```java
package accountsManager;

import java.util.ArrayList;

public class TaxpayerAccountsManager {
    ArrayList<Taxpayer> taxpayersList = new ArrayList<Taxpayer>();
    InputManager inputManager = new InputManager();
    OutputManager outputManager = new OutputManager();

    public void deleteTaxpayer(Taxpayer taxpayerForDelete){
        taxpayersList.remove(taxpayerForDelete);
    }

    public void deleteTaxpayer(int Afm){
        for(int i=0; i<taxpayersList.size(); i++){
            if (taxpayersList.get(i).getAfm() == Afm){
                taxpayersList.remove(i);
            }
            break;
        }
    }

    public void deleteReceiptFromTaxpayer(Taxpayer taxpayerNeedingToDeleteReceipt, int receiptCode){
        ArrayList<Receipt> receipts = taxpayerNeedingToDeleteReceipt.getReceipts();
        for(int i=0; i<receipts.size(); i++){
            if(receipts.get(i).getReceiptCode() == receiptCode){
                receipts.remove(i);
                return;
            }
        }
        System.out.println("!!!Receipt don't found!!!");
    }

    public void insertNewReceiptForTaxpayer(Taxpayer taxpayerNeedingNewReceipt, String category, int receiptCode, int dateOfIssue, int amountPaid, String companyName, String country, String city, String street, int streetNumber){
        Receipt newReceipt = new Receipt();
        ArrayList<Receipt> receipts = taxpayerNeedingNewReceipt.getReceipts();
        newReceipt.initializeReceipt(category, receiptCode, dateOfIssue, amountPaid, companyName, country, city, street, streetNumber);
        receipts.add(newReceipt);
        taxpayerNeedingNewReceipt.setReceipts(receipts);
    }

    public void printTaxpayer(Taxpayer taxpayerForPrint){
        taxpayerForPrint.toString();
    }

    public Taxpayer getTaxpayer(String taxpayersName){
        for(int i=0; i<taxpayersList.size(); i++){
            if(taxpayersList.get(i).getName().equals(taxpayersName)){
                return taxpayersList.get(i);
            }
        }
        System.out.println("Error: Can't find taxpayer !!!");
        return null;
    }

    public Taxpayer getTaxpayer(int taxpayerAfm){
        for(int i=0;i<taxpayersList.size(); i++){
            if(taxpayersList.get(i).getAfm() == taxpayerAfm){
                return taxpayersList.get(i);
            }
        }
        System.out.println("Error: Cant't find taxpayer !!!");
        return null;
    }

    public void insertTaxpayerFromFile(String filePath){
        if(filePath.endsWith(".txt")){
            inputManager.creatTaxpayerDataListFromTextFile(filePath);
        }else if(filePath.endsWith(".xml")){
            inputManager.creatTaxpayerDataListFromXmlFile(filePath);
        }
        taxpayersList.add( inputManager.getTaxpayerFromFileDataList() );
    }

    public void creatInfoXmlFile(Taxpayer taxp){
        outputManager.creatInfoXmlFile(taxp);
    }

    public void creatLogXmlFile(Taxpayer taxp){
        outputManager.creatLogXmlFile(taxp);
    }

    public void creatInfoTextFile(Taxpayer taxp){
        outputManager.creatInfoTextFile(taxp);
    }

    public void creatLogTextFile(Taxpayer taxp){
        outputManager.creatLogTestFile(taxp);
    }

    public void creatBarChart(Taxpayer taxp){
        outputManager.creatBarChart(taxp);
    }

    public void creatPieChart(Taxpayer taxp){
        outputManager.creatPieChart(taxp);
    }

    public void setDomain(String domain){
        outputManager.setDomain(domain);
    }
}

```