```java
// Af Mikael Steenberg Pasovski

public class Main {

    public static void main(String[] args) {
        Printer print = new Printer();
        print.printMembers();
        System.out.println();
        print.printEmployees();
        System.out.println();
        FileHandling fileH = new FileHandling();
        fileH.getPersonList();
    }
}

```

```java
public class Member extends Person {

    private Boolean isBasic;

    public Member(Boolean isBasic, String name, String cpr) {
        super(name, cpr);
        this.isBasic = isBasic;
    }

    public String getMemberType() {
        if (isBasic) {
            return "Basic";
        } else {
            return "Full";
        }
    }

    public int monthlyFee() {
        int monthlyFee;
        getMemberType();
        if (isBasic) {
            monthlyFee = 199;
        } else {
            monthlyFee = 299;
        }
        return monthlyFee;
    }

}

```

```java
public class Person {

    protected String name;
    protected String cpr;

    public Person (String name, String cpr) {
        this.name = name;
        this.cpr = cpr;
    }

    public String getName() {
        return name;
    }

    public String getCpr() {
        return cpr;
    }
}

```

```
1 Ole, 1203401211
2 Patrick, 1712951825
3 Pernille, 0303981644
4 Kamilla, 1411780926
5 Mads, 0905921337
6
7
```

```java
import java.util.ArrayList;

public class Printer {

    //printer members fra arraylist
    public void printMembers() {
        ArrayList<Member> memberList = new ArrayList<>();
        memberList.add(new Member(true, "Kalle", "0505661921"));
        memberList.add(new Member(true, "Jens", "1204980909"));
        memberList.add(new Member(false, "Katrine", "2301451920"));
        memberList.add(new Member(false, "Bob", "0312787623"));
        memberList.add(new Member(true, "Hansine", "1110951022"));

        System.out.println("FITNESS MEMBERS");
        //formaterer output vha printf, left-aligner al tekst
        System.out.printf("%-10s%-15s%-15s%-15s\n", "Name", "Cpr", "Member type", "Fee");
        for (int i = 0; i < 65; i++) {
            System.out.print('*');
        }
        //går igennem listen
        System.out.println();
        for (Member memb : memberList) {
            System.out.printf("%-10s%-15s%-15s%-15d\n", memb.getName(), memb.getCpr(),
                    memb.getMemberType(), memb.monthlyFee());
        }

        System.out.println();
        for (int i = 0; i < 50; i++) {
            System.out.print('=');
        }

    }

```

```
34      public void printEmployees() {
35          //arrayList af super class, tilføjer subclasses
36          ArrayList<Employee> empList = new ArrayList<>();
37          empList.add(new Instructor("Hansi","1204952311", 24));
38          empList.add(new Administration("Bob", "0707672119", 37, 23000, 5));
39          empList.add(new Instructor("Kaj", "1307856737", 31));
40          empList.add(new Administration("Oline","2412051328",37,23000, 5));
41          empList.add(new Instructor("Ida", "1607882356", 37));
42
43          System.out.println("FITNESS EMPLOYEES");
44          System.out.printf("%-10s%-15s%-15s%-15s%-15s\n", "Name", "Cpr", "Hours", "Salary", "
    Vacation");
45
46          for (int i = 0; i < 65; i++) {
47              System.out.print('*');
48          }
49          System.out.println();
50
51              //tjekker efter subclass, og caster Employee hvis den er af Administration-typen,
    for at
52              // få adgang til getVacation-metode
53              for (Employee emp : empList) {
54                  if (emp instanceof Administration) {
55                      System.out.printf("%-10s%-15s%-15d%-15d%-15d\n", emp.getName(),
56                              emp.getCpr(),
57                              emp.getHours(), emp.getSalary(), ((Administration) emp).getVacation
    ());
58                  } else {
59                      System.out.printf("%-10s%-15s%-15d%-15d\n", emp.getName(),
60                              emp.getCpr(),
61                              emp.getHours(), emp.getSalary());
62
63                  }
```

```
64              }
65
66          System.out.println();
67          for (int i = 0; i < 50; i++) {
68              System.out.print('=');
69          }
70      }
71
72      public void printPersons() {
73
74      }
75  }
76
```

```java
public class Employee extends Person {

    private int hours;
    private int salary;

    public Employee (String name, String cpr, int hours, int salary) {
        super(name, cpr);
        this.hours = hours;
        this.salary = salary;
    }

    public Employee (String name, String cpr, int hours) {
        super(name, cpr);
        this.hours = hours;
    }

    public int getHours() {
        return hours;
    }

    public int getSalary() {
        return salary;
    }
}

```

```java
public class Instructor extends Employee {


    public Instructor(String name, String cpr, int hours) {
        super(name, cpr, hours);


    }

    public int getSalary() {
        return getHours() * 456;
    }

}

```

```java
import java.io.File;
import java.io.FileNotFoundException;
import java.util.ArrayList;
import java.util.Scanner;

public class FileHandling {

    private ArrayList<Person> personList;

    private void createList() {
        File file = new File("src/Persons.txt");
        personList = new ArrayList<>();
        // try-with-resources så scanner automatisk lukker efter brug
        try (Scanner reader = new Scanner(file)) {
            //laver et string-array fra hver linje i tekstfilen, som deles op ved komma
            while(reader.hasNext()) {
                String[] arr = reader.nextLine().split(",");
                String name = arr[0];
                String cpr = arr[1];
                personList.add(new Person(name, cpr));
            }
        } catch (FileNotFoundException e) {
            System.out.println("Error: " + e);
        }
    }

    public void getPersonList() {
        System.out.println("EMPLOYEES AND MEMBERS Name and cpr");
        System.out.printf("%-12s%4s\n","Name", "cpr");
        for (int i = 0; i < 40; i++) {
            System.out.print('*');
        }
        System.out.println();
```

```
34
35          //kalder createList-metoden og går igennem listen
36          createList();
37          for (Person p : personList) {
38              System.out.printf("%-12s%10s\n", p.getName(), p.getCpr());
39          }
40
41          for (int i = 0; i < 30; i++) {
42              System.out.print('=');
43          }
44      }
45
46  }
47
```

```java
public class Administration extends Employee {

    private int vacation;

    public Administration(String name, String cpr, int hours, int salary, int vacation) {
        super(name, cpr, hours, salary);
        this.vacation = vacation;

    }

    public int getVacation() {
        return vacation;
    }
}

```