

TaxpayerAccountsManager.java

```

1 package accountsManager;
2
3 import java.util.ArrayList;
4
5
6
7
8
9 public class TaxpayerAccountsManager {
10     ArrayList<Taxpayer> taxpayersList = new ArrayList<Taxpayer>();
11     InputManager inputManager = new InputManager();
12     OutputManager outputManager = new OutputManager();
13
14     public void deleteTaxpayer(Taxpayer taxpayerForDelete) {
15         taxpayersList.remove(taxpayerForDelete);
16     }
17
18     public void deleteTaxpayer(int Afm) {
19         for(int i=0; i<taxpayersList.size(); i++){
20             if (taxpayersList.get(i).getAfm() == Afm) {
21                 taxpayersList.remove(i);
22             }
23             break;
24         }
25     }
26
27     public void deleteReceiptFromTaxpayer(Taxpayer taxpayerNeedingToDeleteReceipt,
28     int receiptCode) {
29         ArrayList<Receipt> receipts = taxpayerNeedingToDeleteReceipt.getReceipts();
30         for(int i=0; i<receipts.size(); i++){
31             if(receipts.get(i).getReceiptCode() == receiptCode){
32                 receipts.remove(i);
33                 return;
34             }
35             System.out.println("!!!Receipt don't found!!!");
36         }
37     }
38
39     public void insertNewReceiptForTaxpayer(Taxpayer taxpayerNeedingNewReceipt,
40     String category, int receiptCode, int dateOfIssue, int amountPaid, String
41     companyName, String country, String city, String street, int streetNumber){
42         Receipt newReceipt = new Receipt();
43         ArrayList<Receipt> receipts = taxpayerNeedingNewReceipt.getReceipts();
44         newReceipt.initializeReceipt(category, receiptCode, dateOfIssue,
45         amountPaid, companyName, country, city, street, streetNumber);
46         receipts.add(newReceipt);
47         taxpayerNeedingNewReceipt.setReceipts(receipts);
48     }
49
50     public void printTaxpayer(Taxpayer taxpayerForPrint) {
51         taxpayerForPrint.toString();
52     }
53
54     public Taxpayer getTaxpayer(String taxpayersName) {
55         for(int i=0; i<taxpayersList.size(); i++){
56             if(taxpayersList.get(i).getName().equals(taxpayersName)){
57                 return taxpayersList.get(i);
58             }
59         }
60         System.out.println("Error: Can't find taxpayer !!!");
61         return null;
62     }
63
64     public Taxpayer getTaxpayer(int taxpayerAfm) {
65         for(int i=0; i<taxpayersList.size(); i++){
66             if(taxpayersList.get(i).getAfm() == taxpayerAfm) {
67                 return taxpayersList.get(i);
68             }
69         }
70     }
71 }

```

TaxpayerAccountsManager.java

```

65     }
66     System.out.println("Error: Cant't find taxpayer !!!");
67     return null;
68 }
69
70 public void insertTaxpayerFromFile(String filePath){
71     if(filePath.endsWith(".txt")){
72         inputManager.creatTaxpayerDataListFromTextFile(filePath);
73     }else if(filePath.endsWith(".xml")){
74         inputManager.creatTaxpayerDataListFromXmlFile(filePath);
75     }
76     taxpayersList.add( inputManager.getTaxpayerFromFileDataList() );
77 }
78
79 public void creatInfoXmlFile(Taxpayer taxp){
80     outputManager.creatInfoXmlFile(taxp);
81 }
82
83 public void creatLogXmlFile(Taxpayer taxp){
84     outputManager.creatLogXmlFile(taxp);
85 }
86
87 public void creatInfoTextFile(Taxpayer taxp){
88     outputManager.creatInfoTextFile(taxp);
89 }
90
91 public void creatLogTextFile(Taxpayer taxp){
92     outputManager.creatLogTestFile(taxp);
93 }
94
95 public void creatBarChart(Taxpayer taxp){
96     outputManager.creatBarChart(taxp);
97 }
98
99 public void creatPieChart(Taxpayer taxp){
100     outputManager.creatPieChart(taxp);
101 }
102
103 public void setDomain(String domain){
104     outputManager.setDomain(domain);
105 }
106 }
107

```