

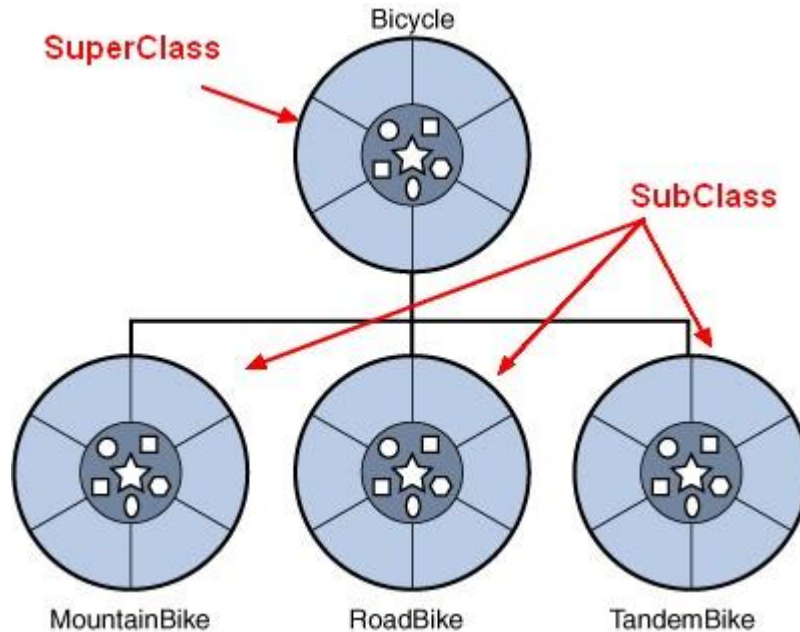
AP Computer Science



mr Hanley



Assignment Inheritance Practice (V2.0)



Refer to the hierarchy listed in the Textbook Big Java

Inheritance is about **Code Reuse** in java.

- Set up a BlueJ project called BankAccountPract(Yes, I said BlueJ!!!)
- Create a BankAccount class, SavingsAccount class, a CheckingAccount class and an AccountTest class)
- Unzip the [Big Java Source](#) to a folder on your computer
- Using BlueJ, make the Checking and Savings account classes children of the BankAccount class (use the inheritance arrow)
- Copy the logic from the unzipped source Chapter 11 to your 4 classes

- Run the account test after compiling all 4 files (Right click on the Square with the class name and choose void main(), since there are no parameters, just click ok to launch)
- Get your “feet wet” by modifying AccountTest as Follows;
 - **Comment out all the stuff inside the main**
 - **Create a SavingsAccount called s1 as follows;**
 - **SavingsAccount s1 = new SavingsAccount(5./12); //monthly rate**
 - **Add 1000 to the account via the deposit method**
 - **Write a for loop to call addInterest and print the balance of the account to the screen**
 - **You should see;**
 - **i = 1000.0**
 - **i = 1004.1666666666666**
 - **i = 1008.3506944444445**
 - **i = 1012.5521556712963**
 - **i = 1016.7711229865934**
 - **i = 1021.0076693323708**
 - **i = 1025.261867954589**
 - **i = 1029.5337924043997**
 - **i = 1033.823516539418**
 - **i = 1038.131114524999**
 - **i = 1042.4566608355199**
 - **i = 1046.8002302556679**
 - **Now, go back to the constructor for the SavingsAccount that takes in an interest rate**
 - **Toss in this line System.out.println(“\n%NSIDE SA, ONE ARG CON %\n”);**
 - **Do this for the BankAccount constructor as well**
 - **Run the program, notice the order of the constructors!!!!**
- Make the following changes to the hierarchy
 - Increase the allowable free monthly transactions for a checking account to be 10
 - Add a new type of bank account called the CollegeSavingsAccount
 - The CollegeSavingsAccount takes in an initial amount, an interest rate (%) and a months to college count into the Constructor
 - Each month if the balance is above \$10,000 then the interest earned is an additional .25% (monthly) above the rate

- If an attempt to withdraw funds is made before the months to college has expired, the holder forfeits 50% of the balance
- Each time **addInterest()** is called, consider this a month of time passing. You will have to use a loop to keep adding interest to test to make sure that the withdraw method is used only after enough months have elapsed
- **HINT** Look at the TimeDeposit account in the source code to see how to do this**
- Add social security number and name to all accounts
 - Provide appropriate constructors so;
 - **An account can be created with**
`BankAccount b1 = new BankAccount("Sara Smith","123456789",5000); //makes account with name, ss, initial balance`
 - **A SavingsAccount can be created with**
`SavingsAccount s1 = new SavingsAccount("Luke Skywalker","444667777",.5); //.5 = MONTHLY percent rate = X12 = 6% annual rate (NOTE: accounts compounded on last day of month)`
 - **A CheckingAccount can be created with**
`CheckingAccount c1 = new CheckingAccount("Steve Ballmer","100100100",1000); //1000=initial balance`
 - **CollegeSavingsAccount can be created with**
`CollegeSavingsAccount cs1 = new CollegeSavingsAccount(1000, .25,72); //1000 = intital balance, .25 = MONTHLY savings rate(%) X 12 = 3% annual rate, 72 = months to college`
 - **CollegeSavingsAccount can also be created with**
`CollegeSavingsAccount cs1 = new CollegeSavingsAccount("Blaise Pascal","987654321",1000, .75, 72); //Name, SS# and 1000 = intital balance, .75 = MONTHLY savings rate(%)X 12 = 9% annual rate, 72 = months to college`
- Write an appropriate test class to test these features

Project Name	BankAccountPract
Class 1 Name	BankAccount
Class 2 Name	CheckingAccount
Class 3 Name	SavingsAccount
Class 4 Name	CollegeSavingsAccount
Class 5 Name	AccountTester

Rubric	
Increase free check trans to 10	5
Add SS and Name to all accounts (including appropriate constructors)	20
CollegeSavingsAccount can be constructed	10
CollegeSavingsAccount prevents withdraws til maturity	15
CollegeSavingsAccount gives a boost of interest rate by .25% if ending monthly balance >=10,000	10
Comments + Tester class	10
TOTAL	70

Recursion*Linear Search*Binary Search*Grid World Case Study*File Processing *nlogn*Hangman