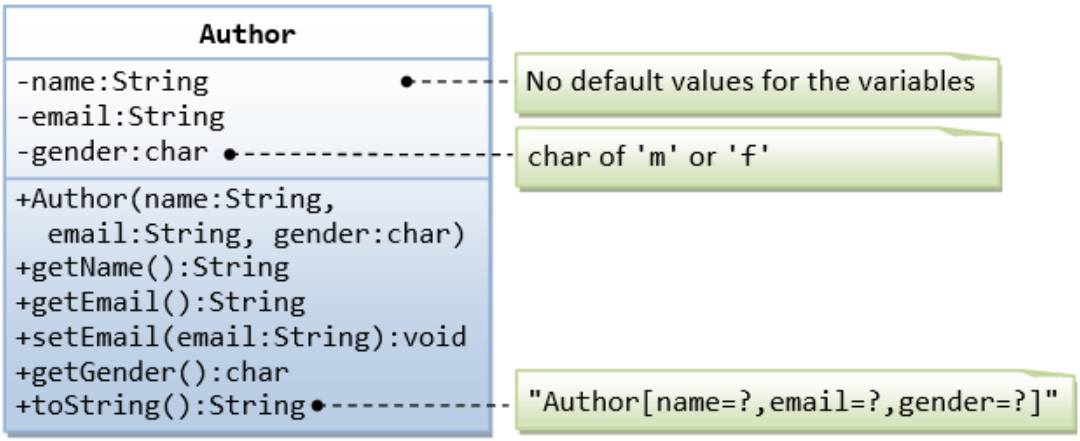
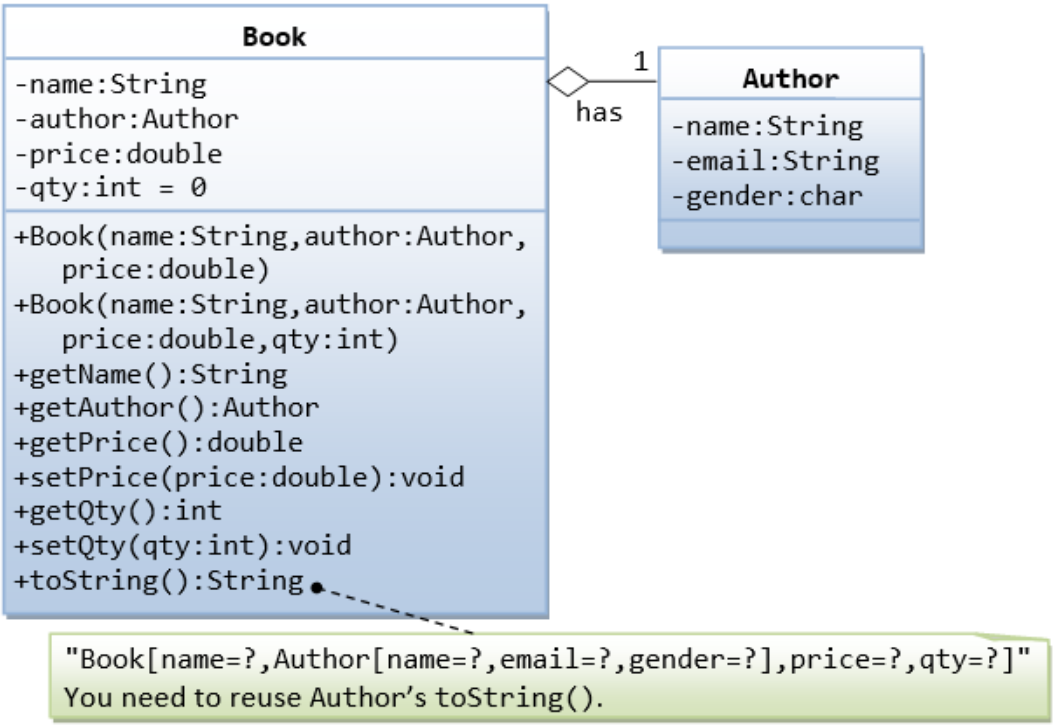
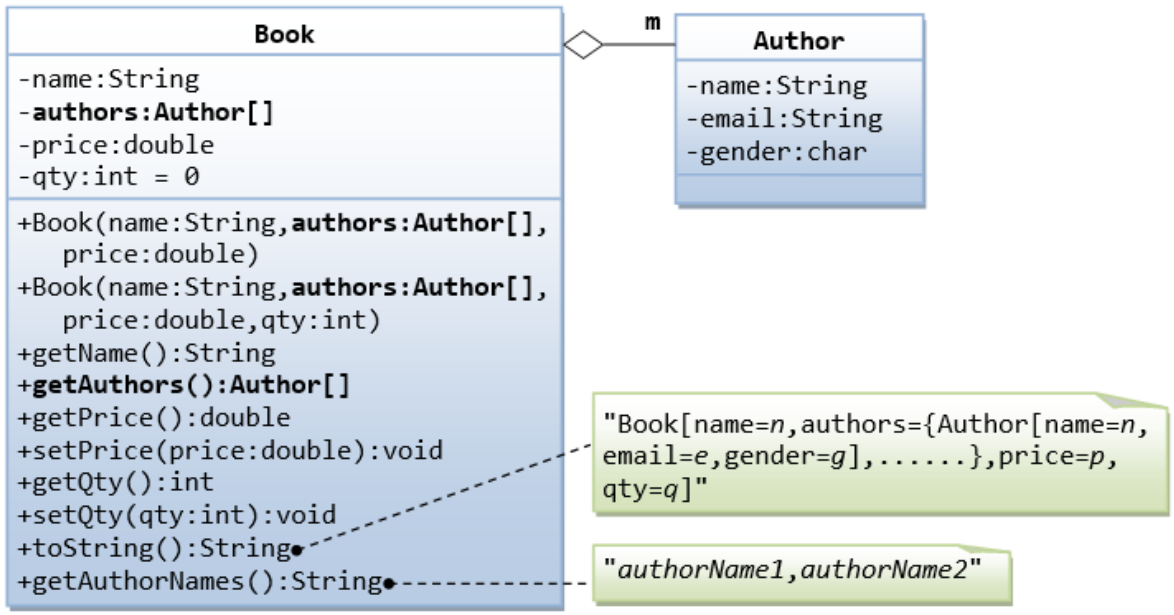
Bài 1:

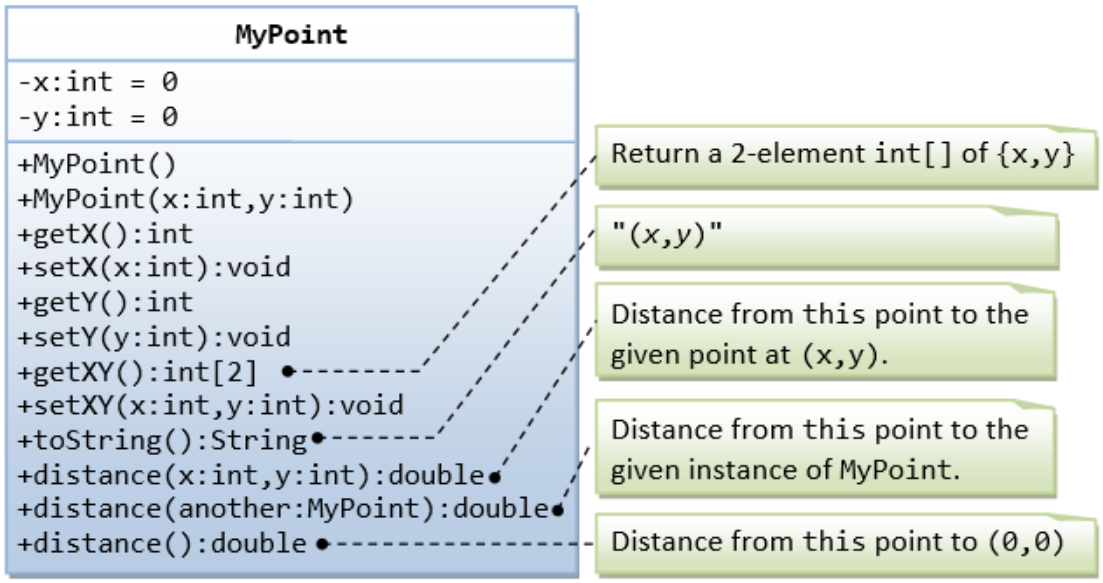

Author
-name:String
-email:String
-gender:char
+Author(name:String,
email:String, gender:char)
+getName():String
+getEmail():String
+setEmail(email:String):void
+getGender():char
+toString():String
No default values for the variables
char of 'm' or 'f'
"Author[name=?,email=?,gender=?]"

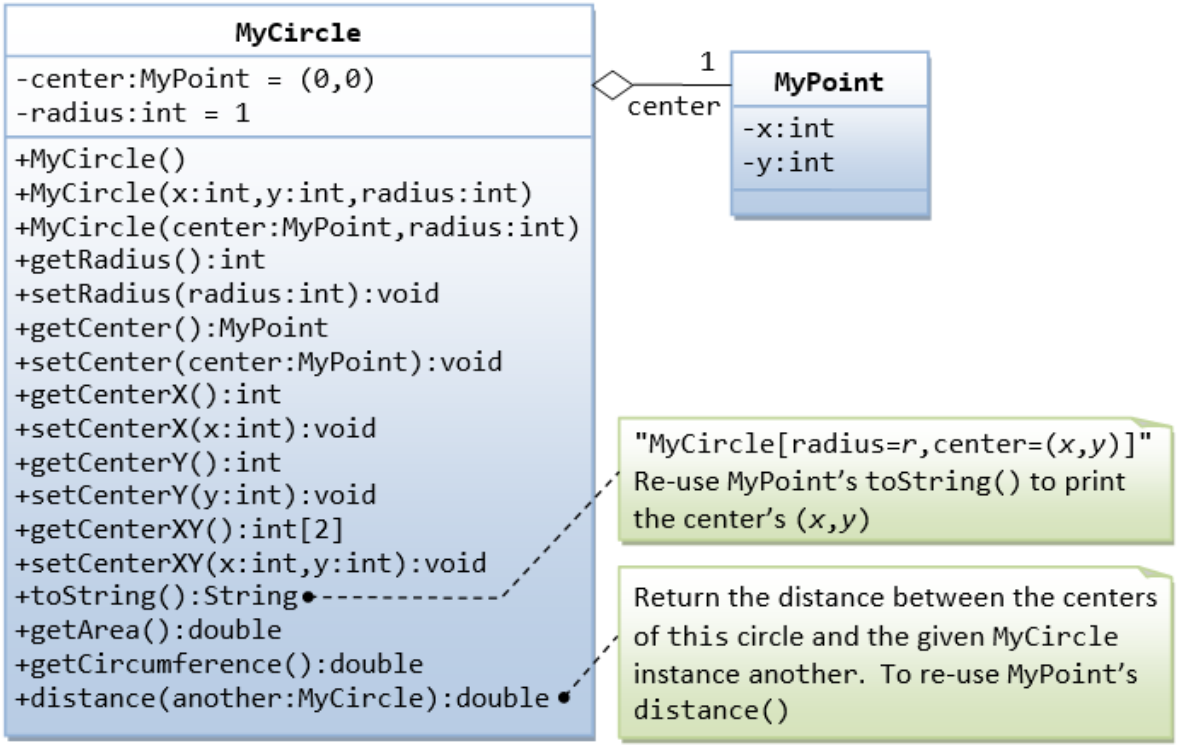

Book
-name:String
-author:Author
-price:double
-qty:int = 0
+Book(name:String,author:Author,
price:double)
+Book(name:String,author:Author,
price:double,qty:int)
+getName():String
+getAuthor():Author
+getPrice():double
+setPrice(price:double):void
+getQty():int
+setQty(qty:int):void
+toString():String
1
has
Author
-name:String
-email:String
-gender:char
"Book[name=?,Author[name=?,email=?,gender=?],price=?,qty=?]"
You need to reuse Author's toString().

Bài 2:

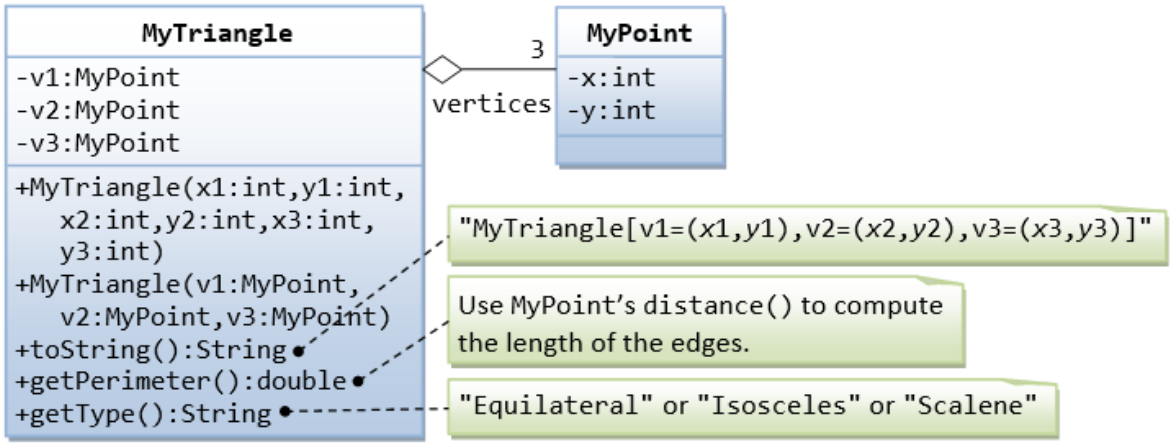

Book
-name:String
-authors:Author[]
-price:double
-qty:int = 0
+Book(name:String,authors:Author[],
price:double)
+Book(name:String,authors:Author[],
price:double,qty:int)
+getName():String
+getAuthors():Author[]
+getPrice():double
+setPrice(price:double):void
+getQty():int
+setQty(qty:int):void
+toString():String
+getAuthorNames():String
m
Author
-name:String
-email:String
-gender:char
"Book[name=n,authors={Author[name=n,
email=e,gender=g],.....},price=p,
qty=q]"
"authorName1,authorName2"

Bài 3:

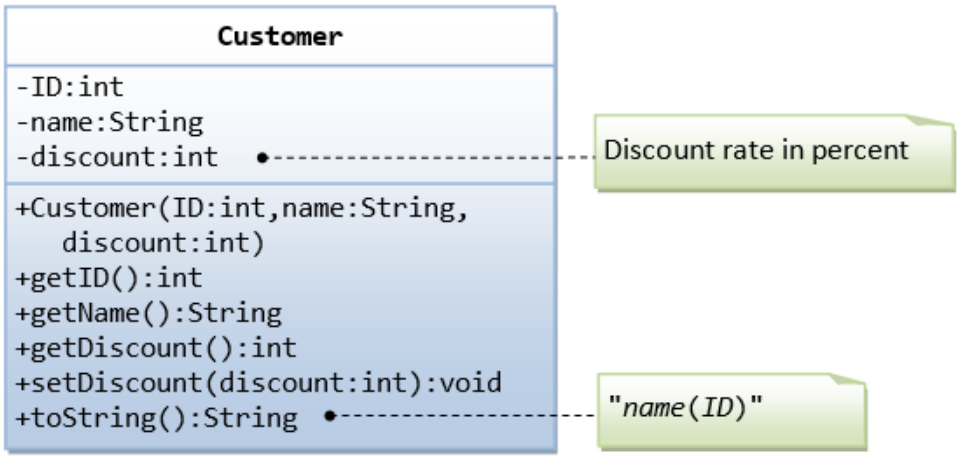

MyPoint
-x:int = 0
-y:int = 0
+MyPoint()
+MyPoint(x:int,y:int)
+getX():int
+setX(x:int):void
+getY():int
+setY(y:int):void
+getXY():int[2]
+setXY(x:int,y:int):void
+toString():String
+distance(x:int,y:int):double
+distance(another:MyPoint):double
+distance():double
Return a 2-element int[] of {x,y}
"(x,y)"
Distance from this point to the given point at (x,y).
Distance from this point to the given instance of MyPoint.
Distance from this point to (0,0)

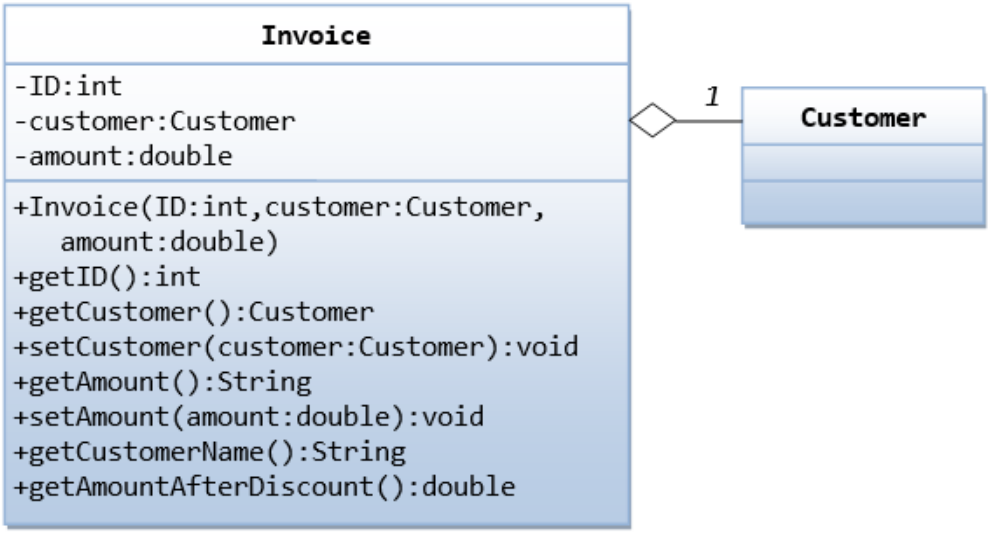

MyCircle
-center:MyPoint = (0,0)
-radius:int = 1
+MyCircle()
+MyCircle(x:int,y:int,radius:int)
+MyCircle(center:MyPoint,radius:int)
+getRadius():int
+setRadius(radius:int):void
+getCenter():MyPoint
+setCenter(center:MyPoint):void
+getCenterX():int
+setCenterX(x:int):void
+getCenterY():int
+setCenterY(y:int):void
+getCenterXY():int[2]
+setCenterXY(x:int,y:int):void
+toString():String
+getArea():double
+getCircumference():double
+distance(another:MyCircle):double
1
center
MyPoint
-x:int
-y:int
"MyCircle[radius=r,center=(x,y)]"
Re-use MyPoint's toString() to print the center's (x,y)
Return the distance between the centers of this circle and the given MyCircle instance another. To re-use MyPoint's distance()

Bài 4:

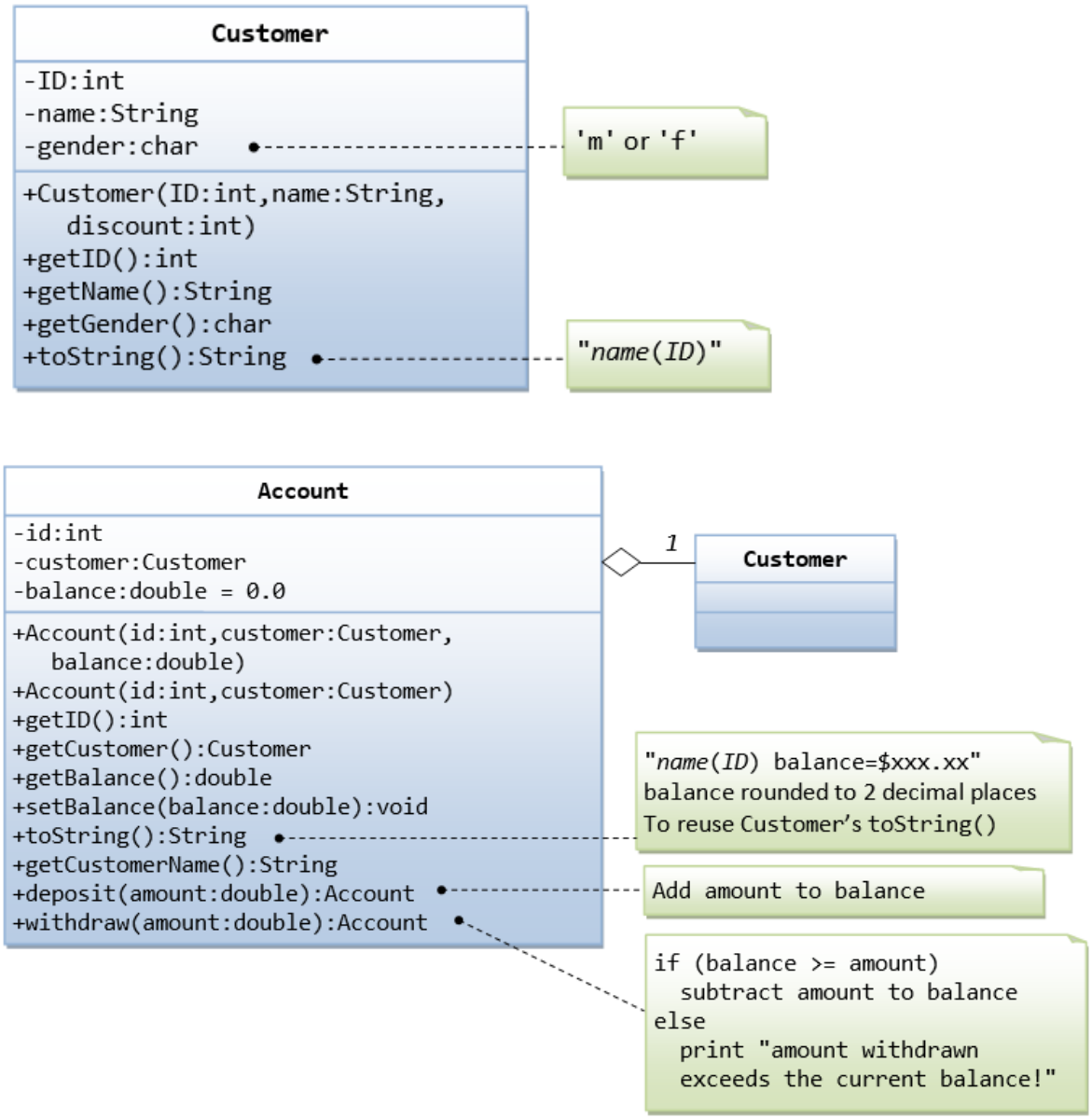

MyTriangle
-v1:MyPoint
-v2:MyPoint
-v3:MyPoint
+MyTriangle(x1:int,y1:int,
x2:int,y2:int,x3:int,
y3:int)
+MyTriangle(v1:MyPoint,
v2:MyPoint,v3:MyPoint)
+toString():String
+getPerimeter():double
+getType():String
3
vertices
MyPoint
-x:int
-y:int
"MyTriangle[v1=(x1,y1),v2=(x2,y2),v3=(x3,y3)]"
Use MyPoint's distance() to compute the length of the edges.
"Equilateral" or "Isosceles" or "Scalene"

Bài 5:


Customer
-ID:int
-name:String
-discount:int
+Customer(ID:int,name:String,
discount:int)
+getID():int
+getName():String
+getDiscount():int
+setDiscount(discount:int):void
+toString():String
Discount rate in percent
"name(ID)"


Invoice
-ID:int
-customer:Customer
-amount:double
+Invoice(ID:int,customer:Customer,
amount:double)
+getID():int
+getCustomer():Customer
+setCustomer(customer:Customer):void
+getAmount():String
+setAmount(amount:double):void
+getCustomerName():String
+getAmountAfterDiscount():double
1
Customer

Bài 6:


Customer
-ID:int
-name:String
-gender:char
'm' or 'f'
+Customer(ID:int,name:String,
discount:int)
+getID():int
+getName():String
+getGender():char
+toString():String
"name(ID)"
Account
-id:int
-customer:Customer
-balance:double = 0.0
1
Customer
+Account(id:int,customer:Customer,
balance:double)
+Account(id:int,customer:Customer)
+getID():int
+getCustomer():Customer
+getBalance():double
+setBalance(balance:double):void
+toString():String
+getCustomerName():String
+deposit(amount:double):Account
+withdraw(amount:double):Account
"name(ID) balance=$xxx.xx"
balance rounded to 2 decimal places
To reuse Customer's toString()
Add amount to balance
if (balance >= amount)
subtract amount to balance
else
print "amount withdrawn
exceeds the current balance!"