Problem 1

Create a class **Person**, objects of which describe persons, and which contains only two fields: name (**String**) and year of birth (**int**). In this class, define

- constructor taking name and year of birth;
- constructor taking only name and setting the year of birth to default value 1990;
- method **getName** returning the name of a given person;
- method **getBirthYear** returning the year of birth of a given person;
- method **isFemale** returning **true** if *this* person is a woman (we assume, not very sensibly, that only women and all women have names ending with the letter 'a'); otherwise the method returns **false**;
- static function **getOlder** taking two references to objects of class **Person** and returning the reference to the older of these two persons;
- static function **getOldest** taking the reference to an *array* of references to objects of class **Person** and returning the reference to the oldest person represented in the array;
- static function **getYoungestFemale** taking the reference to an *array* of references to objects of class **Person** and returning the reference to the youngest *woman* represented in the array, or **null** if there is no woman in the array;
- redefinition of method **toString** from class **Object**.

In a separate class, write a **main** function in which the whole functionality of the class **Person** is tested.

Problem 2

Write a program simulating operations on bank accounts. Accounts are objects of class **Account**. Each account has a field balance and allows to:

- deposit money (method **deposit(...)**);
- withdraw money (method **withdraw(...)**);
- transfer money to another account (method **transfer(...)**);
- add interest given as annual rate (method **addInterest(...)**).

The rate is common for all accounts and can be set by a static function **setInterestRate(...)**.

Accounts are owned by customers (objects of class **BankCustomer**). Customers refer to persons (objects of class **Person**).

Define all the required classes, so the following program:

```
public class BankingTest {
    public static void main(String[] args) {
        Person johnP = new Person("John"),
```

```
            maryP = new Person("Mary");
        BankCustomer john = new BankCustomer(johnP);
        BankCustomer mary = new BankCustomer(maryP);

        john.getAccount().deposit(1000);
        mary.getAccount().deposit(2000);
        john.getAccount().transfer(
                            mary.getAccount(),500);
        mary.getAccount().withdraw(1000);

        System.out.println(john);
        System.out.println(mary);

        Account.setInterestRate(4.5);
        john.getAccount().addInterest();
        mary.getAccount().addInterest();

        System.out.println(john);
        System.out.println(mary);
    }
}
```

prints on standard output (out):

```
Customer: John, balance: 500.0
Customer: Mary, balance: 1500.0
Customer: John, balance: 522.5
Customer: Mary, balance: 1567.5
```

The form of the output is obligatory.

Program should work for any deposits/withdrawals, that is you should take into account a possibility of illegal operations (negative values, withrawals exceeding current balance etc.) The simplest way to accomplish this will be just to ignore those operations. All possible exceptions should be handled (for printing error messages use standard output out).

Do not modify the main class **BankingTest**.
