

LAB 1

UML and Modeling Tools – ASTAH

教授：熊博安 教授

助教：楊子璿

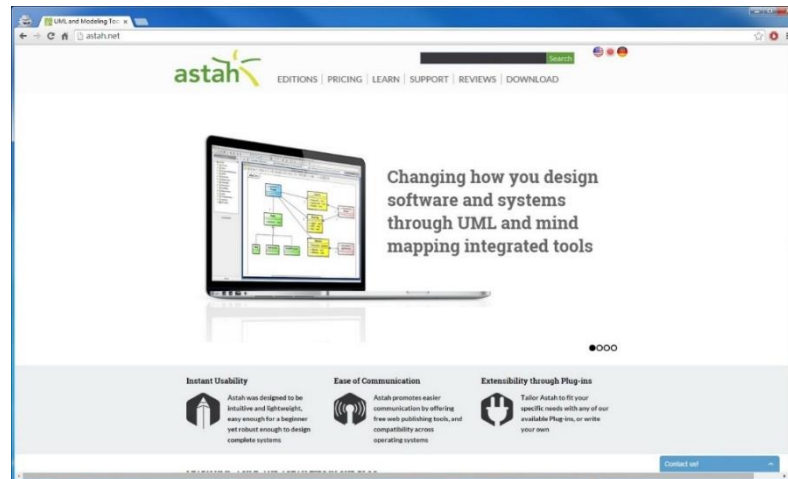
2016/10/14

Outline

- Download & Install ASTAH
- Introduce Astah
- Use Case Diagram
- Activity Diagram
- Class Diagram
- Sequence Diagram
- Practice
- UML Lab

Download & Install Astah

1. Astah官網：<http://astah.net/>
2. 下載community版



3. 安裝Astah (一直按下一步，都用預設就好)

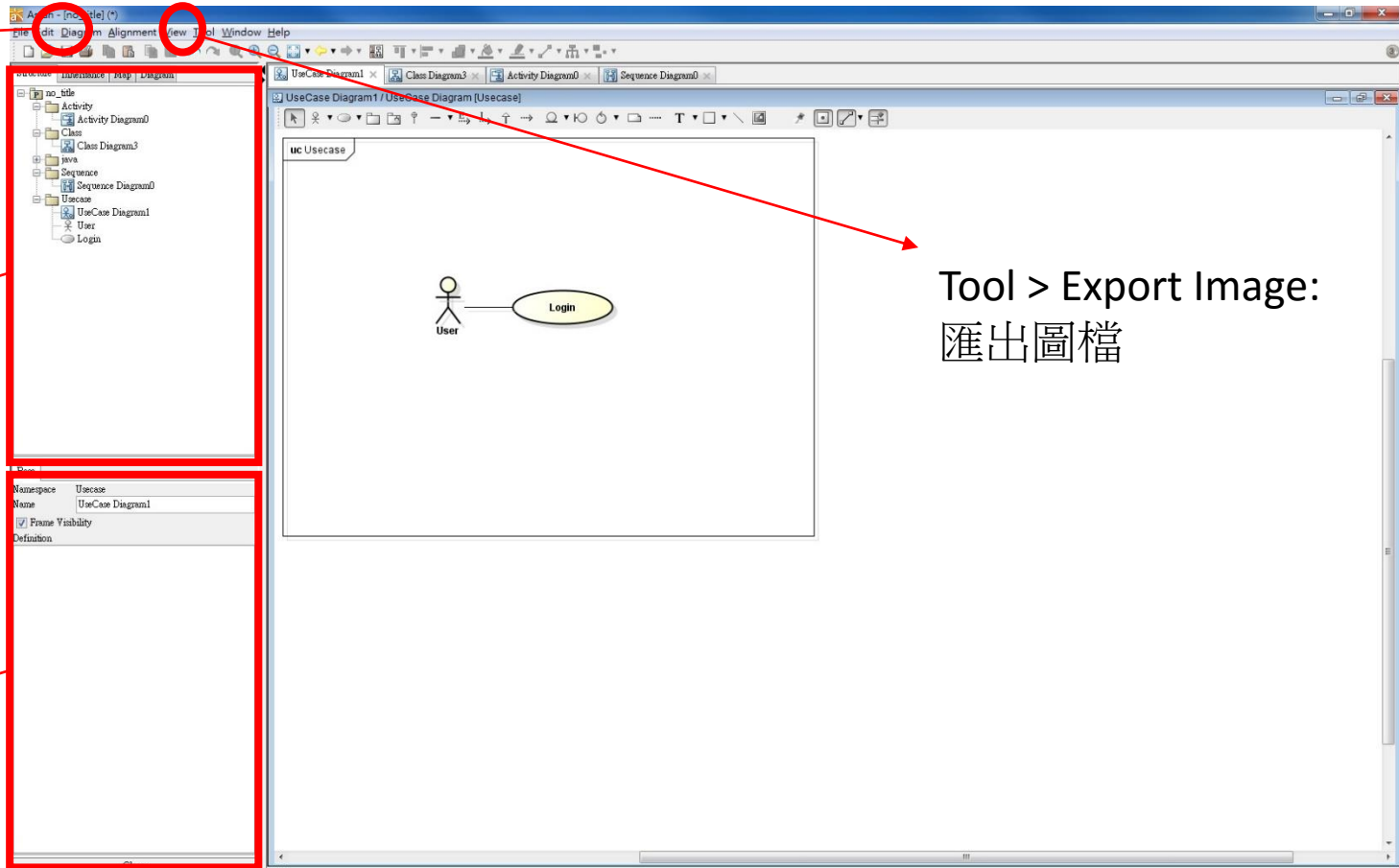
Introduce Astah

Diagram:

選擇要產生哪種圖

專案列表

屬性



Usecase Diagram

- Purpose

- 使用Usecase可幫助專案的成員了解這個系統到底是在作什麼
- 界定系統的範圍有多大
- 有多少相關外部界接系統
- 有多少角色會使用這個的系統
- 系統會提供哪些功能給哪些角色使用

Usecase Diagram (cont.)

- We Should avoid
 - Implementation details
 - GUI Information
 - Internal processing unrelated to a stakeholder request
 - Non-functional requirements

Usecase Diagram (cont.)

- Notation



- Actor不一定是human，可能為裝置、外部系統、外部裝置
- Actor不能為一個特定人物，他是一個角色(Ex:一般用戶User)

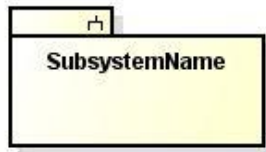


- Usecase為一個使用者的操作結果，通常命名方式是一個動作(Ex: 變更設定)
- Usecase的動作的操作是比較廣義的
- Ex:

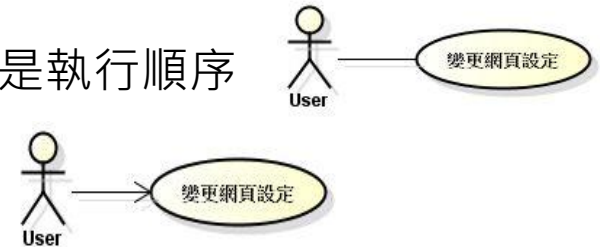


Usecase Diagram (cont.)

- Package/Subsystem界定出一個邊界，決定各系統(子系統)間有哪些Usecase。



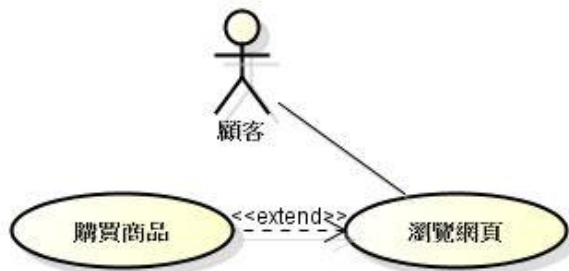
- Association
- 箭頭表示通訊的方向(資料的流向)，而不是執行順序
- 如果方向是雙向的，可加雙箭頭或不加



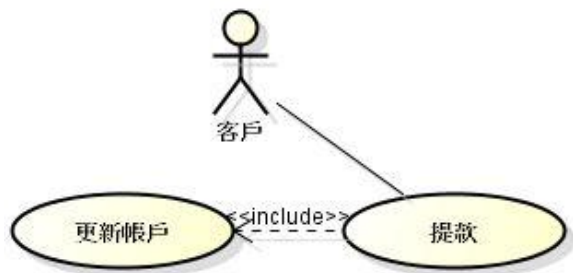
- Generalization
- 可以想成是" IS-A" 的關係
- 繼承



Usecase Diagram (cont.)



- Extends
- 選擇性的功能依附在使用者操作的功能



- Includes
- 一定會發生的功能依附在使用者操作的功能

注意兩者關係的箭頭方向





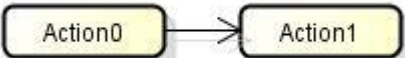
Activity Diagram

- Purpose

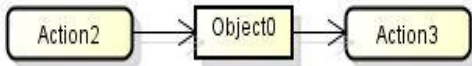
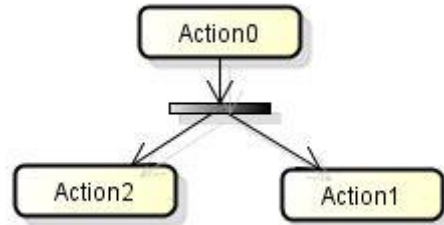
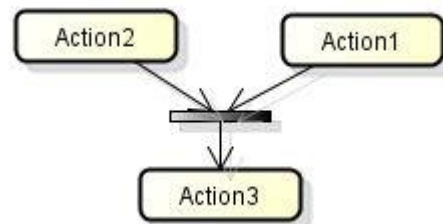
- 活動圖一般用來model：企業處理工作流程、usecase流程分析、功能處理、UI流程等
- 主要是把一系列的程序(流程)顯示出來
- 為了跟客戶溝通，盡量不要畫的太複雜
- 較少物件導向特徵
- 幫助開發人員找到未被發現的例外路徑
- 檢驗usecase描述的正确性

Activity Diagram (cont.)

- Notation

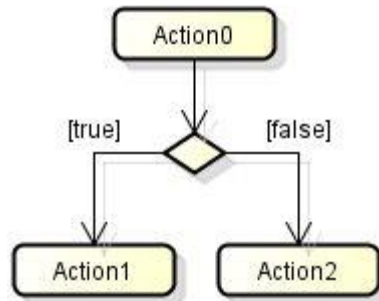
1		活動起始節點(Initial node)
2		活動終止節點(Activity final)
3		流程終止節點(Flow final node) 代表某個分支的流程終止，但還有其他流程在跑，不受此流程影響
4		活動節點(Action node) 物件節點(Object node) 前者為單純的活動，後者為一個活動的資料IO
5		控制流程(Control flow)

Activity Diagram (cont.)

6	 <pre>graph LR; Action2 --> Object0; Object0 --> Action3;</pre>	物件流程(Object flow)
7	 <pre>graph TD; Action0 --> Fork; Fork --> Action2; Fork --> Action1;</pre>	分岔節點(Fork node) 分岔後的動作是同步處理的
8	 <pre>graph TD; Action2 --> Join; Action1 --> Join; Join --> Action3;</pre>	結合節點(Join node) 必須等所有要結合的動作都完成後，才會繼續執行接下來的動作

Activity Diagram (cont.)

9



決策/合併節點(Decision/Merge node)
當一個活動進入，出去有多個活動的時候為決策，反之則為合併。決策的條件寫在guard裡面，且每條流程都要有條件

Class Diagram

- Purpose

- 表達系統內部的主要結構，此圖為UML中最重要圖
- 描述類別的型態、屬性、操作，以及類別之間的關係
- 可透過此圖套用一些Pattern，改良程式架構

Class Diagram (cont.)

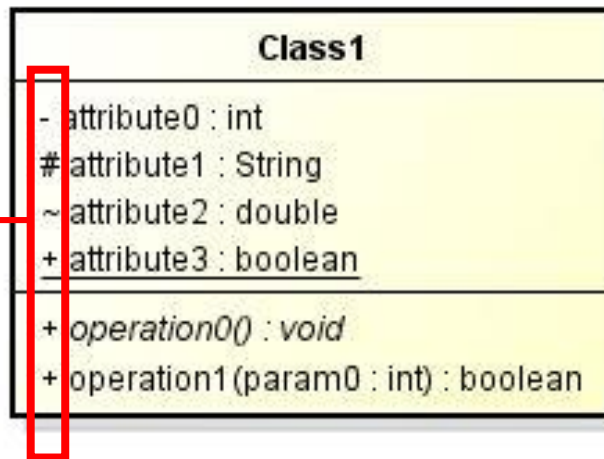
能見度

+ : public

: protected

~ : package

- : private



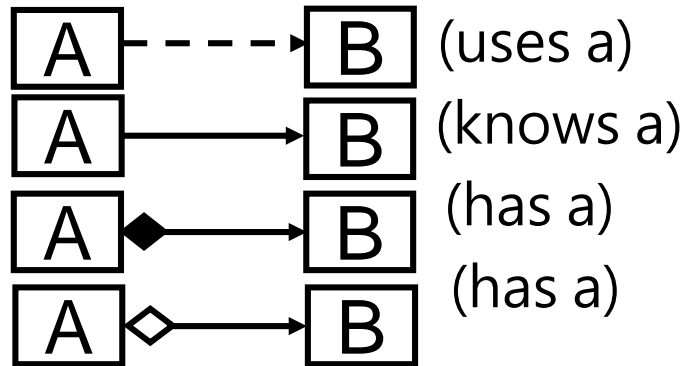
類別名稱，斜體字為抽象類別，Stereotype可詳細定義

屬性名稱:型別，有底線為static

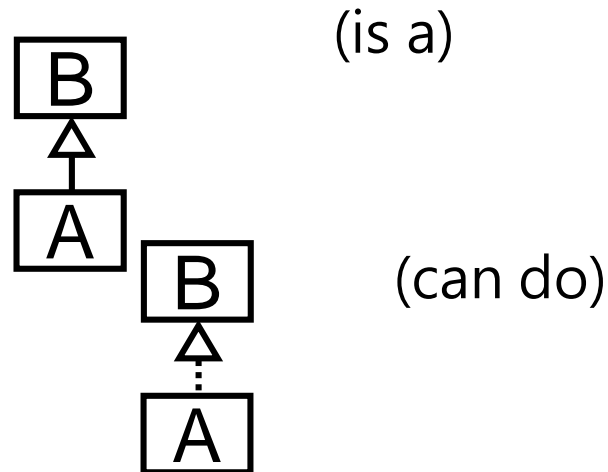
方法名稱(參數):回傳值，斜體字為抽象方法

Class Diagram (cont.)

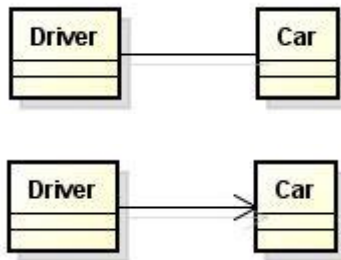
- Dependency
- Association
- Composition
- Aggregation



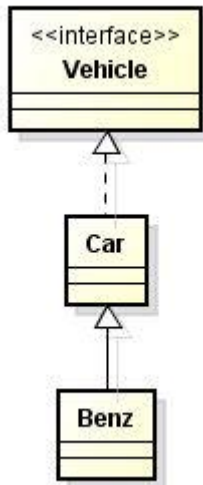
- Inheritance/
Generalization
- Realization /
Refinement



Class Diagram (cont.)

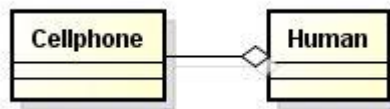


- Association
- 箭頭方向表示類別之間的互相影響
- 箭頭上如果有數字，代表數量

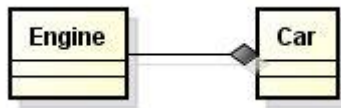


- 虛線為Realization；實線為Generalization
- Realization必須用在interface上

Class Diagram (cont.)



- Aggregation
- 兩類別的生命週期不是共存的



- Composition
- 兩類別生命週期共存，一定同時存在

把菱形畫在較大的一方

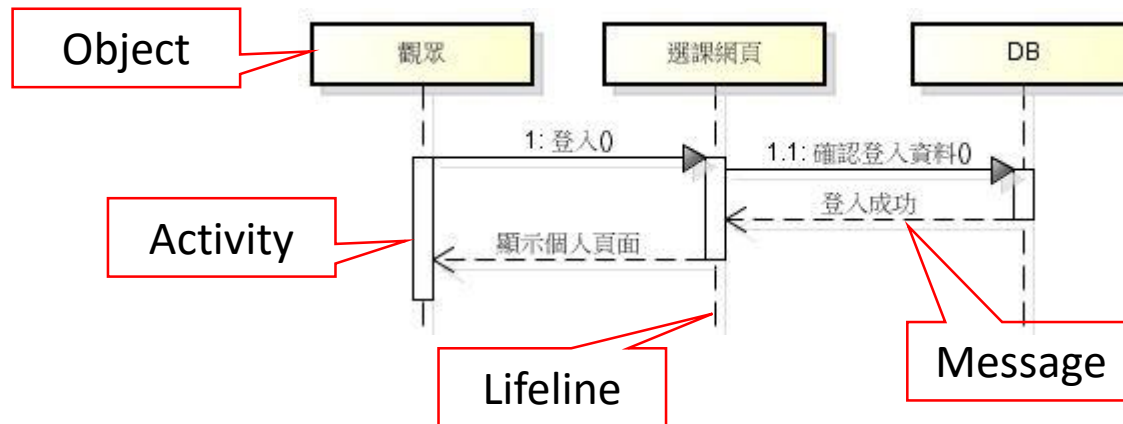
Sequence Diagram

- Purpose

- 描述動態行為 (不是確切的時間間隔)
- 基本上，一個usecase會有一到多張循序圖來描述usecase的操作，每張循序圖可視為一個scenario
- 不需要描述到太細節的部分，例如資料庫如何做連線

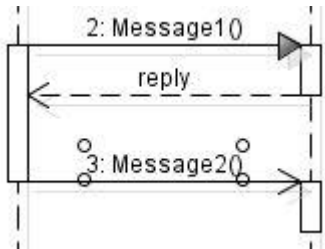
Sequence Diagram (cont.)

- 循序圖會有的元素
 - Class role : 每個物件在系統中的角色
 - Lifeline : 角色的生命週期
 - Activity : 角色處理操作花的一段時間
 - Message : 物件之間溝通的訊息

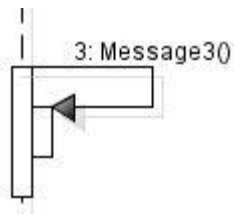


Sequence Diagram (cont.)

- Notation



- 實心箭頭為同步訊息；一般箭頭為非同步訊息
- 同步訊息有可能會要等待reply後才繼續執行
- 非同步訊息傳送過去後可直接繼續執行

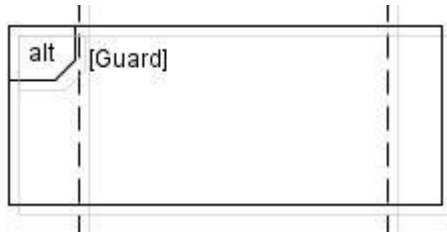


- Self message用來表達遞迴或是呼叫自己物件的方法

Sequence Diagram (cont.)



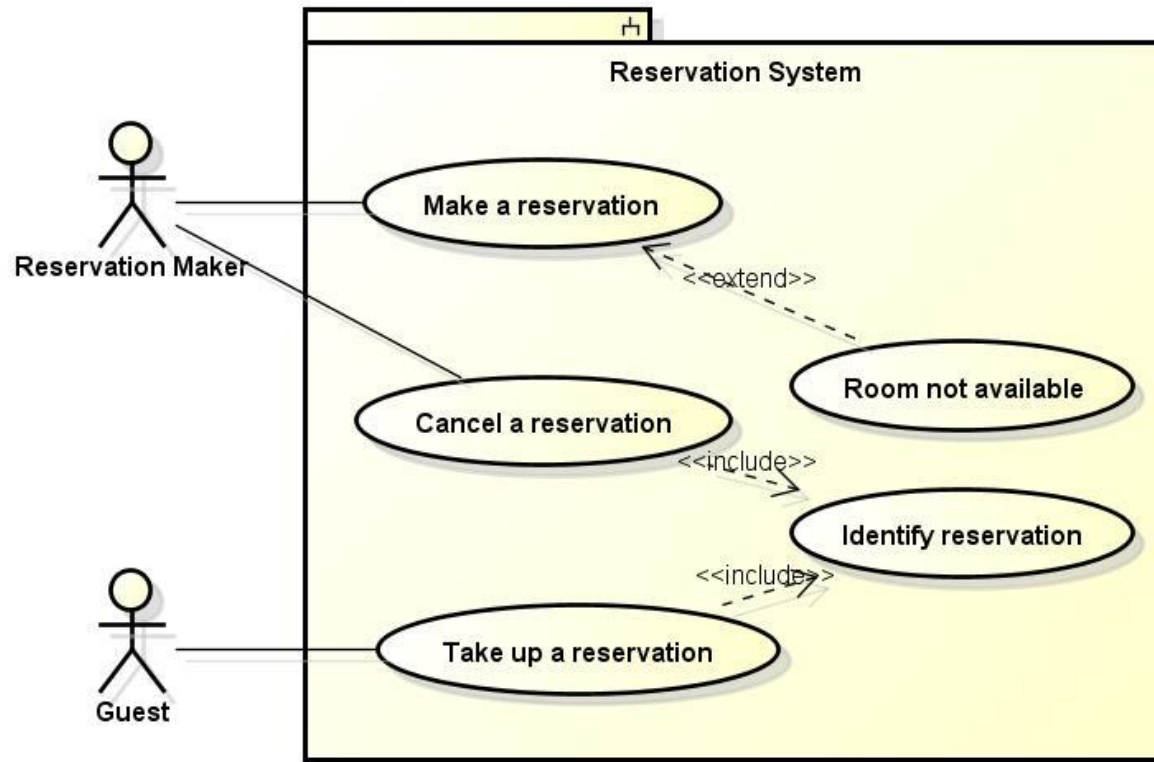
- Stop代表物件的生命週期結束



- Combined fragment
- 其中有數種Operator，常見的有Loop, Alt, Opt
- Loop只會有一個subfragment，條件成立就會執行，有時候break的combined fragment會在其中出現
- Alt會有二至多個subfragment，會執行條件成立的那一個
- Opt只會有一個subfragment，條件成立就會執行

Practice 1

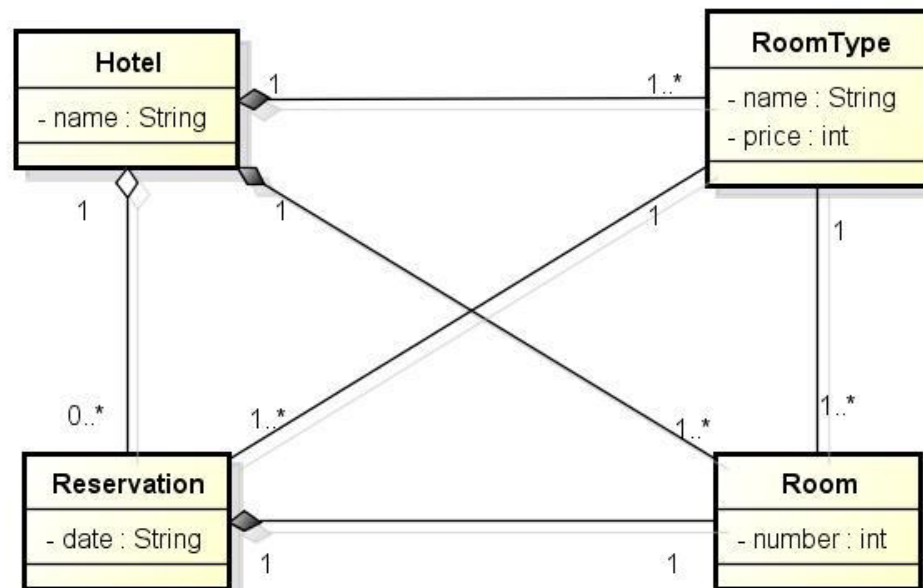
- 請畫出以下描述的Usecase Diagram
 - Reservation maker 可以用 Make a reservation 功能，但房間無法預定時，Make a reservation 會執行 Room not available。
 - Reservation maker 可以用 Cancel a reservation 功能，且 Cancel a reservation 一定會 Identify reservation。
 - Guest 可以用 Take up a reservation 功能，且 Take up a reservation一定會執行 Identify reservation。



Practice 2

- 請畫出以下描述的Class Diagram
 - Hotel 擁有自己的 name，每個Hotel都至少有一種 RoomType，其中 RoomType 擁有不同的name與price
 - 每個 Hotel 都至少會有一間 Room，其中 Room 擁有自己的number
 - 每種 RoomType 會對應至少一間 Room
 - Reservation 擁有自己的訂房日期(date)
 - 每個 Hotel 可能沒有或有多個 Reservation
 - 一個 Reservation 會對應一間 Room
 - 一種 RoomType 可以對應至少一個Reservation

pkgClass



UML Lab

- 繪製四題中的其中兩題即可
- 把匯出的圖檔放在word上，可在圖下方補充描述
- 每題都沒有標準的答案，只要畫的跟需求一樣就算是正確
- 繳交期限：2016/10/28
- 上傳到cyberccu2

UML Lab (cont.)

- 請繪製以下描述的Usecase Diagram
 - 假設你正在為某家銀行開發一個網路銀行系統，這個系統將提供銀行的客戶們直接利用網際網路從事線上的帳務管理。在初期階段，此系統必須提供客戶查詢帳戶、瀏覽交易歷史、轉帳、申請加入會員等功能。另外，使用者必須先行登入到系統才可以使用各項功能。註：我們假設轉帳是指一個人可以將其存款金額從一個帳戶轉移到另一個帳戶(我們假設使用者在這家銀行有兩個以上的帳戶)。請為這個系統畫出使用案例圖。

UML Lab (cont.)

●請繪製以下描述的Activity Diagram

●顧客訂購CD

- 顧客選擇其中的一個 CD 以查看更多的資訊
- 系統提供顧客 CD 的詳細資訊
- 顧客把 CD 加入到購物車
- 系統顯示購物車內容
- 顧客登入系統結帳
- 系統驗證顧客的信用卡資料
- 系統儲存訂單交易資訊
- 系統寄發訂單確認函(email)給顧客
- 系統顯示訂單內容以及交易明細
- 使用者登出

UML Lab (cont.)

- 請繪製以下描述的Class Diagram
 - A Role Play Game
 - 角色要有血量、武器、法術
 - 武器可以攻擊其他角色，讓其他角色損失血量
 - 武器包含槍(gun)、劍等多種
 - 法術包含治療跟傷害兩種，治療可以回復角色血量，傷害會使其他角色損失血量

UML Lab (cont.)

- 請繪製以下描述的Sequence Diagram
 - 到自動販賣機買商品
 - 客戶(Customer)投錢送出insert(amount)訊息給前台物件(Front), 參數amount是金額
 - 前台物件使用sendAmount(amount)訊息通知收銀機物件(Register)
 - 收銀機物件送出checkProduct(product)訊息給分配者物件(Dispenser), 參數是選擇的商品
 - 分配者物件是使用deliverProudct(product)訊息通知前台物件送出商品