

國立雲林科技大學考試答案卷

No 0447715

第 1 頁

學 年	學 期	日 期	考 別	<input type="checkbox"/> 平時考 <input type="checkbox"/> 期中考 <input type="checkbox"/> 學期考
科 目	S.A.D		評 分	39
系 所	四學管	年 級	= 學 號	B10523005 姓 名 李蒙孝 Aliss

1. ① 4 ② 1 ③ 3 ④ 2

2. do

check employee status

if status = true

check hours worked

calculate gross wages

calculate tax

while any more employees

3. I think this is ^{checkout} trivial. Nomatter what the message is, it just need to display. It don't need to do different act when it meet different message.

5. public class Employee {

private Account account;

private String empId;

public Employee () {

account = new Account ();

}

public Account getAccount () {

return account; +2

}

public String getEmpId () {

return empId;

}

public void setEmpId (String empId) {

this.empId = empId;

}

public class Account {

private Employee owner;

public Account (Employee owner) {

owner = new Employee (this);

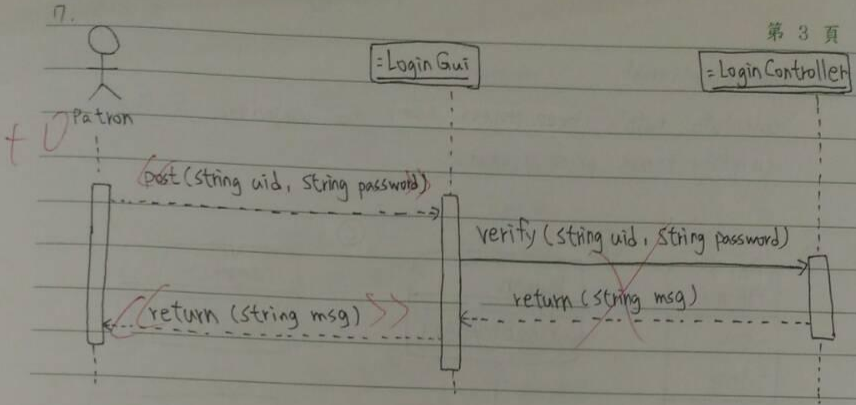
}

public Employee getOwner () {

return owner; +2

}

}



+5 9. "cnList" is an object that others sent when they call the process function. Because we use "for-loop" to get every item in "cnList", we didn't miss any item. so this is data coupling.

In process function, there has a flag about Available.

We check every single item in cnList is available or not.

If it is available then we save it, and then we set this data become not available.

So this is control coupling.

other?

+4 10. ① data ✓

"calculate tax" need to use whole taxable amount to calculate.

② stamp ✓

print routine of the customer billing accept many data,

but it doesn't use everything it get just pick what it need.

+2 11. ① temporal ✓

it is about time, about initializing.

② coincidental ~~Function~~ ②? Coincidental

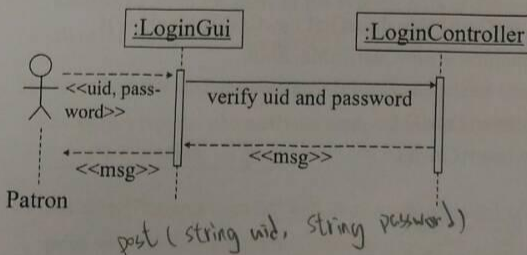
it is mix record, payment, report, competitor.

they don't have any in common things.

is updated from the name of the Employee. In this situation, the Account needs to access its corresponding Employee object. Therefore, we plan to add an owner attribute to Account. We need to ensure that if a given Account has a reference to a specific Employee, and the Employee has a reference to that same Account. Since neither the Employee class nor the Account class can modify the field anywhere else, this ensures that both reference attributes remain consistent. Please write the Java code. **8%** (Hint: As the Account object is created by the Employee constructor, add a parameter to the Account constructor to initialize the owner field to the correct value.)

6. Time passes, and the direction of an association changes again during the development of the system. Assume that an Employee can have several Accounts to track the expenses accrued for different projects. In this situation, the Employee object has a one-to-many association with the Account class. That means an Employee can have a varying number of Accounts. Since the Account object does not invoke the Account constructor, we design a control object for creating and archiving Accounts for invoking the constructor. Remember that Accounts have no specific order and an Account can be part of an Employee at most once, we use references, called accounts to model the "many" part of the association. Moreover, we decide to realize this association as a bidirectional association, and so add the addAccount(), removeAccount(), and setOwner() methods to the Employee and Account classes to update the accounts and owner fields. Please write the Java code. **10%**

7. Given a piece of sequence diagram as below and its statement as: "LoginGui calls LoginController to verify the login. The controller returns a msg indicating result of verification. LoginGui shows msg to Patron." We are now going to move from analysis to design phase. Please draw its corresponding design diagram and annotate its signature. **6%**



unt account;
unt account 1-
unt a
arou

Problem Domain Classes

