```
/* Project 1 - Harley Phung
   Task 2 project 1: Create Cash class that contained current balance, saving
rates, loan rate, loan's limit, overdraft Penalty.
   That can calculate the balance when deposit, withdraw, trasnfer money. Check the
account by the end of the day and month if overdraft penalty is needed to be
applied.
*/

public class Cash {
  //A field that stored account's current balance
  private double balance = 0.0;
  //A field that stored account's current Saving rates
  private double savingsRate = 0.0;
  //A field that stored account's Loan Rates
  private double loanRate = 0.0;
  //A field that stored account's Loan Limit
  private double limit = 0.0;
  //A field that stored account's overdraft Penalty that applied to the account
  private double overdraftPenalty = 0.0;
  //A field that stored deposit amount
  private double amount = 0.0;
  //A field that stored monthly interest
  private double monthlyInterest = 0.0;
  //A method that check if the balance plus current monthly interest smaller than 0
and larger than loan limit
  private boolean checkOverdraft = true;

  //Constructor that stored savings rate, loan rate, loan limit, overdraft Penalty
  public Cash (double savingsRate, double loanRate, double loanLimit, double
overdraftPenalty) {
    this.savingsRate = savingsRate;
    this.loanRate = loanRate;
    this.limit = loanLimit;
    this.overdraftPenalty = overdraftPenalty;
  }

  //A method that return the current balance in the account
  public double getBalance() {
    return this.balance;
  }

  //A method that returns the interest rate when the current balance is positive
  public double getSavingsRate() {
    return this.savingsRate;
  }
  //A method that changes the interest rate when the current balance is positive
  public void setSavingsRate(double savingsRate) {
    this.savingsRate = savingsRate;
  }

  //A method that returns the interest rate when the current balance is negative
  public double getLoanRate() {
    return this.loanRate;
  }
  //A method that changes the loan rates when the current balance is negative
  public void setLoanRate(double loanRate) {
    this.loanRate = loanRate;
  }
```

```
  //A method that returns the loan limit for the account
  public double getLoanLimit() {
    return this.limit;
  }

  //A method that changes the loan limit for the account
  public void setLoanLimit(double limit) {
    this.limit = limit;
  }

  //A method that return the overdrat penalty for the account
  public double getOverdraftPenalty() {
    return this.overdraftPenalty;
  }

  //A method that changes the overdraft penalty for the account
  public void setOverdraftPenalty(double penalty) {
    this.overdraftPenalty = penalty;
  }

  //A method that increase the balance amount by adding money to the account
  public void deposit(double amount) {
    this.balance = this.balance + amount;
  }

  //A method that check whether withdrawn amount is larger or less than current
balance
  public boolean withdraw(double amount) {
    if (amount <= this.getBalance()) {
      this.balance = this.getBalance() - amount;
      return true;
    }
    else {
      return false;
    }
  }

  //A method that reduces the current balance by an amount
  public void transfer(double amount) {
    this.balance = this.getBalance() - amount;
  }

  //A method that calculate how much interest is added or removed each day.
  public void processDay() {
    if(this.getBalance() >= 0) {
      this.monthlyInterest = this.monthlyInterest + this.getBalance() *
this.getSavingsRate() / 365;
    }
    else {
      this.monthlyInterest = this.monthlyInterest - this.getBalance() *
this.getLoanRate() / 365;
    }
    if(this.getBalance() + this.monthlyInterest < 0 && this.getBalance() +
this.monthlyInterest < -this.getLoanLimit() && this.checkOverdraft == true) {
      this.checkOverdraft = false; //false is when the overdraft penalty is apply
    }
  }

  //A method that calculate the balance amount is added or removed each month.
```

```
  public void processMonth() {
    this.balance = this.balance + this.monthlyInterest;
    this.monthlyInterest = 0;
    if(this.checkOverdraft == false) {
      this.balance = this.balance - this.overdraftPenalty;
      this.checkOverdraft = true;
    }
  }
}
```