Set 16: Objects & Classes

Skill 16.01: Describe the purpose of objects and classes
Skill 16.02: Write code to create a class
Skill 16.03: Differentiate between and object and a class
Skill 16.04: Pass parameters to a constructor

Skill 16.01: Describe the purpose of objects and classes

Skill 16.01 Concepts

So far, we have learned about several different data types. For example, *int*, *double*, *String*, and *boolean*. We can use these different data types to store things like a student's name, their grade level, their gpa, or whether or not they have a scholarship. But what if we wanted to do this for thousands of students? Before we answer this questions, check out the video below,

https://hpluska.github.io/APCompSciA/notes/videos/WhatAreClassesAndObjects.mp4

Classes in java allow us to model other data types that are not built into the the programming language and create as many subsequent objects of the class as needed. For example, we could create a data type called *Student* which could then store the name, gradeLevel, and GPA of different students. Below, is an illustration of two Student data types: student1 and student2. For each Student data type we have also defined the name, gradeLevel, GPA, and hasScholarship variables.

Student student1			Student student2		
data type	name	value	data type	name	value
String	name	Bart	String	name	Kyle
int	gradeLevel	10	int	gradeLevel	11
double	GPA	2.6	double	GPA	3.5
boolean	hasScholarship	false	boolean	hasScholarship	true

In the above example *Student* is referred to as a class. *student1* and *student2* are referred to as Student objects, or *instances* of the Student class.

A **class** is used to model a data type.

An **instance** of a class is referred to as an **object**.

Skill 16.01: Exercise 1

Skill 16.02: Write code to create a class

Skill 16.02 Concepts

All classes in java are created using the *class* key word. For example,

```
public class Student{

}
```

To properly describe the students above we need the following *instance* variables: *name*, *gradeLevel*, *GPA*, and *hasScholarship*. *Instance* variables are variables that can be accessed by any memeber of the class. For this reason they are declared at the top of the class.

The *instance* variables necessary to describe our students have been added to our class below.

```
public class Student{

     public String name;
     public int gradeLevel;
     public double GPA;
     public boolean hasScholarship;

}
```

In the above code snippet, the key word *public* is an access level modifier that allows other classes to interact with this class and the variables inside it. For now, all classes and variables will be public.

But, before we can start creating student data types, we need to add one more thing to our class, a *constructor*,

```
public class Student{

     public String name;
     public int gradeLevel;
     public double GPA;
     public boolean hasScholarship;

     public Student(){

     }
}
```

Instance variables

Constructor

A constructor is required for all classes. In fact, if you do not specify one, java will include one for you! But, the above code is all that is needed to start creating *Student* data types.

Skill 16.02: Exercise 1

Skill 16.03: Differentiate between an object and a class

Skill 16.03 Concepts

Instantiating is a fancy word for *creating*. If you *instantiate* a class you are creating an *object* (or data type) of that class. In a previous example we defined a *Student* data type called *student1* and another *Student* data type called *student2*. A Student *object* for each of these students can be created by instantiating the Student class.

To create Student objects from the Student class we use the *new* key word as illustrated below.

```
public class StudentMaker{

     public static void main(String args[]){

          Student student1 = new Student();
          Student student2 = new Student();
     }
}
```

The code above created two *instances* of the *Student* class called *student1* and *Student2*. This is just another way of saying we created two student objects, *student1* and *student2*.

Notice, that we created our *Student* objects in the main method of a *different* class called *StudentMaker*. We were able to do this because the student class was given the *public* access modifier. Below are the two classes side by side,

Student

```
public class Student{

    public String name;
    public int gradeLevel;
    public double GPA;
    public boolean hasScholarship;

    public Student(){

    }
}
```

StudentMaker

```
public class StudentMaker{

    public static void main(String args[]){

        Student student1 = new Student();
        Student student2 = new Student();
    }
}
```

Now, that we have created our students, we can start defining their characteristics, like their name, gradeLevel, GPA, etc. To access variables associated with our student object we use the *dot* notation,

Student

```
public class Student{

    public String name;
    public int gradeLevel;
    public double GPA;
    public boolean hasScholarship;

    public Student(){

    }
}
```

StudentMaker

```
public class StudentMaker{

    public static void main(String args[]){

        Student student1 = new Student();
        student1.name = "Bart";
        student1.gradeLevel = 10;
        student1.GPA = 2.6;
        student1.hasScholarship = false;
    }
}
```

Skill 16.03: Exercise 1

Skill 16.04: Pass parameters to a constructor

Skill 16.04 Concepts

A *parameter* is a value you can pass to a method (or a constructor). For example, rather than defining the students *name*, *gradeLevel*, *GPA*, etc. after the *Student* object has been created, this can be done all at once. Before we can do so however, we must modify the constructor in the *Student* class to allow for parameters.

```
public class Student{

     public String name;
     public int gradeLevel;
     public double GPA;
     public boolean hasScholarship;

     public Student( String n, int g, double gpa, boolean s ){

     }
}
```

Once the values are received by the constructor, they can be assigned to the variables *name*, *gradeLevel*, *GPA*, and *hasScholarship* as follows,

```
public class Student{

     public String name;
     public int gradeLevel;
     public double GPA;
     public boolean hasScholarship;

     public Student( String n, int gl, double gpa, boolean s ){

          name = n;
          gradeLevel = gl;
          GPA = gpa;
          hasScholarship = s;
     }
}
```

Now, that we have modified the constructor, we must also modify our code to create our *Student* instances. Below, we create two *Student* instances, *student1* and *student2*,

```
public class StudentMaker{

     public static void main(String args[]){

          Student student1 = new Student("Bart", 10, 2.6, false);
          Student student2 = new Student("Kyle", 11, 3.5, true);
     }
}
```

Skill 16.04: Exercises 1 thru 3