**1 class DataAccess** có các function tạo, cập nhật và xóa dữ liệu database như sau:

public class DataAccess {

    public void createData(){

        System.out.println("Created data record.");

    }

    public void updateData(){

        System.out.println("Updated data record");

    }

    public void deleteData(){

        System.out.println("Deleted data record");

    }

}

Trong chương trình chính ta thực hiện gọi các function của DataAccess:

public class Main {

    public static void main(String[] args) {

        DataAccess dao=new DataAccess();

        dao.createData();

        dao.updateData();

        dao.deleteData();

    }

}

Cách chúng ta thường làm là định nghĩa 1 class Log và trong DataAccess gọi phương thức lả Log để ghi ra log

public class Log {

    public void logMessage(String message){

        System.out.println("["+ new SimpleDateFormat("yyyy-MM-dd kk:mm:ss").format(new Date()) +"] Meesage from logger: "+message);

    }

}

DataAccess phải sửa lại như sau:

public class DataAccess {

    private Log log=new Log();

    public void createData(){

        log.logMessage("call create data");

        System.out.println("Created data record.");

    }

    public void updateData(){

        log.logMessage("call update data");

        System.out.println("Updated data record");

    }

    public void deleteData(){

        log.logMessage("call delete data");

        System.out.println("Deleted data record");

    }

}

**Định nghĩa 1 aspect như sau:**

public aspect LogDataAccessAspect {

    private Log log=new Log();

    pointcut dataAccessExecution() : (execution(public \* DataAccess.\*(..)));

    before() : dataAccessExecution() {

        log.logMessage("call "+thisJoinPointStaticPart.getSignature());

    }

}

Ý nghĩa của LogDataAccessAspect như sau:  
  
+ Chọn các phương thức public của class DataAccess, đưa vào 1 tập hợp tên là dataAccessExecution() 

|  |  |
| --- | --- |
|  | pointcut dataAccessExecution() : (execution(public \* DataAccess.\*(..))); |

+ Với mỗi phương thức trong tập hợp trên, trước khi thực thi phương thức đó thì thực hiện câu lệnh log.logMessage

|  |  |
| --- | --- |
|  | before() : dataAccessExecution() {          log.logMessage("call "+thisJoinPointStaticPart.getSignature());  } |

Sau khi chỉnh sửa thì ta có lớp DataAccess như sau:

public class DataAccess {

    public void createData(){

        System.out.println("Created data record.");

    }

    public void updateData(){

        System.out.println("Updated data record");

    }

    public void deleteData(){

        System.out.println("Deleted data record");

    }

}

Output project

