Having a Bank class (see the UML class diagram), we would like to manage the customer accounts.

```java
Bank

- name : String
- accNum [ ] : int
- balance [ ] : double
- type [ ] : String
- current : int

+ Bank (bankName : String, size: int)
+ searchIndexOfAccNumber(accNb : int) : int
+ deposit(accNb : int, amount : double) : void
+ withdraw(accNb : int, amount : double) : void
+ addNewAccount(accNb: int, bal : double) : boolean
+ deleteAccount(accNb : int) : boolean
+ display(typ: String) : void
+ accNumThatHasMaxBalance( void) : int
+ averageInBalanceForAccount(typ: String) : double
+ whichTypeHasMaxAccNumber( void) : String
+ minNumberOfAccount(Bank :b) : Bank
+ totalBalanceInTheBank() : double
+ nameOfBankThathasGreatestTotalBalance(Bank b) : String
+ deleteAllAccountOfType(typ: String) : void
+constructBranchOfBankforAllAccountOfType(typ: String) : Bank
+constructTwoBranches(Bank b1, Bank b2) : void
```

The attributes:

- **name**: name of the Bank.
- **accNum**: array containing the account number of accounts.
- **balance**: array containing the balance of accounts.
- **type**: array containing the type of accounts. A type of account is “golden” if the balance is greater than or equal 100,000 SR, otherwise it is a “silver” type. When we perform a deposit or withdraw operation the type of the account may change according to the value of balance.
Methods:

- **Bank(bankName : String, size : int)**: each new account starts with zero balance.
- **searchIndexOfAccNumber(accNb : int )**: returns the index in array of the account number accNb, otherwise returns -1.
- **deposit(accNb : int, amount : double)**: allows for depositing an amount amount for the account number accNb.
- **withdraw(accNb : int, amount:  double)**: allows for debiting an amount amount from the account number accNb.
- **addNewAccount(accNb: int, bal : double )**: allows for adding a new account to the Bank with account number accNb and balance bal. It returns true if the operation is done, otherwise it returns false.
- **deleteAccount(accNb: int )**: allows for deleting an account from the Bank with account number accNb. It returns true if the operation is done, otherwise it returns false.
- **display(typ:  String)**: allows for displaying the information of all accounts that have type typ.
- **accNumThatHasMaxBalance( void)**: It returns the account number which has max balance.
- **averageInBalanceForAccount(typ: String)**: average in balances for all accounts in type "typ".
- **whichTypeHasMaxAccNumbers(void)**: String
- **minNumberOfAccount(Bank :b )**: compare the current bank and b and return the bank that has minimum number of account.
- **totalBalanceInTheBank( ): double**
- **nameOfBankThathasGreatestTotalBalance(Bank b) : String**
- **deleteAllAccountsOfType(typ: String)**: void
- **constructBranchOfBankforAllAccountOfType(typ: String)**: This method returns a Bank Branch from the current Bank and put on it all the accounts in type "typ".
- **constructTwoBranches(Bank b1, Bank b2)**: this method creates a branch b1 for all accounts with type "Golden" and branch b2 for all accounts with type "Silver".

**Question:**

1- Write in Java the class Bank
2- Test all the methods in the class Bank