```
import java.util.Arraylist;
class BankAccount {
    private static int accountNumber;
    private double balance;
    private String name;
    public BankAccount (String name){
        this(name, 0);
        accountNumber++;
    }
    public BankAccount (String name, double balance){
        this.name =name;
        this.balance = balance;
        accountNumber ++;
    }
    public int getAccountNumber(){ return accountNumber;}
    public void setAcccountNumber (int accountNumber){ this.accountNumber = accocentNumber;}
    public double getBalance (){return balance;}
    public void setBalance (double balance){ this.balance =balance;}
    public String getName (){return name;}
    public void setName (String name){ this.name = name;}
}


class Transaction {
    private double money;
    public string getDate(){
        SimpleDateFormat s1 = new SimpleDateFormat("dd/MM/yyyy");
        Date d1 = new Date();
        return (s1.format(d1)+"");
    }
    public String getMoney(){return money;}
    public void setMoney(String money){ this.money =money;}

}
```

```
class Checking extends BankAccount {
    private ArrayList<Transaction> transaction = new ArrayList<>();
    public Checking (String name){ super(name); }
    public Checking (String name, double balance){ super(name, balance); }
    public void deposit (double amount) {
        setBalance (getBalance() + amount);
        if (transactions.size() ≤ 100 ){
            Transaction t1 = new Transaction();
            t1.setMoney("+ "+amount);
            transactions.add(t1);
        } else {
            System.out.println("Your transaction is full. Please change");
        }
    }
    public void withdraw(double amount) {
        setBalance(getBalance() - amount);
        if ( transations.size() ≤ 100 ){
            Transaction t1 = new Transaction();
            t1.setMoney ("- "+amount);
            transactions.add(t1);
        }else {
            System.out.println ( "Your transaction is full. Please change");
        }
    }
    public void getAllTransaction() {
        for (int i=0; i< transactions.size(); i++){
            System.out.println ((i+1) + "    "+ transactions.get(i).getDate() + "    "+ transactions.get(i).getMoney());
        }
    }
}
```

```
class Saving extends BankAccount {
    private double futureValue;
    private double interestRate;
    private double depositMoney;
    private double numOfReceiveYr;
    private double TimePeriodYr;
    private ArrayList<Transaction> transaction = new ArrayList<>();
    public Saving (String name){ super(name); }
    public Saving (String name, double balance){ super(name, balance); }
    public double getInterestRate(){ return interestRate; }
    public void setInterestRater(double interestRate){ this.InterestRate = interestRate; }
    public double getDepositMoney (){ returr depositMoney; }
    public void setDepositMoney(double depositMoney){ th depositMoney = depositMoney; }
    public double getNumberOfReceivePerYr (){ retur numberOfReceivePerYr; }
    public void setNumberOfReceivePerYr(double numberOfReceivePerYr){
        this.numberOfReceiveYr = numberOfReceiveYr;
    }
    public double getTimePeriodYr(){ return timePeriodYr; }
    public void setTimePeriodYr (double timePeriodYr){ this timePeriodYr = timePeriodYr; }
    public void deposit (double amount){
        setBalance (getBalance() + amount);
        if (transactions.size() <= 100){
            Transaction t1 = new Transaction();
            t1.setMoney("+" + amount);
            transactions.add(t1);
        } else {
            System.out.println("Your transaction is full. Please change");
        }
    }
```

```java
public void withdraw(double amount){
    if((getBalance()-amount)>0){
        setBalance(getBalance()-amount);
        if(transations.size() <= 100){
            Transaction t1 = new Transaction();
            t1.setMoney("-"+amount);
            transactions.add(t1);
        }else{
            System.out.println("Your transaction is full. Please change");
        }
    }else{
        System.out.println("Your money not enough to withdraw");
    }
}
public void getAllTransaction(){
    for(int i=0;i<transactions.size();i++){
        System.out.println((i+1)+"    "+transactions.get(i).getDate()+"    "+transactions.get(i).getMoney());
    }
}
public void calculateFutureValue(double deposit,double interestPercent, double numOfReceivePerYr,
double timePeriodYr){
    this.depositMoney = depositMoney;
    this.interestRate = interestPercent;
    this.numberOfReceivePerYr = numberOfReceivePerYr;
    this.timePeriodYr = timePeriodYr;
    if(numOfReceivePerYr == 1){
        this.futureValue=depositMoney*Math.pow((1+(this.interestRate/100)), this.timePeriodYr);
        System.out.printf("After calculate the interest rate in %.1f : %.2f ", this.timePeriod, this.futureValue);
        System.out.println();

    }else if(numOfReceivePerYr == 2){
        this.futureValue=depositMoney*Math.pow((1+(this.interestRate/100)), this.timePeriodYr*2);
        System.out.printf("After calculate the interest rate in %.1f : %.2f ", this.timePeriod, this.futureValue);
        System.out.println();
```

```
} else if (numofReceivePerYr == 12) {
    this.futureValue = depositMoney * Math.pow((1 + (this.interestRate/100)), this.timePeriodYr * 12);
    System.out.printf("After calculate the interest rate in %.1f : %.2f ", this.timePeriod, this.futureValue);
    System.out.println();
  }
}
```