Classes, casting, and enums review quiz *(not collected)*

1. What is printed after the following code is executed?

```
double myNum = 2.718;
int myInt = (int) myNum;
System.out.println(myNum + ", " + myInt);
```

2. What is printed after the following code is executed? What do you think the programmer intended?

```
public enum Day { MON, TUE, WED, THU, FRI, SAT, SUN };
...
// we will talk more about this for-each loop
for (Day today : Day.values()) {
    switch (today) {
        case MON:
        case TUE:
        case WED:
        case THU:
        case FRI:
            System.out.println("work");
        default:
            System.out.println("play");
    }
}
```

3. The following code contains a skeleton of a `BankAccount` class. Fill in the missing code so that the class as one *instance variable*, the `balance` of the account. Someone with a bank account should be able to deposit (add money to their account), and withdraw (retrieve money from their account).

   Here main is used for some small tests, but typically the application entry point would be in a different file.

```
public class BankAccount {
    // declare an instance variable here


    public BankAccount(double initialBalance) { // finish constructor


    }

    public BankAccount() { // default constructor with initial balance 0


    }

    public _______ getBalance() { // return the balance


    }

    public _______ deposit(double amount) {


    }

    public _______ withdraw(double amount) {







    }

    public static void main(String[] args) {
        BankAccount myAccount = new BankAccount();
        System.out.println("balance: " + myAccount.getBalance());
        myAccount.deposit(50);
        System.out.println("balance: " + myAccount.getBalance());
        myAccount.withdraw(75);
        System.out.println("balance: " + myAccount.getBalance());
    }
}
```