

Account class

The following program implements an **Account** class and two other classes that inherit from that class.

Account.java

```
1 public class Account {
2     protected double balance;
3     protected String owner;
4
5     public Account() {
6         this.owner = "John Doe";
7         this.balance = 100;
8     }
9     public Account(String owner){
10        this.owner=owner;
11        balance = 0;
12    }
13    public Account(String owner, double balance){
14        this.owner =owner;
15        this.balance = balance;
16    }
17    public double deposit (double value){
18        if (value >= 0){
19            balance+= value;
20        }
21        return balance;
22    }
23    public double withdraw (double value){
24        if (value >= 0 && (balance - value) >= 0){
25            balance -= value;
26            return value;
27        }
28        else{
29            return 0;
30        }
31    }
32    public String toString(){
33        return String.format(
34            "%s, balance: %.2f$", owner, balance);
35    }
36 }
```

CheckingAccount.java

```
1 public class CheckingAccount extends Account{
2
3     private double overdraft = -500.00;
4
5     //allow an overdraft on the account
6     public double withdraw (double value){
7         if (balance - value > overdraft) {
8             balance -= value;
9             return value;
10        }
11        else
12            return 0;
13    }
14 }
```

SavingsAccount.java

```
1 public class SavingsAccount extends Account {
2     private double annualRate = 0.001;
3
4     public SavingsAccount(String owner,
5                             double balance, double interest){
6         super(owner, balance);
7
8         if(interest > 0 ){
9             annualRate = interest;
10        }
11    }
12
13    public void compoundMonthly(){
14        double interestEarned = balance * (annualRate/12);
15        balance += interestEarned;
16    }
17 }
```

Questions:

1. List all the data fields in the **Account** class.
2. List all the data fields in the **CheckingAccount** class.
3. List all the data fields in the **SavingsAccount** class.
4. List all the methods that are available in the **CheckingAccount** class.
5. List all the methods that are available in the **SavingsAccount** class.
6. How many constructors are in the **CheckingAccount** class? How many are in the **SavingsAccount** class?
7. What does the line 6 in **SavingsAccount** class do?
8. Consider the following code lines. Determine what each line does and if there are any problems with it.

```
1
2 Account a = new Account ("James", 2500);
3 SavingsAccount sa = new SavingsAccount ("Rose", 1587, 0.02);
4 CheckingAccount ca = new CheckingAccount ( );
5 CheckingAccount ca1 = new CheckingAccount ( "Carol", 420 );
6 Account a1 = new SavingsAccount ( "Joe", 3500, 0.01);
7 Account a2 = new CheckingAccount ( );
8 SavingsAccount sa1 = new CheckingAccount();
9 SavingsAccount sa2 = new Account ("Amy", 12000);
10
11 a.deposit(200);
12 sa.deposit(300);
13 ca.deposit(100);
14
15 sa.compoundMonthly();
16 a1.compoundMonthly();
17 ca.compoundMonthly();
18
19 ca.withdraw(100);
20 a2.withdeaw(100);
21
```