

```
package ec.edu.espe.urbanizationtreasury.view;

import ec.edu.espe.urbanizationtreasury.model.Resident;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.InputMismatchException;
import java.util.Scanner;
import java.util.ArrayList;

/**
 *
 * @author WebMasterTeam, DCCO-ESPE
 */
public class UrbanizationTreasurySystem {

    public static void main(String[] args) {

        int option; //For options menu
        boolean exit = false; //Exit menu
        int[] residentNumber = new int[1]; //position in the array

        ArrayList<Resident> residents = new ArrayList();

        Scanner sc = new Scanner(System.in);

        System.out.println(" ---- National Polytechnic School Urbanization
        System.out.println(" ---- Treasury System ---- ");

        while (!exit) {

            System.out.println("=====");
            System.out.println("#####Residents and Payments#####");
            System.out.println("1. Enter Resident");
            System.out.println("2. Aliquot");
            System.out.println("3. Access Card");
            System.out.println("4. Access Control");
            System.out.println("5. Extraordinary");
            System.out.println("6. Archive the residents");
            System.out.println("7. Exit");
            System.out.println("=====");
            try {
```

----- ") ;

```

System.out.print("Enter an option: ");
option = sc.nextInt();

switch (option) {
    case 1 -> {
        enterResident(residents, sc); //Enter Residents
        residentNumber[0]++;
    }
    case 2 -> {

        if (residentNumber[0] != 0) {
            System.out.println("-----");
            int returnMenu = 2;
            do {
                System.out.println("1. Pay");
                System.out.println("2. History");
                System.out.println("3. Print bill");
                System.out.println("4. Exit");
                try {

                    System.out.print("Enter an option: ");
                    option = sc.nextInt();

                    switch (option) {
                        case 1 -> {
                            int searchId;
                            int searchIdFound = 0;
                            if (residentNumber[0] != 0) {
                                System.out.println("Enter r
                                searchId = sc.nextInt();

                                for (int i = 0; i < residen
                                    if (residents.get(i).ge
                                        System.out.println(
                                            enterAliquots(resid
                                                searchIdFound++;
                                        }
                                    }
                                if (searchIdFound == 0) {
                                    System.out.println("Res
                                }
                            } else {
                                System.out.println("Residen
                            }
                        }
                    }
                }
            } while (returnMenu != 4);
        }
    }
}

```

-----");

esident id: ");

```
ts.size(); i++) {  
    tDni() == searchId) {  
        "=====  
Resident " + searchId + " =====");  
        ents.get(i));
```

ident DNI not found");

t no registrated");

```
    }  
    case 2 -> {  
        int searchId;  
        int searchIdFound = 0;  
        if (residentNumber[0] != 0) {  
            System.out.println("Enter r  
            searchId = sc.nextInt();  
  
            for (int i = 0; i < residen  
                if (residents.get(i).ge  
                    System.out.println(  
                        printHistory(reside  
                            searchIdFound++;  
                }  
            }  
            if (searchIdFound == 0) {  
                System.out.println("Res  
            }  
        } else {  
            System.out.println("Residen  
        }  
    }  
    case 3 -> {  
        int searchId;  
        int searchIdFound = 0;  
        if (residentNumber[0] != 0) {  
            System.out.println("Enter r  
            searchId = sc.nextInt();  
  
            for (int i = 0; i < residen  
                if (residents.get(i).ge  
                    System.out.println(  
                        printBill(residents  
                            searchIdFound++;  
                }  
            }  
            if (searchIdFound == 0) {  
                System.out.println("Res  
            }  
        } else {  
            System.out.println("Residen  
        }  
    }  
    case 4 -> {
```

```
resident DNI for most the history: ");

    for (int i = 0; i < residents.size(); i++) {
        if (residents.get(i).getDni() == searchId) {
            "==== Resident " + searchId + " =====");
            residents.get(i), residents.get(i).getAliquot());
        }
    }

    if (searchId == null) {
        System.out.println("Resident DNI not found");
    }

    if (searchId == 0) {
        System.out.println("No registered");
    }

    resident DNI for print bill: ");

    for (int i = 0; i < residents.size(); i++) {
        if (residents.get(i).getDni() == searchId) {
            "==== Resident " + searchId + " =====");
            residents.get(i), residents.get(i).getAliquot());
        }
    }

    if (searchId == null) {
        System.out.println("Resident DNI not found");
    }

    if (searchId == 0) {
        System.out.println("No registered");
    }
}
```

```

        System.out.println("Do you want
        returnMenu = sc.nextInt();

    }
    default ->
        System.out.println("Option inva
    }

    } catch (InputMismatchException e) {
        System.out.println("Opcion invalid");
        sc.next();
    }
    } while (returnMenu != 1);
    System.out.println("-----
} else {
    System.out.println("|||No residents registered|
}

}

case 3 -> {
    if (residentNumber[0] != 0) {
        System.out.println("-----
        int returnMenu = 2;
        do {
            System.out.println("1. Pay");
            System.out.println("2. History");
            System.out.println("3. Print bill");
            System.out.println("4. Exit");
            try {

                System.out.print("Enter an option: ");
                option = sc.nextInt();

                switch (option) {
                    case 1 -> {
                        int searchId;
                        int searchIdFound = 0;
                        if (residentNumber[0] != 0) {
                            System.out.println("Enter r
                            searchId = sc.nextInt();

                            for (int i = 0; i < residen
                                if (residents.get(i).ge

```

```
go out? Yes(1) No(0)");
```

```
lid ");
```

```
-----");
```

```
||");
```

```
-----");
```

```
esident id: ");
```

```
ts.size(); i++) {  
tDni() == searchId) {
```



```
        System.out.println(
            enterAccessCard(res
            searchIdFound++;
        }
    }
    if (searchIdFound == 0) {
        System.out.println("Res
    }
} else {
    System.out.println("Residen
}
}
case 2 -> {
    int searchId;
    int searchIdFound = 0;
    if (residentNumber[0] != 0) {
        System.out.println("Enter r
        searchId = sc.nextInt();

        for (int i = 0; i < residen
            if (residents.get(i).ge
                System.out.println(
                    printHistory(reside
                    searchIdFound++;
            }
        }
        if (searchIdFound == 0) {
            System.out.println("Res
        }
    } else {
        System.out.println("Residen
    }
}
case 3 -> {
    int searchId;
    int searchIdFound = 0;
    if (residentNumber[0] != 0) {
        System.out.println("Enter r
        searchId = sc.nextInt();

        for (int i = 0; i < residen
            if (residents.get(i).ge
                System.out.println(
                    printBill(residents
```

```
"===== Resident " + searchId + " =====");
idents.get(i));

ident DNI not found");

t no registrated");

resident DNI for most the history: ");

ts.size(); i++) {
tDni() == searchId) {
"===== Resident " + searchId + " =====");
nts.get(i), residents.get(i).getAccessCard());

ident DNI not found");

t no registrated");

resident DNI for print bill: ");

ts.size(); i++) {
tDni() == searchId) {
"===== Resident " + searchId + " =====");
.get(i), residents.get(i).getAccessCard());
```

```

                searchIdFound++;
            }
        }
        if (searchIdFound == 0) {
            System.out.println("Res
        }
    } else {
        System.out.println("Residen
    }
}
case 4 -> {
    System.out.println("Do you want
    returnMenu = sc.nextInt();

}
default ->
    System.out.println("Option inva
}

} catch (InputMismatchException e) {
    System.out.println("Opcion invalid");
    sc.next();
}
} while (returnMenu != 1);
System.out.println("-----
} else {
    System.out.println("|||No residents registered|
}
}
case 4 -> {

    if (residentNumber[0] != 0) {
        System.out.println("-----
        int returnMenu = 2;
        do {
            System.out.println("1. Pay");
            System.out.println("2. History");
            System.out.println("3. Print bill");
            System.out.println("4. Exit");
            try {

                System.out.print("Enter an option: ");
                option = sc.nextInt();

```

```
ident DNI not found");
```

```
t no registrated");
```

```
go out? Yes(1) No(0)");
```

```
lid ");
```

```
-----");
```

```
||");
```

```
-----");
```

```
switch (option) {
    case 1 -> {
        int searchId;
        int searchIdFound = 0;
        if (residentNumber[0] != 0) {
            System.out.println("Enter resident number to search for:");
            searchId = sc.nextInt();

            for (int i = 0; i < residents.size(); i++) {
                if (residents.get(i).getResidentNumber() == searchId) {
                    System.out.println("Resident found. Access control:");
                    enterAccessControl(residents.get(i));
                    searchIdFound++;
                }
            }
            if (searchIdFound == 0) {
                System.out.println("Resident not found.");
            }
        } else {
            System.out.println("Resident number cannot be zero.");
        }
    }
    case 2 -> {
        int searchId;
        int searchIdFound = 0;
        if (residentNumber[0] != 0) {
            System.out.println("Enter resident number to search for:");
            searchId = sc.nextInt();

            for (int i = 0; i < residents.size(); i++) {
                if (residents.get(i).getResidentNumber() == searchId) {
                    System.out.println("Resident found. History:");
                    printHistory(residents.get(i));
                    searchIdFound++;
                }
            }
            if (searchIdFound == 0) {
                System.out.println("Resident not found.");
            }
        } else {
            System.out.println("Resident number cannot be zero.");
        }
    }
    case 3 -> {
```

```
resident id: ");

for (int i = 0; i < residents.size(); i++) {
    if (residents.get(i).getDni() == searchId) {
        System.out.println("==== Resident " + searchId + " =====");
        System.out.println(residents.get(i));
    }
}

if (searchId != null) {
    System.out.println("Resident DNI not found");
} else {
    System.out.println("No registered");
}

// For history
System.out.println("Resident DNI for most the history: ");

for (int i = 0; i < residents.size(); i++) {
    if (residents.get(i).getDni() == searchId) {
        System.out.println("==== Resident " + searchId + " =====");
        System.out.println(residents.get(i), residents.get(i).getAccessControl());
    }
}

if (searchId != null) {
    System.out.println("Resident DNI not found");
} else {
    System.out.println("No registered");
}
```

```

        int searchId;
        int searchIdFound = 0;
        if (residentNumber[0] != 0) {
            System.out.println("Enter r
            searchId = sc.nextInt();

            for (int i = 0; i < residen
                if (residents.get(i).ge
                    System.out.println(
                        printBill(residents
                            searchIdFound++;
                }
            }
            if (searchIdFound == 0) {
                System.out.println("Res
            }
        } else {
            System.out.println("Residen
        }
    }
    case 4 -> {
        System.out.println("Do you want
        returnMenu = sc.nextInt();

    }
    default ->
        System.out.println("Option inva
    }

    } catch (InputMismatchException e) {
        System.out.println("Opcion invalid");
        sc.next();
    }
    } while (returnMenu != 1);
    System.out.println("-----
} else {
    System.out.println("|||No residents registered|
}

}
case 5 -> {

    if (residentNumber[0] != 0) {
        System.out.println("-----

```

```
resident DNI for print bill: ");

ts.size(); i++) {
    tDni() == searchId) {
        "===== Resident " + searchId + " =====");
        .get(i), residents.get(i).getAccessControl());

    ident DNI not found");

    t no registrated");

    go out? Yes(1) No(0)");

    lid ");

    -----");

    | |");

    -----");
```



```

int returnMenu = 2;
do {
    System.out.println("1. Pay");
    System.out.println("2. History");
    System.out.println("3. Print bill");
    System.out.println("4. Exit");
    try {

        System.out.print("Enter an option: ");
        option = sc.nextInt();

        switch (option) {
            case 1 -> {
                int searchId;
                int searchIdFound = 0;
                if (residentNumber[0] != 0) {
                    System.out.println("Enter r
                    searchId = sc.nextInt();

                    for (int i = 0; i < residen
                    if (residents.get(i).ge
                    System.out.println(
                    enterExtraordinary(
                    searchIdFound++;
                }
            }
            if (searchIdFound == 0) {
                System.out.println("Res
            }
        } else {
            System.out.println("Residen
        }
    }
    case 2 -> {
        int searchId;
        int searchIdFound = 0;
        if (residentNumber[0] != 0) {
            System.out.println("Enter r
            searchId = sc.nextInt();

            for (int i = 0; i < residen
            if (residents.get(i).ge
            System.out.println(
            printHistory(reside

```

```
resident id: ");
```

```
ts.size(); i++) {  
    tDni() == searchId) {  
        "===== Resident " + searchId + " =====");  
        residents.get(i));
```

```
ident DNI not found");
```

```
t no registrated");
```

```
resident DNI for most the history: ");
```

```
ts.size(); i++) {  
    tDni() == searchId) {  
        "===== Resident " + searchId + " =====");  
        nts.get(i), residents.get(i).getExtraordinary());
```

```

                searchIdFound++;
            }
        }
        if (searchIdFound == 0) {
            System.out.println("Res
        }
    } else {
        System.out.println("Residen
    }
}
case 3 -> {
    int searchId;
    int searchIdFound = 0;
    if (residentNumber[0] != 0) {
        System.out.println("Enter r
        searchId = sc.nextInt();

        for (int i = 0; i < residen
            if (residents.get(i).ge
                System.out.println(
                    printBill(residents
                        searchIdFound++;
                    }
                }
            if (searchIdFound == 0) {
                System.out.println("Res
            }
        } else {
            System.out.println("Residen
        }
    }
case 4 -> {
    System.out.println("Do you want
    returnMenu = sc.nextInt();

}
default ->
    System.out.println("Option inva
}

} catch (InputMismatchException e) {
    System.out.println("Opcion invalid");
    sc.next();
}

```

```
ident DNI not found");
```

```
t no registrated");
```

```
esident DNI for print bill: ");
```

```
ts.size(); i++) {  
tDni() == searchId) {  
"===== Resident " + searchId + " =====");  
.get(i), residents.get(i).getExtraordinary());
```

```
ident DNI not found");
```

```
t no registrated");
```

```
go out? Yes(1) No(0)");
```

```
lid ");
```

```

        } while (returnMenu != 1);
        System.out.println("-----");
    } else {
        System.out.println("|||No residents registered|");
    }
}
case 6 -> {
    int saveChickenId;
    int saveIdFound = 0;
    if (residentNumber[0] != 0)
    {

        System.out.println("What recident do you want i");
        System.out.print("Resident DNI: ");
        saveChickenId = sc.nextInt();

        for (int i = 0; i < residents.size(); i++) {
            if (residents.get(i).getDni() == saveChicke
                ArchiveListResidents(residents.get(i));
                System.out.println("Resident Registered");
                System.out.println("");
                saveIdFound++;
            }
        }
        if (saveIdFound == 0) {
            System.out.println("Resident DNI not found")
        }
    } else {
        System.out.println("Resident no registrated");
    }
}
case 7 -> {
    System.out.println("=====");
    System.out.println("== Thanks for using the system");
    System.out.println("=====");
    exit = true;
}
default ->
    System.out.println("Option invalid ");
}
} catch (InputMismatchException e) {
    System.out.println("You need to enter a number");
    sc.next();
}
}

```

```
-----");
```

```
||");
```

```
include in the file?");
```

```
hId) {
```

```
");
```

```
);
```

```
=="");
```

```
=="");
```

```
=="");
```

```

    }
}

private static void enterResident(ArrayList<Resident> residents, Scanner
    long validation, repeatDni;

    Resident resident = new Resident();
    System.out.print("Resident Id:");
    validation = sc.nextLong();
    resident.setDni(validation);
    String dniValidation = String.valueOf(validation);
    repeatDni = dniValidation(dniValidation);

    if(repeatDni == 0){
        while(repeatDni == 0){
            System.out.println("Re-enter the DNI:");
            validation = sc.nextLong();
            resident.setDni(validation);
            String dniValidate = String.valueOf(validation);
            repeatDni = dniValidation(dniValidate);
        }
    }
    System.out.print("Enter resident name: ");
    resident.setName(sc.next());
    System.out.print("Enter batch number: ");
    resident.setBatchNumber(sc.nextInt());
    residents.add(resident);
    System.out.println("|||User entered |||");
}

private static int dniValidation(String dniValidation) {

    int number = 0, suma = 0, resultado = 0, result = 0;

    for (int i = 0; i < dniValidation.length(); i++) {
        number = Integer.parseInt(String.valueOf(dniValidation.charAt(i)
        if (i % 2 == 0) {
            number = number * 2;
            if (number > 9) {
                number = number - 9;
            }
        }
    }
}

```

```
r sc) {
```

```
)) );
```



```

        }
    }
    suma = suma + number;
}
if (suma % 10 != 0) {
    resultado = 10 - (suma % 10);
    if (resultado == number) {
        System.out.println("Valid DNI");
    } else if (resultado != number) {
        System.out.println("Invalid DNI");
    }
} else {
    System.out.println("Valid DNI");
    result = 1;
}
return (result);
}

private static void enterAliquots(Resident residents) {

    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the Mounth: ");
    residents.setMounths(sc.next());
    System.out.println("Enter the payment: ");
    residents.setAliquot(sc.nextFloat());

}

private static void enterAccessCard(Resident residents) {

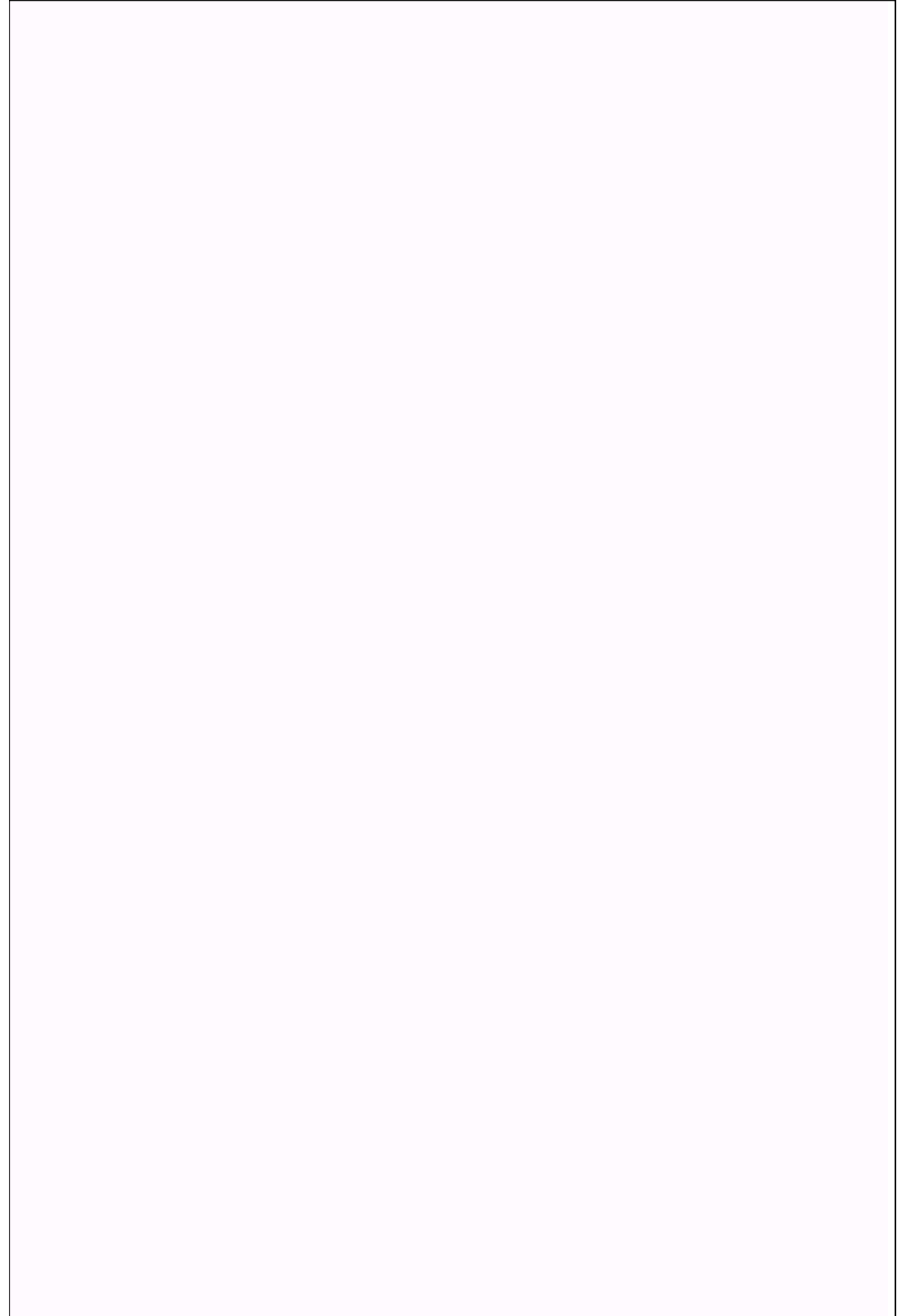
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the Mounth: ");
    residents.setMounths(sc.next());
    System.out.println("Enter the payment: ");
    residents.setAccessCard(sc.nextFloat());

}

private static void enterAccessControl(Resident residents) {

    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the Mounth: ");
    residents.setMounths(sc.next());
    System.out.println("Enter the payment: ");

```



```

        residents.setAccessControl(sc.nextFloat());

    }

    private static void enterExtraordinary(Resident residents) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the Mounth: ");
        residents.setMounths(sc.next());
        System.out.println("Enter the paymentt: ");
        residents.setExtraordinary(sc.nextFloat());

    }

    //Los demas enters
    private static void printHistory(Resident residents, float pay) {
        System.out.println("-----");
        System.out.println("Mounth \t | \t " + residents.getMounths());
        System.out.println("Payment \t | \t " + pay);
        System.out.println("-----");
    }

    public static void printBill(Resident residents, float pay) {

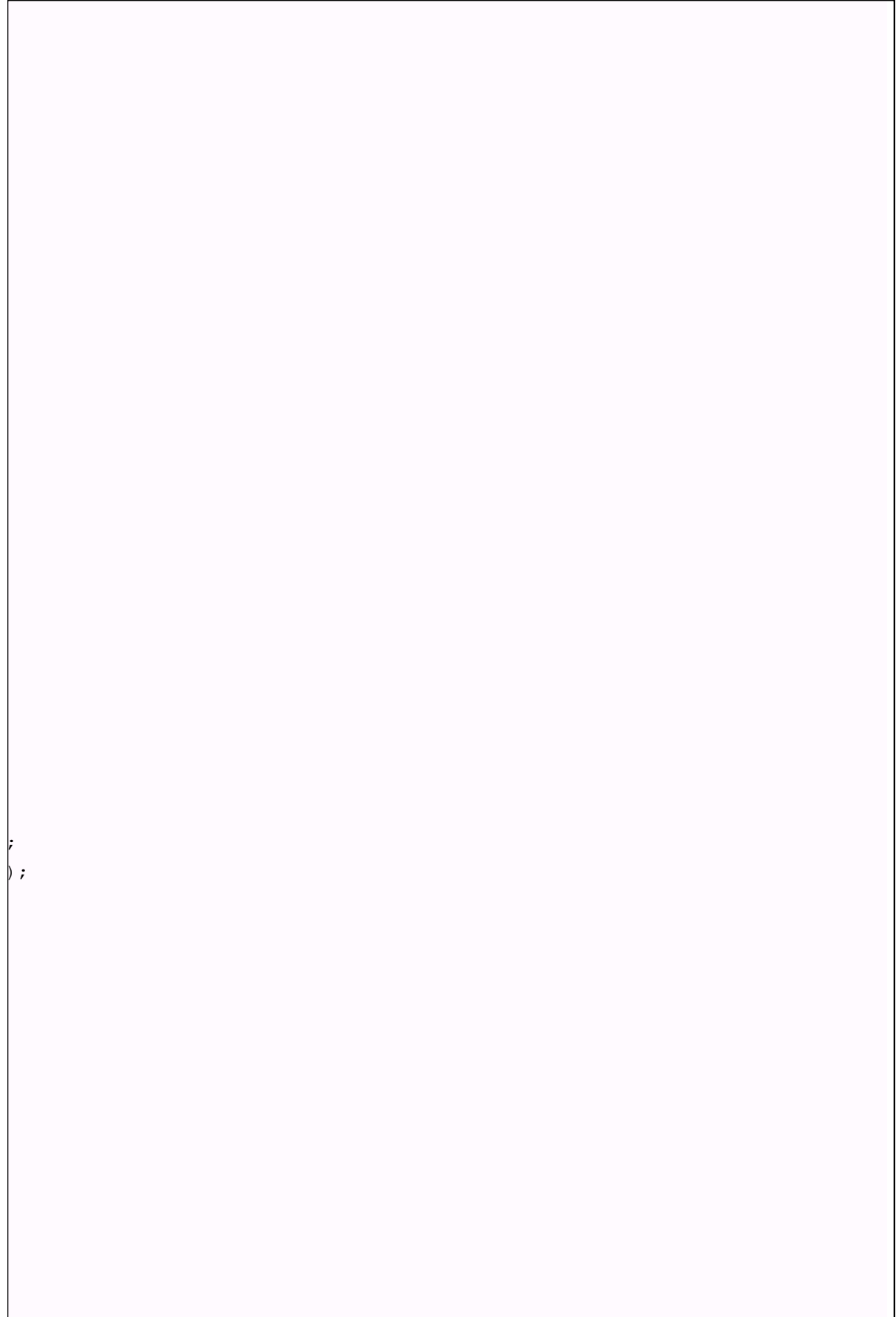
        System.out.println("-----");
        System.out.println("| National Polytechnic School Urbanization |");
        System.out.println("| Resident DNI: " + residents.getDni());
        System.out.println("| Name: " + residents.getName());
        System.out.println("| Batch Number: " + residents.getBatchNumber());
        System.out.println("| Mounth of payment: " + residents.getMounths());
        System.out.println("| Payment: " + pay);
        System.out.println("-----");

    }

    private static void ArchiveListResidents(Resident residents) {

        File residentFile = new File("ResidentsList.csv");
        long residentDni = residents.getDni();
        String residentName = residents.getName();
        float payAliquot = residents.getAliquot();
        float payAccessCard = residents.getAccessCard();
        float payAccessControl = residents.getAccessControl();
        float payExtraordinary = residents.getAccessControl();
    }

```



;
) ;

```
try {  
  
    PrintWriter writer = new PrintWriter(new FileWriter(residentFil  
writer.print(residentDni + "," + residentName + "," + payAliquo  
writer.close());  
  
    } catch (FileNotFoundException e) {  
        e.printStackTrace(System.out);  
    } catch (IOException e) {  
        e.printStackTrace(System.out);  
    }  
}
```

```
}
```

```
e, true));  
t + "," + payAccessCard + "," + payAccessControl + "," + payExtraordinary);
```