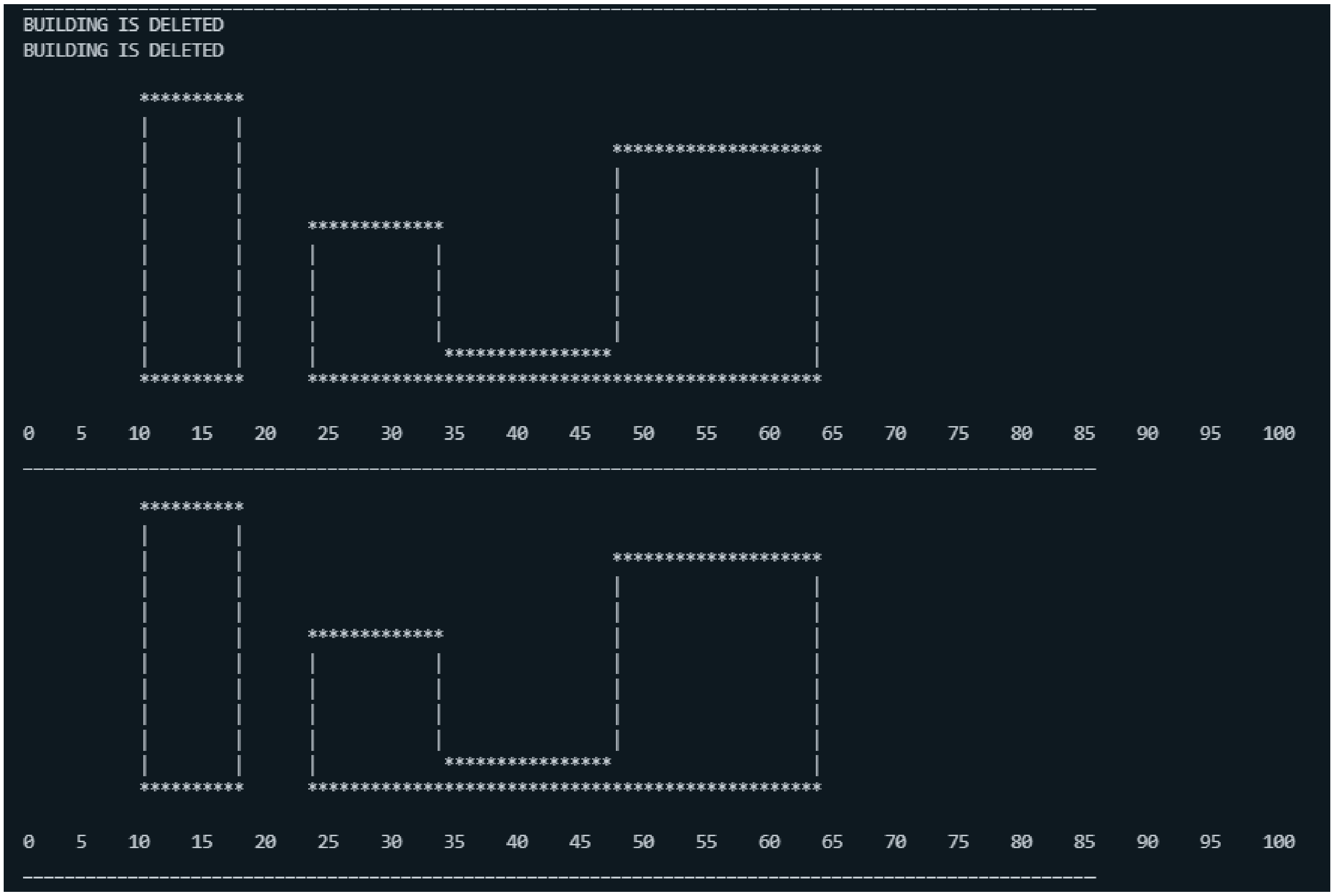
```
 ********
 |      |
 |      |
 ********

0      5      10     15     20     25     30     35     40     45     50     55     60     65     70     75     80     85     90     95     100
----------------------------------------------------------------------------------------------------------------------
 ********
 |      |
 |      |
 ********

0      5      10     15     20     25     30     35     40     45     50     55     60     65     70     75     80     85     90     95     100
----------------------------------------------------------------------------------------------------------------------
 ********
 |      |
 |      |
 ********

0      5      10     15     20     25     30     35     40     45     50     55     60     65     70     75     80     85     90     95     100
----------------------------------------------------------------------------------------------------------------------
     ****************
     |              |
     |              |
     |              |
     |              |
 ****               |
 |                  |
 |                  |
 ********************

0      5      10     15     20     25     30     35     40     45     50     55     60     65     70     75     80     85     90     95     100
----------------------------------------------------------------------------------------------------------------------
```

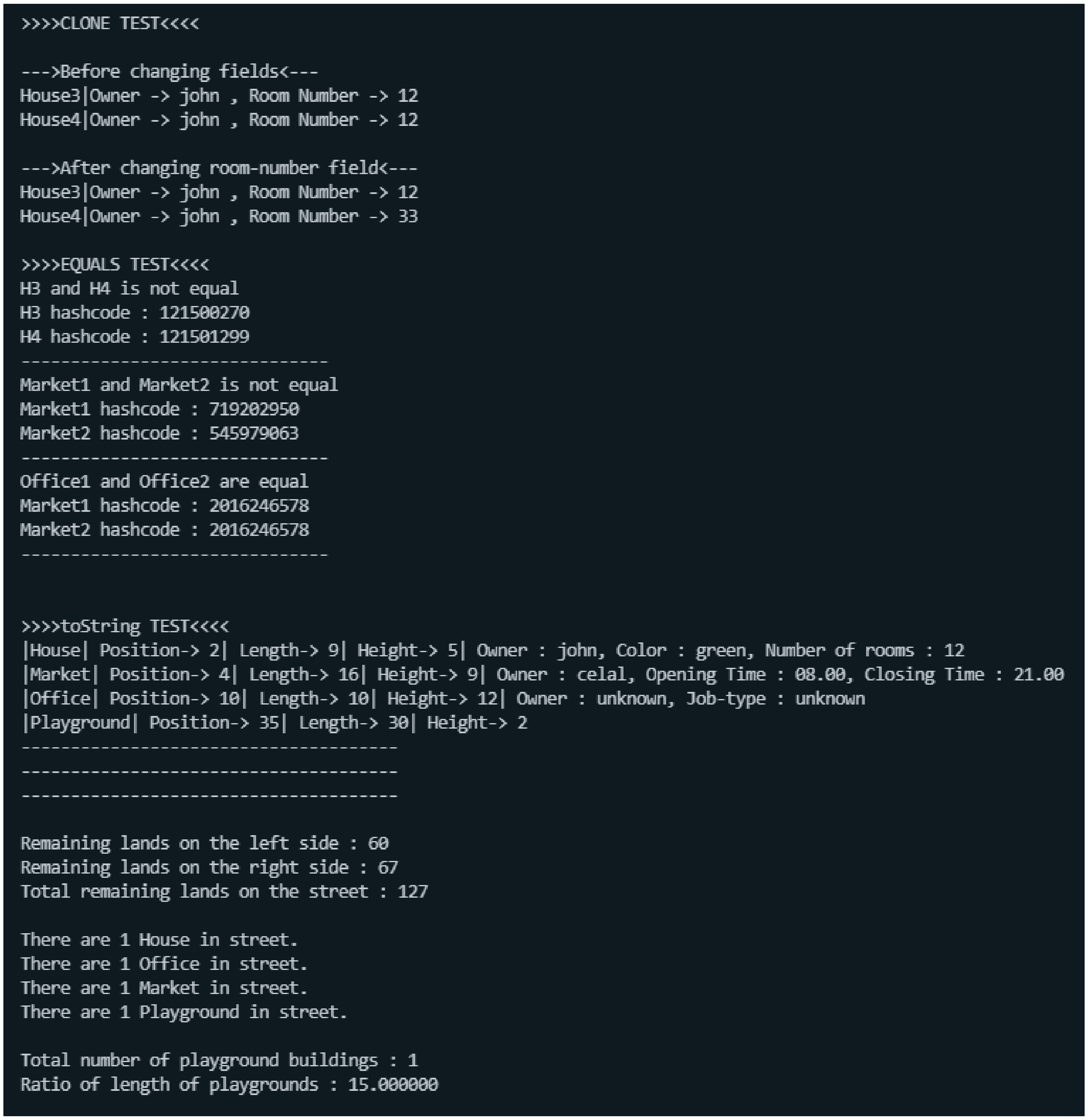


```
           **********
           |        |
           |        |
     ******         |
     |              |
     |              |
     |              |
     |              |
 ****               |
 |                  |
 |                  |
 ********************

0    5    10    15    20    25    30    35    40    45    50    55    60    65    70    75    80    85    90    95    100
-----------------------------------------------------------------------------------------------------
           **********
           |        |
           |        |
     ******         |
     |              |
     |              |      *************
     |              |      |           |
     |              |      |           |
 ****               |      |           |
 |                  |      |           |
 |                  |      |           |
 ********************      *************

0    5    10    15    20    25    30    35    40    45    50    55    60    65    70    75    80    85    90    95    100
-----------------------------------------------------------------------------------------------------
           **********
           |        |
           |        |
     ******         |
     |              |
     |              |      *************
     |              |      |           |
     |              |      |           |
 ****               |      |           |
 |                  |      |           |
 |                  |      |            **************************
 ********************      ***************************************
```

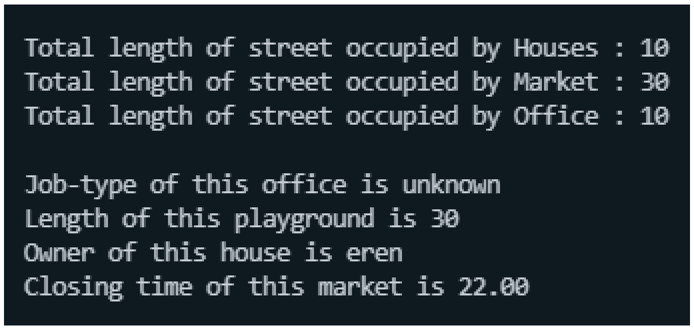


```
           **********
           |        |
           |        |                                   ********************
     ******         |                                   |                  |
     |              |                                   |                  |
     |              |      *************                |                  |
     |              |      |           |                |                  |
     |              |      |           |                |                  |
 ****               |      |           |                |                  |
 |                  |      |           |                |                  |
 |                  |      |            ****************                   |
 ********************      *************************************************

0    5    10    15    20    25    30    35    40    45    50    55    60    65    70    75    80    85    90    95    100
```


```
------------------------------------------------------------------------------------------------------
BUILDING IS DELETED
BUILDING IS DELETED

           **********
           |        |
           |        |                                   ********************
           |        |                                   |                  |
           |        |                                   |                  |
           |        |      *************                |                  |
           |        |      |           |                |                  |
           |        |      |           |                |                  |
           |        |      |           |                |                  |
           |        |      |           |                |                  |
           |        |      |            ****************                   |
           **********      *************************************************

0    5    10    15    20    25    30    35    40    45    50    55    60    65    70    75    80    85    90    95    100
------------------------------------------------------------------------------------------------------

           **********
           |        |
           |        |                                   ********************
           |        |                                   |                  |
           |        |                                   |                  |
           |        |      *************                |                  |
           |        |      |           |                |                  |
           |        |      |           |                |                  |
           |        |      |           |                |                  |
           |        |      |           |                |                  |
           |        |      |            ****************                   |
           **********      *************************************************

0    5    10    15    20    25    30    35    40    45    50    55    60    65    70    75    80    85    90    95    100
------------------------------------------------------------------------------------------------------
```



```
>>>>CLONE TEST<<<<

--->Before changing fields<---
House3|Owner -> john , Room Number -> 12
House4|Owner -> john , Room Number -> 12

--->After changing room-number field<---
House3|Owner -> john , Room Number -> 12
House4|Owner -> john , Room Number -> 33

>>>>EQUALS TEST<<<<
H3 and H4 is not equal
H3 hashcode : 121500270
H4 hashcode : 121501299
-------------------------------
Market1 and Market2 is not equal
Market1 hashcode : 719202950
Market2 hashcode : 545979063
-------------------------------
Office1 and Office2 are equal
Market1 hashcode : 2016246578
Market2 hashcode : 2016246578
-------------------------------


>>>>toString TEST<<<<
|House| Position-> 2| Length-> 9| Height-> 5| Owner : john, Color : green, Number of rooms : 12
|Market| Position-> 4| Length-> 16| Height-> 9| Owner : celal, Opening Time : 08.00, Closing Time : 21.00
|Office| Position-> 10| Length-> 10| Height-> 12| Owner : unknown, Job-type : unknown
|Playground| Position-> 35| Length-> 30| Height-> 2
--------------------------------------
--------------------------------------
--------------------------------------

Remaining lands on the left side : 60
Remaining lands on the right side : 67
Total remaining lands on the street : 127

There are 1 House in street.
There are 1 Office in street.
There are 1 Market in street.
There are 1 Playground in street.

Total number of playground buildings : 1
Ratio of length of playgrounds : 15.000000
```



```
Total length of street occupied by Houses : 10
Total length of street occupied by Market : 30
Total length of street occupied by Office : 10

Job-type of this office is unknown
Length of this playground is 30
Owner of this house is eren
Closing time of this market is 22.00
```