```java
import java.util.*;
public class Account2{
    private int id;
    String name="unknown";
    private double balance;
    private double annualInterestRate=0;
    private Date dateCreated;
    ArrayList<transactions> transactions = new ArrayList<transactions>();
    public Account2(){
        id = 0;
        balance = 0;
        annualInterestRate = 0;
        dateCreated = new Date();
    }
    public Account2(int id,double balance){
        this.id = id;
        if(id<0){
            System.out.println("Account ID must be positive");
            System.exit(0);
        }
        this.balance = balance;
        if(balance <0){
            System.out.println("Invalid balance");
            System.exit(0);
        }
        dateCreated = new Date();
    }
    public Account2(String name,int id,double balance){
        this.name = name;
        this.id = id;
        if(id<0){
            System.out.println("Account ID must be positive");
            System.exit(0);
        }
        this.balance = balance;
        if(balance <0){
            System.out.println("Invalid balance");
            System.exit(0);
        }
        dateCreated = new Date();
    }
    //accessor
    public int getId()              {return id;}
    public double getBalance()          {return balance;}
    public double getAnnualInterestRate()   {return annualInterestRate;}
    public Date getdateCreated()        {return dateCreated;}
    public double getMonthlyInterestRate()  {return annualInterestRate/12;}
    public double getMonthlyInterest()      {return balance*getMonthlyInterestRate();}
    //mutator
    public void setID(int id)           {this.id = id;}
    //public void setBalance(double balance)  {this.balance = balance;}
    public void setAnnualInterestRate(double annualInterestRate){this.annualInterestRate=annualInterestRate;}
    //function
    public void printlog(){
        if(transactions.size()==0){
            System.out.println("No logs");
        }
        else{
            System.out.println("Date                        Type       Amount      Balance");
            for(int i=0;i<transactions.size();i++){
                System.out.print(transactions.get(i).getDate());
                System.out.print("        ");
                System.out.print(transactions.get(i).getType());
                System.out.print("           ");
```

```java
                System.out.print(transactions.get(i).getAmount());
                for(int j=0;j<13-String.valueOf(transactions.get(i).getAmount()).length();j++)
                    System.out.print(" ");
                System.out.println(transactions.get(i).getBalance());
            }
        }
    }
    public void withdraw(double amount){
        if(amount<0){
            System.out.println("Invalid amount");
        }
        else if(balance<amount){
            System.out.println("Insufficient funds");
        }
        else{
            transactions.add(new transactions('W',amount,balance,"Withdraw"));
            balance -= amount;
        }
    }
    public void deposit(double amount){
        if(amount<0){
            System.out.println("Invalid amount");
        }
        else{
            transactions.add(new transactions('D',amount,balance,"Deposit"));
            balance += amount;
        }
    }
}
```