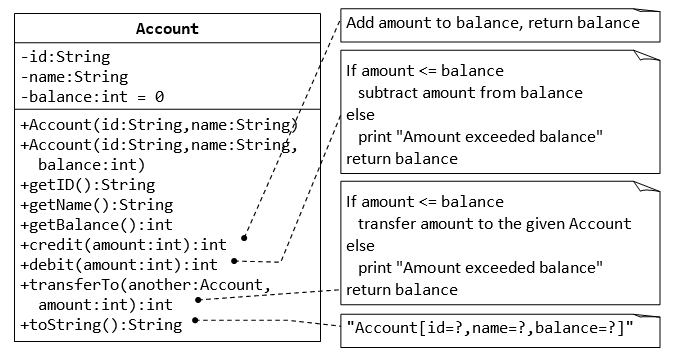
Core Java OOPS -Lab

Class and Object

1. 

**package** com.accountproject;

**public** **class** Account1 {

String id;

String name;

**int** balance=0;

**int** credit(**int** a1)

{

balance = balance + a1;

**return** balance;

}

**int** debit(**int** a2)

{

**if**(a2 <= balance)

{

balance = balance - a2;

}

**else**

{

System.***out***.println("Amount exceeded balance");

}

**return** balance;

}

**int** transfer(**int** a3)

{

**if**(a3 <= balance)

{

balance = balance - a3;

}

**else**

{

System.***out***.println("Amount exceeded balance");

}

**return** balance;

}

}

**package** com.accountproject;

**import** java.util.Scanner;

**public** **class** account2 {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**int** a1,a2,a3,res,res2,res3;

Scanner get = **new** Scanner(System.***in***);

Account1 ac=**new** Account1();

System.***out***.println("enter id:");

String id = get.nextLine();

System.***out***.println("enter name:");

String name=get.nextLine();

System.***out***.println("enter amout to credit:");

a1 = get.nextInt();

System.***out***.println("enter amout to debit:");

a2 = get.nextInt();

System.***out***.println("enter amout to transfer:");

a3 = get.nextInt();

res = ac.credit(a1);

System.***out***.println("after credited:" + res);

res2 = ac.debit(a2);

System.***out***.println("after debited:" + res2);

res3 = ac.transfer(a3);

System.***out***.println("after transfer:" + res3);

}

}

Output

enter id:

12

enter name:

kooor

enter amount to credit:

100

enter amount to debit:

300

enter amount to transfer:

40

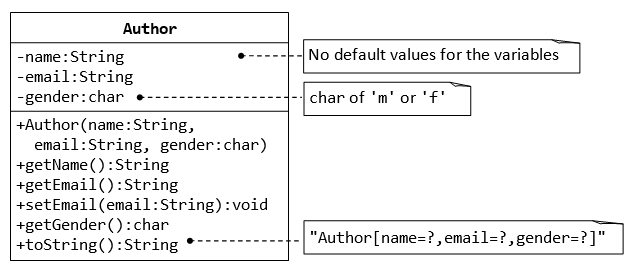
after credited:100

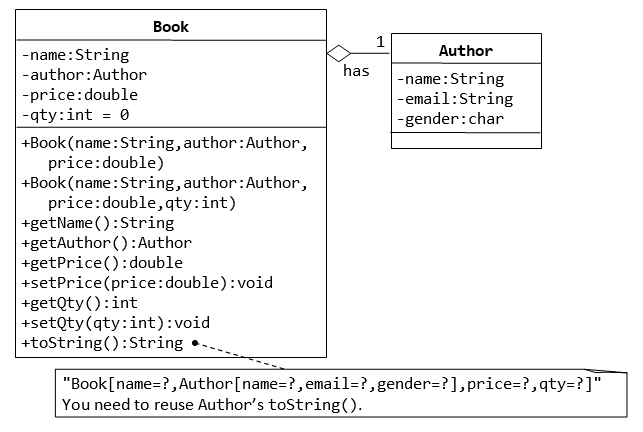
Amount exceeded balance

after debited:100

after transfer:60

#### 2. Composition- Author and Book Classes





**package** com.authorproject1;

**public** **class** Author

{

**static** String *name*;

**static** String *email*;

**static** **char** *gender*;

**public** Author(String name, String email, **char** gender)

{

**super**();

**this**.*name* = name;

**this**.*email* = email;

**this**.*gender* = gender;

}

**public** String getEmail()

{

**return** *email*;

}

**public** **void** setEmail(String email)

{

**this**.*email* = email;

}

**public** String getName()

{

**return** *name*;

}

**public** **char** getGender()

{

**return** *gender*;

}

@Override

**public** String toString()

{

**return** "Author [name= " + *name* + "email=" + *email* + "Gender= " + *gender* + "]";

}

}

**package** com.authorproject1;

**public** **class** Book **extends** Author

{

String bname;

**double** price;

**int** qty=0;

**public** Book(String bname)

{

**super**(*name*, *email*, *gender*);

**this**.bname=bname;

}

**public** **double** getPrice()

{

**return** price;

}

**public** **void** setPrice(**double** price)

{

**this**.price = price;

}

**public** **int** getQty()

{

**return** qty;

}

**public** **void** setQty(**int** qty)

{

**this**.qty = qty;

}

**public** String getBname()

{

**return** bname;

}

@Override

**public** String toString()

{

**return** "Book[name= " + bname + ",Author [name= " + *name* + ",email= " + *email* + ",Gender= " + *gender* +

"] , price= " +price + ",qty= " + qty + "]";

}

}

**package** com.authorproject1;

**public** **class** main

{

**public** **static** **void** main(String[] args)

{

Author a=**new** Author("koor","koormisharavoor@gmail.com",'F');

System.***out***.println(a);

Book b=**new** Book("7hugygy");

b.setPrice(800);

b.setQty(20);

System.***out***.println(b);

}

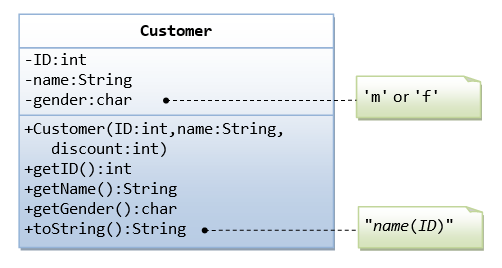
}

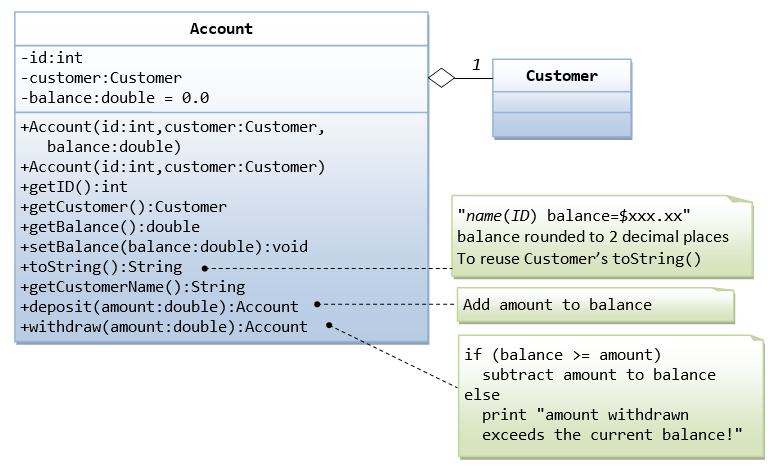
Output:

Author [name= kooremail=koormisharavoor@gmail.comGender= F]

Book[name= 7hugygy,Author [name= koor,email= koormisharavoor@gmail.com,Gender= F] , price= 800.0,qty= 20]

**3. The Customer and Account Class**





The Account class models a bank account, design as shown in the class diagram, composes a Customer instance (written earlier) as its member. Write the codes for the Account class and a test driver to test all the public methods.

**package** com.accountcustomer;

**import** java.lang.Math;

**public** **class** Account **extends** Customer

{

**int** aid;

**double** balance=0.0;

**public** Account( )

{

**super**(*id*, *name*, *gender*);

**this**.balance=balance;

}

**public** **double** getBalance()

{

**return** balance;

}

**public** **void** setBalance(**double** balance)

{

**this**.balance = balance;

}

**public** **int** getAid()

{

**return** aid;

}

**public** **void** credit(**double** amount)

{

balance = balance+amount;

System.***out***.println("After credit:");

}

**public** **void** withdraw(**double** amount)

{

**if**(balance>=amount)

{

balance=balance-amount;

System.***out***.println("After withdraw:");

}

**else**

{

System.***out***.println("amount withdrawn exceeds the current balance!");

}

}

@Override

**public** String toString()

{

**return** String.*format*("%.2f", balance);

}

}

**package** com.accountcustomer;

**public** **class** Account2 {

**public** **static** **void** main(String[] args)

{

Customer c=**new** Customer(1,"niha",20);

System.***out***.println(c);

Account a=**new** Account();

a.setBalance(890.367788);

a.credit(2000.3789);

System.***out***.println(a);

a.withdraw(899.2898);

System.***out***.println(a);

}

}

**package** com.accountcustomer;

**public** **class** Customer {

**static** **int** *id*;

**static** String *name*;

**static** **char** *gender*;

**static** **int** *discount*;

**public** Customer(**int** id, String name, **int** discount)

{

**super**();

**this**.*id* = id;

**this**.*name* = name;

**this**.*discount*=discount;

}

**public** **int** getId() {

**return** *id*;

}

**public** String getName() {

**return** *name*;

}

**public** **char** getGender() {

**return** *gender*;

}

@Override

**public** String toString()

{

**return** *name* + "( " + *id* + ")";

}

}

output

koor( 1)

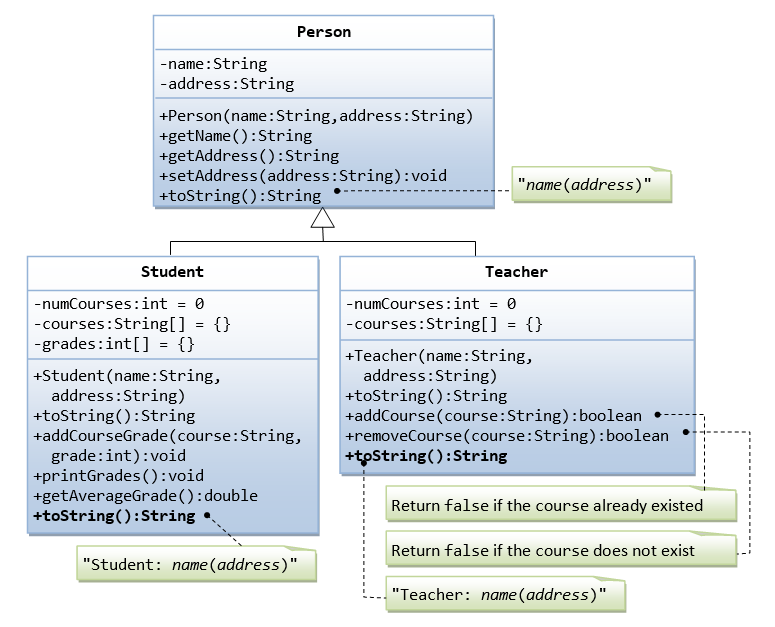
After credit:

3576.04

After withdraw:

2897.75

4. inheritance



Suppose that we are required to model students and teachers in our application. We can define a superclass called Person to store common properties such as name and address, and subclasses Student and Teacher for their specific properties. For students, we need to maintain the courses taken and their respective grades; add a course with grade, print all courses taken and the average grade. Assume that a student takes no more than 30 courses for the entire program. For teachers, we need to maintain the courses taught currently, and able to add or remove a course taught. Assume that a teacher teaches not more than 5 courses concurrently.

5.

