1. Write a complete Java program called AtmSimulator (not ATMSimluator!) that simulates a simple one-transaction ATM according to the following guidelines.

The program should start with an initial account balance, which you can set to any legitimate double value. Prompt the user with the following prompt (without the dashed lines).

Enter the number of your desired transaction type.

1. Balance
2. Deposit
3. Withdrawal
4. Quit
* If a balance is requested, the program should output “Your current balance is X.” where X is the initial balance, and then stop.
* If a deposit is requested, prompt the user to enter the amount of the deposit (use a double for this). Add the deposit amount to the initial balance and then print “Your current balance is X.” where X is the new balance after the deposit, and then stop.
* If a withdrawal is requested, prompt the user to enter the amount of the withdrawal (use a double for this). If the proposed withdrawal amount is less than or equal to the initial balance,  print “Your current balance is X.” where X is the new balance after the withdrawal, and then stop. If the proposed withdrawal amount exceeds the initial balance, print “Insufficient funds. Your current balance is X.” where X is the initial balance, and then stop.
* If “Quit” is requested, the program should print “Good-bye.” and then stop.

Thoughts

* Before you begin coding, sketch out the logic that will handle each condition appropriately.
* Note that this simple ATM simulator handles, for simplicity, just a single transaction of Balance, Deposit, or Withdrawal.

2. Write a complete Java program called IncomeTax that prompts a user for his or her marital status (single or married) and annual income (a double value), and then uses nested if statements to compute that person's tax, based on the simplified tax-rate table below.

Output of your program should look close to the following and should be constructed using both String literals and variables for income, tax rate, and income tax.

Based on a status of "single" and an annual income of $10000.00,

the income tax is 15.00% of $10000.00, which equals $1500.00.

|  |
| --- |
| Simplified Tax-Rate Table |
| **Income** | **Marital Status - Single** | **Marital Status - Married** |
| <= $30,000 | 15% | 12% |
| > $30,000 | 25% | 20% |

Thoughts

* Note the requirement for this program to use nested if statements. This requirement is intended to help you build some logic with good formatting.
* Use comments to help readers understand your code.

3. Write a complete Java program called StringReverse that:

1. prompts the user at the command line for one 3-character string
2. then (after the user inputs that first 3-character string) prompts the user for another 3-character string
3. then prints out the two input strings with a space between them
4. and finally prints on a separate line the two input strings 'in reverse' (see example below) with a space between them.

So, for example, if the first string is 'usr' and the second string is 'bin' your program would output something like the following:

The two strings you entered are: usr bin.

The two strings in reverse are: nib rsu.

Note that the reversed SECOND string comes FIRST when printing the strings in reverse.

See Horstmann pp. 62-63 for some ideas. Make sure your program includes the command line prompts for the user and that it formats the output appropriately.

4. Write a complete Java program called PasswordChecker that gets a String from the user at the command line and checks whether the String, called inputPassword, conforms to the following password policy.

The password must:

* be 3 characters in length
* include at least one uppercase character
* include at least one digit

If the password conforms to the policy, output "The provided password is valid." Otherwise, output "The provided password is invalid. Please try again." A loop is not needed for this program.