Classes and Objects

- Class—non primitive data type. Example: String
- Class has methods, and data.
- Can create objects of a class. Eg: String s = "Hi"
- Can perform method on an object. Eg: s.length()
Creating a Class

A class to represent a Bank Account

- Identify the methods.
- Provide specification for the methods
  - What does the method do?
  - What are the parameters?
  - What is the type of the method
- implement the method.
  - How? Code for method
  - What should be the data?
  - Constructors
Methods of Bank Account

- deposit
- withdraw
- getBalance
- getName
- getAccountType
Specifications

deposit

- Deposits money into the bank account.
Specifications

deposit

- Deposits money into the bank account.
  - Parameter?
Specifications

deposit

- Deposits money into the bank account.
  - Parameter?
  - amount to deposit.
Specifications

deposit

- Deposits money into the bank account.
  - Parameter?
  - amount to deposit.
  - Type of the parameter?
Specifications

deposit

- Deposits money into the bank account.
  - Parameter?
  - amount to deposit.
  - Type of the parameter?
  - double
Specifications

**deposit**

- Deposits money into the bank account.
  - Parameter?
  - amount to deposit.
  - Type of the parameter?
  - double
  - Type of the method?
Specifications

deposit

- Deposits money into the bank account.
  - Parameter?
  - amount to deposit.
  - Type of the parameter?
  - double
  - Type of the method?
  - void
Specifications

**deposit**

- Deposits money into the bank account.
  - Parameter?
  - amount to deposit.
  - Type of the parameter?
  - double
  - Type of the method?
  - void

**Method Header**

```java
public void deposit(double amount)
```
Specifications of Bank Account

getBalance

- Gets the current balance of the account
Specifications of Bank Account

getBalance

- Gets the current balance of the account
- Parameter?
getBalance

- Gets the current balance of the account
- Parameter?
- None
Specifications of Bank Account

getBalance

- Gets the current balance of the account
  - Parameter?
  - None
  - Type of the method?

**Public double getBalance()**
Specifications of Bank Account

getBalance
- Gets the current balance of the account
- Parameter?
  - None
- Type of the method?
  - double
Specifications of Bank Account

getBalance

- Gets the current balance of the account
  - Parameter?
  - None
  - Type of the method?
  - double

public double getBalance()
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
- Parameter?

public void withdraw(double amount)
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
  - Parameter?
  - amount to withdraw.
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
  - Parameter?
  - amount to withdraw.
  - Type of the parameter?

\[ \text{public void withdraw(double amount)} \]
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
  - Parameter?
  - amount to withdraw.
  - Type of the parameter?
  - double
  - Type of the method?
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
  - Parameter?
  - amount to withdraw.
  - Type of the parameter?
  - double
  - Type of the method?
  - void
Specifications of Bank Account

withdraw

- Withdraws money from the bank account.
  - Parameter?
  - amount to withdraw.
  - Type of the parameter?
  - double
  - Type of the method?
  - void

public void withdraw(double amount)
data

What should be the instance variables of the class?

- name, balance, account type

What are the types of the instance variables?

private String name;
private double balance;
private String accountType;

Provide implementation details for the methods.
Constructors

What should be the initial values of (instance variables) of the Objects?

```java
public BankAccount(String nameP)
public BankAccount(String nameP, double balanceP)
public BankAccount(String nameP, double balanceP, String accountTypeP)
```

No return type for constructors.
Using/Testing the class BankAccount

```java
BankAccount b = new BankAccount("Harry", 100);
b.withdraw(50);
b.deposit(200);
System.out.println(b.getBalance());
```

Creates a BankAccount object.