

Ownable.java

```

1 package game.fields;
2
3 import game.Player;
4
5
6 public abstract class Ownable extends Field
7 {
8
9     private static final int PAWN_FACTOR = 2;
10    protected int price;
11    protected Player owner;
12    protected boolean pawned;
13
14    // Constructor
15    protected Ownable(String title, int price)
16    {
17        super(title);
18        this.price = price;
19        this.owner = null;
20        this.pawned = false;
21    }
22
23    //TODO move to Ownable
24    public boolean isPawned() {
25
26        return pawned ;
27    }
28    //TODO move to Ownable
29    public void setPawned(boolean isPawned){
30        this.pawned = isPawned;
31    }
32    @Override
33    public String[] decoratorMessage(Player player){
34        String[] first = super.decoratorMessage(player);
35        String[] second = null;
36        if (getOwner() == null){
37            second = new String[] { "PriceMessage",
Integer.toString(getPrice())};
38        }
39        else if (getOwner() != player)
40        {
41            second = new String[] { "YouHaveToPayMessage",
Integer.toString(getRent()) };
42        } else if (getOwner() == player){
43            second = new String[] { "OwnFieldMessage"};
44        }
45        return StringTools.add(first, second);
46    }
47
48
49    public void setOwner(Player owner)

```

Ownable.java

```
50     {
51         this.owner = owner;
52     }
53
54     public Player getOwner()
55     {
56         return owner;
57     }
58
59     public abstract int getBaseRent();
60
61     public abstract int getRent();
62
63     public abstract String toString();
64
65     public int getPrice()
66     {
67         return price;
68     }
69
70     public int getPawnValue() {
71         return price/PAWN_FACTOR;
72     }
73 }
74
```