```java
/*
 * Code1.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A very basic Hello, World application.
 *
 */

class Code1 {

    public static void main(String [] args) {
        System.out.print("Hello, World!");
        System.out.println();
    }

}
```

```java
/*
 * Code2.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Defining and assigning values to fields and local variables.
 *
 */

class Code2 {

    // define a field
    private static int num;

    public static void main(String [] args) {
        // local variable
        int anotherNum = 5;

        // assigning a value to a variable
        num = 2;

        // print out some information
        System.out.println("num: " + num);
        System.out.println("anotherNum: " + anotherNum);
    }

}
```

```java
/*
 * Code3.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Assigning values to various types of variables.
 *
 */

class Code3 {

    public static void main(String [] args) {

        int number = 123;
        String str = "Hello, World!";

        // print out an integer
        System.out.println("number: "+number);

        // print out a string
        System.out.println("str: "+str);

        // print out a substring, a method of the String class
        // a list of methods is available from:
        // http://java.sun.com/javase/6/docs/api/java/lang/String.html
        System.out.println("substring: "+str.substring(7, 12));

    }

}
```

```java
/*
 * Code4.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Changing variable type through typecasting.
 *
 */

class Code4 {

    public static void main(String [] args) {

        int myInt = 123;
        double myDouble = 123.0;
        String myStr = "123";
        String txtStr = "Hello, World!";

        // Integer math, is there a problem?
        System.out.println("myInt/5: " + (myInt / 5));

        // modulus
        System.out.println("myInt%5: " + (myInt % 5));

        // Double math, that's better
        System.out.println("myDouble/5: " + (myDouble / 5));

        // Double math by type casting an int
        System.out.println("(double)myInt/5: "+( (double)myInt / 5));

        // Another type-caste by forcing double math
        System.out.println("myInt/5.0: "+(myInt / 5.0));

        // Attempt to convert a string to int
        //System.out.println("Converting myStr to int: "+ (int)myStr);

        // Convert a string to an int
        System.out.println("Converting myStr to int: "+Integer.parseInt(myStr));

        // Convert another string to an int
        //System.out.println("Converting txtStr to int: "+Integer.parseInt(txtStr));
    }

}
```

```java
/*
 * Code5.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Reading values from the keyboard, if statements, boolean expressions.
 *
 */

// allow us use of the keyboard scanner. More information from the docs:
// http://java.sun.com/javase/6/docs/api/java/util/Scanner.html
import java.util.Scanner;

class Code5 {

    public static void main(String [] args) {

        int number = 123;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        // wait for the user to enter an integer
        int input = keyboard.nextInt();

        // test to see if what the user entered matches our number.
        if(input == number) {
            System.out.println("Numbers match! :-)");
        } else {
            System.out.println("Numbers do not match! :-(");
        }

    }

}
```

```java
/*
 * Code6.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A better version of reading values from the keyboard.
 *
 */

// allow us use of the keyboard scanner. More information from the docs:
// http://java.sun.com/javase/6/docs/api/java/util/Scanner.html
import java.util.Scanner;

class Code6 {

    public static void main(String [] args) {

        int number = 123;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        System.out.print("Please enter an integer: ");

        // wait for the user to enter an integer
        int input = keyboard.nextInt();

        // test to see if what the user entered matches our number.
        if(input == number) {
            System.out.println("Numbers match! :-)");
        } else {
            System.out.println("Numbers do not match! :-(");
        }

    }

}
```

```java
/*
 * Code7.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A better version of reading values from the keyboard, with exception
 * handling.
 *
 */

// allow us use of the keyboard scanner. More information from the docs:
// http://java.sun.com/javase/6/docs/api/java/util/Scanner.html
import java.util.Scanner;

class Code7 {

    public static void main(String [] args) {

        int number = 123;
        int input = 0;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        System.out.print("Please enter an integer: ");

        // try inputting an integer, if a user doesn't ..
        try {

            input = keyboard.nextInt();

        } catch(Exception e) {
            // an exception will be thrown, and we can catch it to alert
            // the user that something bad happened.

            System.out.println("Invalid input! Quitting..");
            System.exit(1);
        }

        // test to see if what the user entered matches our number.
        if(input == number) {
            System.out.println("Numbers match! :-)");
        } else {
            System.out.println("Numbers do not match! :-(");
        }

    }

}
```

```java
/*
 * Code8.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Comparing input via a switch.
 *
 */

// allow us use of the keyboard scanner. More information from the docs:
// http://java.sun.com/javase/6/docs/api/java/util/Scanner.html
import java.util.Scanner;

class Code8 {

    public static void main(String [] args) {

        int input = 0;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        System.out.print("Please enter an integer: ");

        // try inputting an integer, if a user doesn't ..
        try {

            input = keyboard.nextInt();

        } catch(Exception e) {
            // an exception will be thrown, and we can catch it to alert
            // the user that something bad happened.

            System.out.println("Invalid input! Quitting..");
            System.exit(1);
        }


        // determine which number the user selected
        switch(input) {
            case 1: System.out.println("You're number one!"); break;
            case 3: System.out.println("Third time's a charm!"); break;
            case 6: System.out.println("That matches my file name!"); break;
            default: System.out.println("That's a boring number.");
        }

    }

}
```

```java
/*
 * Code9.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * String input and comparison
 *
 */

// allow us use of the keyboard scanner. More information from the docs:
// http://java.sun.com/javase/6/docs/api/java/util/Scanner.html
import java.util.Scanner;

class Code9 {

    public static void main(String [] args) {

        String str = "Hello, world";
        String input = null;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        System.out.print("Please type a string: ");

        // wait for the user to enter an integer
        try {
            input = keyboard.nextLine();
        } catch(Exception e) {
            System.out.println("Invalid input! Quitting..");
            System.exit(1);
        }

        // test to see if what the user entered matches our string.
        if(str == input) {
            System.out.println("Strings match! :-)");
        } else {
            System.out.println("Strings do not match! :-(");
        }

    }

}
```

```java
/*
 * Code10.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * String input and comparison - fixed!
 *
 */

// allow us use of the keyboard scanner. More information from the docs:
// http://java.sun.com/javase/6/docs/api/java/util/Scanner.html
import java.util.Scanner;

class Code10 {

    public static void main(String [] args) {

        String str = "Hello, world";
        String input = null;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        System.out.print("Please type a string: ");

        // wait for the user to enter an integer
        try {
            input = keyboard.nextLine();
        } catch(Exception e) {
            System.out.println("Invalid input! Quitting..");
            System.exit(1);
        }

        // test to see if what the user entered matches our string, using
        // the equals method.
        if(str.equals(input)) {
            System.out.println("Strings match! :-)");
        } else {
            System.out.println("Strings do not match! :-(");
        }

    }

}
```

```java
/*
 * Code11.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Introduction to arrays.
 *
 */

class Code11 {

    public static void main(String [] args) {

        // declare the array
        int[] grades;

        // allocate memory for 5 indices
        grades = new int[5];

        // assign some values to the array
        grades[0] = 100;
        grades[1] = 76;
        grades[2] = 92;
        grades[3] = 95;
        grades[4] = 14;

        // print out each value
        System.out.println(grades[0]);
        System.out.println(grades[1]);
        System.out.println(grades[2]);
        System.out.println(grades[3]);
        System.out.println(grades[4]);

    }

}
```

```java
/*
 * Code12.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A less stupid way of printing an array's contents.
 *
 */

class Code12 {

    public static void main(String [] args) {

        // declare the array
        int[] grades;

        // allocate memory for 5 indices
        grades = new int[5];

        // assign some values to the array
        grades[0] = 100;
        grades[1] = 76;
        grades[2] = 92;
        grades[3] = 95;
        grades[4] = 14;

        for(int i = 0; i < grades.length; i++) {
            System.out.println("Grade "+(i+1)+": "+grades[i]);
        }

    }

}
```

```java
/*
 * Code13.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A less stupid way of printing an array's contents, and an optimization.
 *
 */

class Code13 {

    public static void main(String [] args) {

        // declare the array
        int[] grades;

        // allocate memory for 5 indices
        grades = new int[5];

        // assign some values to the array
        grades[0] = 100;
        grades[1] = 76;
        grades[2] = 92;
        grades[3] = 95;
        grades[4] = 14;

        // pre-store the length into a variable
        for(int i = 0, j = grades.length; i < j; i++) {
            System.out.println("Grade "+(i+1)+": "+grades[i]);
        }

    }

}
```

```java
/*
 * Code14.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Printing an array's contents with a while loop.
 *
 */

class Code14 {

    public static void main(String [] args) {

        // declare the array
        int[] grades;

        // allocate memory for 5 indices
        grades = new int[5];

        // assign some values to the array
        grades[0] = 100;
        grades[1] = 76;
        grades[2] = 92;
        grades[3] = 95;
        grades[4] = 14;

        int i = 0, j = grades.length;

        // loop the code while the condition evaluates to true.
        while(i < j) {

            System.out.println("Grade "+(i+1)+": "+grades[i]);
            i++;

        }

    }

}
```

```java
/*
 * Code15.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Printing an array's contents with a do-while loop.
 *
 */

class Code15 {

    public static void main(String [] args) {

        // declare the array
        int[] grades;

        // allocate memory for 5 indices
        grades = new int[5];

        // assign some values to the array
        grades[0] = 100;
        grades[1] = 76;
        grades[2] = 92;
        grades[3] = 95;
        grades[4] = 14;

        int i = 0, j = grades.length;

        // evaluate a condition after an initial run. In other words,
        // this loop is guaranteed to run at least once.
        do {
            System.out.println("Grade "+(i+1)+": "+grades[i]);
        } while(++i < j);

    }

}
```

```java
/*
 * Code16.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Using a do-while loop to guarantee valid input from the user.
 *
 */

import java.util.Scanner;

class Code16 {

    public static void main(String [] args) {

        boolean invalid;
        int input = 0;

        // instantiate the Scanner class, accessing data from the keyboard
        Scanner keyboard = new Scanner(System.in);

        do {

            invalid = false;
            System.out.print("Please enter an integer: ");

            // try inputting an integer, if a user doesn't ..
            try {

                input = keyboard.nextInt();

            } catch(Exception e) {
                // an exception will be thrown, and we can catch it to alert
                // the user that something bad happened.

                System.out.println("Invalid input! Please try again..");
                invalid = true;
                keyboard.next();
            }

        } while (invalid);

        System.out.println("You have finally entered a valid integer: "+input);

    }

}
```

```java
/*
 * Code17.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A brief introduction to methods.
 *
 */

class Code17 {

    // declare a field
    private int num;

    public static void main(String [] args) {
        Code17 myObj = new Code17();
        System.out.println("Val: " + myObj.get());
        myObj.set(2);
        System.out.println("Val: " + myObj.get());
    }

    // declare a Constructor for the class and initialize our fields.
    public Code17() {
        num = 0;
    }

    // define a public 'set' function to modify state
    public void set(int val) {
        num = val;
    }

    // define a public 'get' function to get returned the current state
    public int get() {
        return num;
    }

}
```

```java
/*
 * Code18.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A brief introduction to methods.
 *
 */

class Code18 {

    private static void printSuccess(Car myCar) {
        System.out.println("Wow, your "+myCar.getYear()+" "+myCar.getMake()+
                           " "+myCar.getModel()+" is really booking it at "+
                           myCar.getSpeed()+" mph!");

    }

    private static void printFailure(Car myCar) {
        System.out.println("Too bad, your "+myCar.getYear()+" "+myCar.getMake()+
                           " "+myCar.getModel()+" can't go that fast.");
    }

    public static void main(String [] args) {
        Car tortoise = new Car("Toyota", "Camry", 2009);
        Car hare = new Car("Ferrari", "F430", 2009);

        tortoise.setMaxSpeed(100.0);
        hare.setMaxSpeed(200.0);

        if(hare.setSpeed(155.0))
            printSuccess(hare);
        else
            printFailure(hare);

        if(tortoise.setSpeed(135.0))
            printSuccess(tortoise);
        else
            printFailure(tortoise);
    }


}
```

```java
/*
 * Car.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * A brief introduction to methods.
 *
 */

class Car {

    private String make;
    private String model;
    private int year;
    private double speed;
    private double maxSpeed;

    public Car(String mk, String mdl, int yr) {
        make = mk;
        model = mdl;
        year = yr;
    }

    public void setMaxSpeed(double max) {
        maxSpeed = max;
    }

    public boolean setSpeed(double spd) {
        if(spd > maxSpeed) return false;

        speed = spd;
        return true;
    }

    public double getSpeed() {
        return speed;
    }

    public String getMake() {
        return make;
    }

    public String getModel() {
        return model;
    }

    public int getYear() {
        return year;
    }

}
```

```java
/*
 * Code19.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Demonstrates implications of passing data into methods by value.
 *
 */

class Code19 {

    // define two integer fields
    private static int firstNum;
    private static int secondNum;

    // an object that contains two pieces of data: an int x and an int y
    // The constructor for this class accepts two values that are then
    // inserted into these fields.
    private static class Point {
        public int x;
        public int y;

        public Point(int first, int second) {
            x = first;
            y = second;
        }
    }

    // Accepts two integers as parameters and should swap them.
    private static void swap(int x, int y) {
        int temp;

        temp = x;
        x = y;
        y = temp;
    }

    // Accepts a Point object and swaps the data in the x and y fields.
    private static void swap(Point a) {
        int temp;

        temp = a.x;
        a.x = a.y;
        a.y = temp;
    }

    public static void main(String [] args) {

        // assign some values to our integer fields
        firstNum = 1;
        secondNum = 2;

        // instantiate a new Point object, and provide some data
        Point a = new Point(3, 4);

        // attempt to swap the data in the two integer fields
        System.out.println("firstNum: "+firstNum+", secondNum: "+secondNum);
        swap(firstNum, secondNum);
        System.out.println("firstNum: "+firstNum+", secondNum: "+secondNum);

        // attempt to swap the data in the Point object
        System.out.println("a.x: "+a.x+", a.y: "+a.y);
        swap(a.x, a.y);
        System.out.println("a.x: "+a.x+", a.y: "+a.y);

    }

}
```

```java
/*
 * Code20.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Demonstrates class inheritance.
 *
 */

class Code20 {

    private static Computer macPro;
    private static Laptop macBookAir;
    private static Server xserve;

    private static void showRunningMachines() {
        System.out.println();

        System.out.println("The following machines are turned on:");

        if(macPro.isOn()) System.out.println("- "+macPro.model);
        if(macBookAir.isOn()) System.out.print("- "+macBookAir.model);
        if(xserve.isOn()) System.out.println("- "+xserve.model);

        System.out.println();

    }

    public static void main(String [] args) {

        macPro = new Laptop("Apple", "MacPro", 2009, 2.8);
        macBookAir = new Laptop("Apple", "MacBook Air", 2009, 1.8);
        xserve = new Server("Apple", "Xserve", 2010, 3.2);

        macPro.turnOn();
        macBookAir.turnOn();

        macBookAir.setBattery(0.5);
        xserve.setRackHeight(1);

        System.out.println(macPro.built + " " + macPro.make + " " + macPro.model
                            + " runs at " + macPro.speed + "ghz.");

        System.out.println(macBookAir.built + " " + macBookAir.make + " " +
                           macBookAir.model + " runs at " + macBookAir.speed +
                           "ghz and has a battery level of "+
                           (macBookAir.getBattery() * 100) + "%.");

        System.out.println(xserve.built + " " + xserve.make + " " + xserve.model +
                           " runs at " + xserve.speed + "ghz and is " +
                           xserve.getRackHeight() + "U tall.");

        showRunningMachines();

        System.out.println("\nUh oh, MacBook Air's battery is dying..");
        macBookAir.setBattery(0.0);

        showRunningMachines();

    }

}
```

```java
/*
 * Computer.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Demonstrates class inheritance; a parent class for 'computer' sub-classes.
 *
 */

class Computer {

    public String make;
    public String model;
    public int built;
    public double speed;
    private boolean on;

    public Computer(String mk, String mdl, int yr, double spd) {
        make = mk;
        model = mdl;
        built = yr;
        speed = spd;
        on = false;
    }

    public void turnOn() {
        on = true;
    }

    public void turnOff() {
        on = false;
    }

    public boolean isOn() {
        return on;
    }

}
```

```java
/*
 * Laptop.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Demonstrates class inheritance; Laptop is a subclass of Computer.
 *
 */

class Laptop extends Computer {

    private double battery;

    public Laptop(String mk, String mdl, int yr, double spd) {
        super(mk, mdl, yr, spd);
    }

    public void setBattery(double lvl) {
        battery = lvl;

        if (battery < 0.01) super.turnOff();
    }

    public double getBattery() {
        return battery;
    }

}
```

```java
/*
 * Server.java
 * Dan Armendariz
 * Computer Science S-76
 *
 * Demonstrates class inheritance; Server is a subclass of Computer.
 *
 */

class Server extends Computer {

    private int rackHeight;

    public Server(String mk, String mdl, int yr, double spd) {
        super(mk, mdl, yr, spd);
    }

    public void setRackHeight(int height) {
        rackHeight = height;
    }

    public int getRackHeight() {
        return rackHeight;
    }

}
```