Assigned in: Lesson 2

Due before: Lesson 3

Upload your finished lab to the Learning Hub (Activities / Lab 1) before the start of lesson 3.

For this lab we will create a hierarchy of several classes: IDevice (the parent), IPod, IPad, and IPhone.

The parent class – IDevice – has a non-abstract method called **getPurpose()** which returns its purpose (a String). The “purpose” instance variable is set in the constructor. The IDevice class also has an abstract method called **printDetails()** which prints all of the child class’s instance variables.

Child classes of IDevice:

* IPod: the purpose of this iDevice is “music”
* IPad: the purpose of this iDevice is “learning”
* IPhone: the purpose of this iDevice is “talking”

Note: the child classes also contain instance variables, constructor parameters, accessor methods, and mutator methods for several other data members:

* IPod: (int) number of songs stored, (double) maximum volume in decibels
* IPad: (boolean) has a case, (String) operating system version
* IPhone: (double) number of minutes remaining on phone plan, (String) carrier

Also, each of these four classes also overrides the **toString()** method to return all of the instance variables in a single String. Use the @Override annotation. Child classes’ **toString()** methods must also call their parent’s toString() method.

Furthermore, each of these four classes also overrides the **equals()** method. IPods with the same number of songs stored are considered equal; IPads with the same operating system version are considered equal; IPhones which the same amount minutes as each other are considered equal. Use the @Override annotation when overriding **equals()**.

Remember to override the **hashCode()** method too for each class, whenever you override **equals()**.

Continuing from above, add more data and methods as described below.

Extend the IPhone class; it has a child called IPhoneFifteen.

The IPhoneFifteen class also contains instance variables, constructor parameters, accessor methods, and mutator methods for several other data members:

* (boolean) high-resolution camera
* (int) gigabytes of memory

Also, the IPhoneFifteen class overrides the **toString()** method to return all of the object data in a String. Use the @Override annotation. This **toString()** method must also call its parent’s **toString()** method.

Furthermore, this class also overrides **equals()** (and therefore also **hashCode()**). IPhoneFifteen objects that have the same amount of minutes remaining on their phone plan are considered equal, but only if they also have the same value for “high-resolution camera”. Use the @Override annotation.

Remember to override **hashCode()** properly too for this class.