TRUE/FALSE

1. If `x` is a `String`, then `x = new String("OH");` and `x = "OH";` will accomplish the same thing.

   ANS: T
   In Java, to instantiate (assign a value to) an object, you must use `new` and the class's constructor. However, since `Strings` are so common in Java, they can be instantiated in a way similar to assigning primitive types their values. So, both of the above assignment statements will accomplish the same task.

2. If `x` is the `String` `"Hi There"`, then `x.toUpperCase().toLowerCase();` will return the original value of `x`.

   ANS: F
   `x.toUpperCase()` returns `x` as all capital letters, while `x.toLowerCase()` will return `x` as all lower case letters. So, this code will first convert `x` to all upper case letters and then convert the new version to all lower case characters.

3. If `String name = "George W. Bush";` then the instruction `name.length();` will return `14`.

   ANS: T
   There are 14 characters between the quote marks including two blanks and a period.

4. If `String a = "ABCD"` and `String b = "abcd"` then `a.equals(b);` returns `false` but `a.equalsIgnoreCase(b);` returns `true`.

   ANS: T
   Since `"ABCD"` is not the same as `"abcd"`, the `equals` method returns `false`, but by ignoring case in `equalsIgnoreCase`, the two are considered to be the same.

5. Unlike the `String` class where you must pass a message to an object (instance) of the class, as in `x.length()`, in order to use either the `Scanner` or `Math` classes, you pass messages directly to the class name, as in `Math.abs()` or `scan.nextInt()`.

   ANS: T
   The `Math` and `Scanner` classes use methods known as static methods (or class methods) which are invoked by passing a message directly to the class name itself rather than to an object of the class.

6. In order to generate a random number, you must use `Math.random()`.

ANS: F
There is also a `Random` class available in `java.util`. This class can generate random `int`, `float`, and `double` values. This is a better mechanism for generating random numbers because you can instantiate several different random number generators.

7. A `double` is wider than a `float` and a `float` is wider than an `int`.

ANS: T
Wider types are larger in size or can store a greater range of values. The `double` is 64 bits whereas the `float` is 32 bits and the `float`, because of the way it is stored, can store a significantly larger range of values than the `int`.

8. A variable of type `boolean` will store either a `0` or a `1`.

ANS: F
A `boolean` variable can store only one of two values, but these values are the reserved words `true` and `false`. In C, C++, and C# `boolean`s are implemented as `int` variables that store only a `0` or a `1`, but in Java, the authors of the language opted to use the `boolean` literals `true` and `false` as this is considered to be semantically more understandable (and is much safer).

9. In Java, `'a'` and `'A'` are considered to be different values.

ANS: T
Characters values are stored using ASCII, and `'a'` and `'A'` have different ASCII values as well as different Unicode values.

10. You cannot cast a `String` to be a `char` and you cannot cast a `String` which stores a number to be an `int`, `float`, or `double`.

ANS: T
There is no mechanism available to cast a `String` to one of the primitive types, but there are methods available to perform a similar action and return a character at a given location (`CharAt`) or to return the `int`, `float`, or `double` value equivalent to the number stored in the `String`.

11. The values of `(double)5/2` and `(double)(5/2)` are identical.

ANS: F
In the first expression, the `(double)` cast applies to the`int 5`, changing it to the `double` value, `5.0`. Then `5.0/2` is calculated, yielding the `double` value, `2.5`. In the second expression the `int` division is performed first, yielding the value `2`. `2` is then changed to a `double`, yielding the `double` value, `2.0`.

12. There are three ways that data conversion may occur: by assignment, by promotion, and by casting.

ANS: T
Assignment conversion occurs when a value on the right side of the assignment operator is converted prior to being stored in the variable on the left. Promotion occurs within an expression when values of differing widths are combined. Casting is a programmer's explicit way to control the data conversion process.

13. As explained in the Software Failure section in the text, the Mars Lander most likely crash landed when its descent engines cut off too high over the Mars surface.

ANS: T
The software on board the lander mistook the vibrations caused by the deployment of the lander's legs for the vibration caused by actually landing on the planet's surface.

MULTIPLE CHOICE

Example Code Ch 02-1

```
public class Questions1_4
   {
         public static void main(String[] args)
         {
                System.out.print("Here");
                System.out.println("There  " + "Everywhere");
                System.out.println("But  not" + "in  Texas");
         }
   }
```

1. **Refer to the class definition in Example Code Ch 02-1.** The program will print the word `"Here"` and then print

   a. `"There Everywhere"` on the line after `"Here"`
   b. `"The"` on the line after `"Here"` and `"Everywhere"` on the line after `"There"`
   c. `"There Everywhere"` on the same line as `"Here"`
   d. `"ThereEverywhere"` on the same line as `"Here"`
   e. `"ThereEverywhere"` on the line after `"Here"`

   ANS: C
   `System.out.print` will output the word `Here` but will leave the cursor at that point rather than starting a new line. The next statement will output `There Everywhere` immediately after the word `Here`. Since there is a blank space within the quote marks for `There`, there is a blank space inserted between `There` and `Everywhere`.

2. **Refer to the class definition in Example Code Ch 02-1.** The final `println` command will output

   a. `"But not in Texas"`
   b. `"But notin Texas"`
   c. `"But not"` on one line and `"in Texas"` on the next line
   d. `"But not+in Texas"`
   e. `"But not + in Texas"`

   ANS: B

The + performs `String` concatenation, so that `But not` and `in Texas` are concatenated together. Notice that there is no blank space after `not` or before `in` so that when they are concatenated, they are placed together without a blank space.

3. **Refer to the class definition in Example Code Ch 02-1.** How many lines of output are provided by this program?

   a. 1
   b. 2
   c. 3
   d. 4
   e. 5

   ANS: B
   There will be one line of output for the first two statements combined because the `print` statement does not return the cursor to start a new line. And since the second statement is a `println`, it returns the cursor and the last `println` outputs its message on a separate line.

4. **Refer to the class definition in Example Code Ch 02-1.** A reasonable documentation comment for this program might be

   a. `//a program that demonstrates the differences between print, println, and how + works`
   b. `//a program that outputs a message about Texas`
   c. `//a program that demonstrates nothing at all`
   d. `//a program that outputs the message "Here There Everywhere But not in Texas`
   e. `//a program that contains three output statements`

   ANS: A
   Remember that comments should not state the obvious (ruling out D and E) but instead should explain what the program is doing or why. This program demonstrates `print` and `println` and +.

5. The following statement will output _____ lines of text.

```
System.out.println("1 big bad wolf\t8 the 3 little pigs\n4
                                dinner\r2night");
```

   a. 1
   b. 2
   c. 3
   d. 4
   e. 5

   ANS: B
   The `\t` escape sequence inserts a tab, but leaves the cursor on the same line. The `\n` escape sequence causes a new line to be produced so that `4 dinner` is output on the next line. The escape sequence `\r` causes the carriage to return (that is, the cursor to be moved back to the left margin) but because it does not start a new line, `2night` is output over `4 dinn` resulting in a second line that looks like `2nighter`.

6. If you want to output the text `"hi there"` including the quote marks, which of the following would you use?

a. `System.out.println("hi there");`
b. `System.out.println(""hi there"");`
c. `System.out.println("\"hi there");`
d. `System.out.println("\"hi there\"");`
e. None of these; it is not possible to output a quote mark because it is used to mark the beginning and end of the `String` to be displayed.

ANS: D
`\"` is an escape sequence used to place a quote mark in a `String`, so it is used here to output the quote marks with the rest of the `String`.

7. The word `println` is a(n)

a. method
b. reserved word
c. variable
d. class
e. `String`

ANS: A
The word `println` is passed as a message to the `System.out` object, and so `println` is a method.

8. A Java variable is the name of a

a. numeric data value stored in memory
b. data value stored in memory that cannot change during the program's execution
c. data value stored in memory that can change its value but cannot change its type during the program's execution
d. data value stored in memory that can change both its value and its type during the program's execution
e. data value or a class stored in memory that can change both its value and its type during the program's execution

ANS: C
A variable can change its value as long as it is within the same type, but the variable cannot change type. A constant is similar to a variable but it cannot change its value. Variables can be numeric but are not restricted to being numeric, they can also be `boolean`, `char`, or an object of any class.

9. Of the following types, which one cannot store a numeric value?

a. `int`
b. `double`
c. `float`
d. `char`
e. All of these can store numeric values

ANS: D
`int` and `byte` are used to store whole numbers (integers) and `float` is used to store a real or floating point value (value with a decimal point). A `char` stores a single character including letters, punctuation marks and digits. However, storing the numeric digit `'5'` is not the same as storing the number `5`.

10. What is the value of `z` after the following assignment statement is executed?

```
float z = 5/10;
```

a. `0.0`
b. `0.5`
c. `5.0`
d. `0.05`
e. None of these; a run-time error will occur because `z` is a `float` and `5/10` is an `int`.

ANS: A
`5` and `10` are both `int` values, so `5 / 10` is an integer division. The result is `0`. Even though `z` is a `float` and can store the real answer, `0.5`, it only gets `0` because of the integer division. In order to get `0.5`, we would have to first cast `5` or `10` as a `float`.

11. Which of the following situations would require a cast?

a. using `charAt` to take an element of a `String` and store it in a `char`
b. storing an `int` in a `float`
c. storing a `float` in a `double`
d. storing a `float` in an `int`
e. All of these require casts

ANS: D
For A, `charAt` returns a `char`, so there is no problem. In B and C, the situations are widening operations taking a narrower type and storing the value in a wider type. Only in D is there a situation where a wider type is being stored in a narrower type, so a cast is required.

12. If `x` is an `int` and `y` is a `float`, which of the following statements is not a legal assignment statement?

a. `y = x;`
b. `x = y;`
c. `y = (float)x;`
d. `x = (int)y;`
e. All of these are legal assignment statements

ANS: B
Since `x` is an `int`, it cannot except a `float` unless the `float` is cast as an `int`. There is no explicit cast in the assignment statement in B. In A, a cast is not necessary because a `float` (`y`) can accept an `int` value (`x`), and in C and D, explicit casts are present making them legal.

13. What will be the result of the following assignment statement, given that `b = 5` and `c = 10`?

```
int a = b * (-c + 2)/2;
```

a. `30`
b. `-30`
c. `20`
d. `-20`
e. `-6`

ANS: D
The unary minus is applied first giving `-c + 2 = -8`. Next, the `*` is performed giving `5*-8=-40`, and finally the `/` is performed giving `-40/2 = -20`.

14. Which of the following is true regarding the mod operator, `%`?

a. It can only be performed on `int` values and its result is a `double`.
b. It can only be performed on `int` values and its result is an `int`.
c. It can only be performed on `float` or `double`.values and its result is an `int`.
d. It can only be performed on `float` or `double`.values and its result is a `double`.
e. It can be performed on any numeric values and its result is always numeric.

ANS: E
Mod, or modulo, returns the remainder that results from a division. The remainder is always is numeric. Although usually integer values are used, the `%` operator may be used on all kinds of numeric data.

15. Assume that `x`, `y`, and `z` are all `ints` equal to `50`, `20`, and `6` respectively. What is the result of:

```
x / y / z
```

a. `0`
b. `12`
c. `16`
d. this would cause a syntax error
e. this would cause a run-time error because it is a division by zero

ANS: A
This division is performed left to right, so first `50/20` is performed. Since `50` and `20` are `ints`, this results in `2`. Next, `2/6` is performed which is `0`. Notice that if the division were performed right to left, the evaluation would instead be `50/(20/6) = 50/3 = 16`.

16. Assume that `x` and `y` are `ints` equal to `10` and `5` respectively. What is the output of the following statement?

```
System.out.println(x + y);
```

a. `15`
b. `105`
c. `10 5`
d. `x + y`
e. this would cause an error since neither `x` nor `y` is a `String`

ANS: A
Java first computes `x+y` and then casts it as a `String` to be output. `x+y= 10 + 5 = 15`, so the statement outputs `15`.

17. Assume that `x` and `y` are `ints` equal to `10` and `5` respectively. What is the output of the following statement?

```
System.out.println("" + x + y);
```

a. `15`
b. `105`
c. `10 5`
d. `x + y`
e. this would cause an error since neither `x` nor `y` is a `String`

ANS: B
The `""` causes the rest of the expression to be treated as a `String`, and so the two + signs are used as `String` concatenation. Therefore `x+y` becomes `x` concatenated with `y`, or `105`.

18. If you want to store the value `"Harry Potter"` in the `String` variable `name`, which of the following statements could you use?

    a. `String name = "Harry Potter";`
    b. `String name = new String("Harry Potter");`
    c. `String name = "Harry" + " " + "Potter";`
    d. `String name = new String("Harry" + " " + "Potter");`
    e. Any of these would work

    ANS: E
    There are two ways to store a character string into a `String` variable, by constructing a new `String` using `new String(string value);` or by using an assignment statement, so either A or B will work. In C and D, we have variations where the `String` concatenation operator is used. So all four approaches will work.

19. Given three `String` variables, `a`, `b`, and `c`, which of the following statements could you use to achieve the same thing as:

    ```
    c = a + b;
    ```

    a. `c = a.length() + b.length();`
    b. `c = (int)a + (int)b;`
    c. `c = a.concat(b);`
    d. `c = b.concat(a);`
    e. `c = a.plus(b);`

    ANS: C
    The statement `c = a + b` uses the concatenation operator (not to be confused with numeric addition). The same result can be achieved by passing `a` the concat message with `b` as the parameter. Answer D will set `c` to be `b + a` rather than `a + b`.

20. If the `String` `major = "Computer Science"`, what is returned by `major.charAt(1)`?

    a. `'C'`
    b. `'o'`
    c. `'m'`
    d. `"C"`
    e. `"Computer"`

    ANS: B
    Neither D nor E would be correct because `charAt` returns a `char` (single character) whereas these answers are `Strings`. So, the question is, which character is returned? In Java, the first character of a `String` is numbered `0`. So `charAt(1)` returns the second character of the `String`, or `'o'`.

21. Which of the following would return the last character of the `String` `x`?

    a. `x.charAt(0);`
    b. `x.charAt(last);`
    c. `x.charAt(length(x));`

d. `x.charAt(x.length()-1);`
e. `x.charAt(x.length());`

ANS: D
Since `last` is not defined, B is syntactically invalid. The 0th character is the first in the `String`, so A is true only if the `String` has a single character. The answer in C is syntactically invalid as `length` can only be called by passing the message to `x`. Finally, D and E are syntactically valid, but since `length` returns the size of the `String`, and since the first character starts at the 0th position, the last character is at `x.length()-1`, so E would result in a run-time error.

22. Given `String name = "Arleen Crabtree"`. What will the following instruction return?

```
name.toUpperCase().replace('R', 'Z');
```

a. `"ARLEEN CRABTREE"`
b. `"AZLEEN CZABTREE"`
c. `"Azleen Czabtree"`
d. `"Arleen Crabtree"`
e. `"ArZleen CrZabtree"`

ANS: B
The `toUpperCase` method returns the `String` as all upper case characters, or `"ARLEEN CRABTREE"`. The `replace` method will replace each instance of `R` with `Z`.

23. Which library package would you import to use `NumberFormat` and `DecimalFormat`?

a. `java.beans`
b. `java.io`
c. `java.lang`
d. `java.text`
e. `java.util`

ANS: D
Both of these classes are used for "text processing", that is, to handle values like `Strings`. Such classes are found in `java.text`. You might think these would be in `java.io`, but this library has classes related to input and sending output to locations other than the monitor.

24. Which library package would you import to use the class `Random`?

a. `java.beans`
b. `java.io`
c. `java.lang`
d. `java.text`
e. `java.util`

ANS: E
This is a Java numeric utility, and so is found in the `java.util` package.

25. The `Random` class has a method, `nextFloat()` which returns a random `float` value between

a. `-1` and `+1`
b. `0` and `1`
c. `0` and `99`
d. `1` and `100`

e. `-2,147,483,648` and +2,147,483,647

ANS: B
The method `nextFloat()` returns a `float` value between 0 and 1 so that it may be used as a probability.

26. If you want to output a `double` so that at least one digit appears to the left side of the decimal point and exactly one digit appears to the right side, which pattern would you give a `DecimalFormat` variable when you instantiate it?

a. `"0.0"`
b. `"0.#"`
c. `"0.0#"`
d. `"0.##"`
e. `"#.#"`

ANS: A
The pattern `"0.0"` says to output all of the digits to the left of the decimal point or a `0` if there are none (you want at least 1 digit to the left, including a 0 if there are no digits) and exactly one digit to the right of the decimal point, even if the digit is 0. The patterns `"0.#"` and `"#.#"` would not output any digits to the right side of the decimal point if the value had no decimal portion (e.g., 39.0). The patterns in C and D can output up to 2 values to the right of the decimal point.

27. Given the `double likelihood = 0.013885` and given

```
DecimalFormat dformatter = DecimalFormat("0.00##");
```

What would be the output if you execute the following:

```
System.out.println(df.format(likelihood));
```

a. `0.013885`
b. `0.0139`
c. `0.0145`
d. `.0138`
e. `.014`

ANS: B
The format `"0.00##"` means that at least 1 digit should appear to the left of the decimal point, even if it is 0, that at least two digits should appear to the right of the decimal point even if they are zeros, and if there are two additional digits to the right, they should also be printed, rounding the value off as needed. So, this gives `0.0139` since the number, `0.013885` will be rounded up to `0.0139`.

28. Using `getCurrencyInstance()` formats a variable, automatically inserting

a. a decimal point for cents
b. a dollar sign
c. a percent sign
d. all three of these
e. a decimal point for cents and a dollar sign but not a percent sign

ANS: E
`getCurrencyInstance` will format a `double` or `float` variable so that it has 2 digits to the right of the decimal point and a dollar sign preceding the value. `getPercentInstance` is used to format a `double` or `float` to be output with a percent sign.

29. What will be the value of `z` after the following statement is executed?

```
int z = 50 /10.00;
```

a. `5`
b. `5.0`
c. `50`
d. `10`
e. None of thse; a run-time error will occur because `z` is an `int` and `50/10.00` is not

ANS: E
Because `10.00` is not an `int`, the division produces a double precision value which cannot be stored in the `int z`. For this to work, the result of the division must be cast as an `int` before being stored in `z`, or the value `10.00` would have to first be cast as an `int` before the division takes place.

30. Since you cannot take the square root of a negative number, which of the following could you use to find the square root of the variable `x`?

a. `Math.sqrt(x*x);`
b. `Math.sqrt((int)x);`
c. `Math.sqrt(Math.abs(x));`
d. `Math.abs(Math.sqrt(x));`
e. `Math.sqrt(-x);`

ANS: C
`Math.abs` returns the absolute value of `x`. If `x` is negative, `Math.sqrt(x)` causes a run-time error, but `Math.sqrt(Math.abs(x))` does not since `x` is first converted to its positive equivalent before the square root is performed. Answer A returns `x` (square root of x^2 is x). In answer B, casting `x` to an `int` will not resolve the problem if `x` is negative. In answer D, the two `Math` functions are performed in opposite order and so if `x` is negative, it still generates a run-time error. Answer E only will work if `x` is not positive and so if `x` is positive, it now generates a run-time error.

31. Given `x` is a `double` and has the value `0.362491`. To output this value as `36%`, you could use the `NumberFormat` class with:

```
NumberFormat nf = NumberFormat.getPercentInstance();
```

Which of the following statements then would output `x` as 36%?

a. `System.out.println(x);`
b. `System.out.println(nf);`
c. `System.out.println(nf.format(x));`
d. `System.out.println(nf.x);`
e. `System.out.println(format(x));`

ANS: C
`nf` is an object and so must be passed a message to use it. The method to format a `float` or `double` is called `format` and the value to be formatted is the parameter passed to `format`. Therefore, the proper way to do this is `nf.format(x)`. The answer in A merely outputs `0.362491` while the answers to B, C, and E are syntactically invalid.

Example Code Ch 02-2

```
import java.util.Scanner;
```

```
public class Questions33_34
{
   public static void main(String[] args)

   {
     int x, y, z;
     double average;
     Scanner scan = new Scanner(System.in);
     System.out.println("Enter an integer value");
     x = scan.nextInt();
     System.out.println("Enter another integer value");
     y = scan.nextInt();
     System.out.println("Enter a third integer value");
     z = scan.nextInt();
     average = (x + y + z) / 3;
     System.out.println("The result of my calculation
                          is " + average);
   }
}
```

32. **Refer to Example Code Ch 02-2.** This code computes

    a. The correct average of `x`, `y`, and `z` as a `double`
    b. The correct average of `x`, `y`, and `z` as an `int`
    c. The average of `x`, `y`, and `z` as a `double` but the result may not be accurate
    d. The sum of `x`, `y`, and `z` as an `int`
    e. The remainder of the sum of `x`, `y`, and `z` divided by `3`

    ANS: C
    Because the division is an `int` division, even though the result is stored in a `double`, the resulting `double` may not be accurate. For instance, if `x`, `y` and `z` are `1`, `2`, and `4`, the `double` average should be `2.33333` but average will instead be `2.00000`.

33. **Refer to Example Code Ch 02-2.** What is the output if `x = 0`, `y = 1`, and `z = 1`?

    a. `0`
    b. `0.0`
    c. `0.6666666666666666`
    d. `0.6666666666666667`
    e. `0.67`

    ANS: B
    The division is performed as an `int` division since `x`, `y`, `z`, and `3` are all `ints`. Therefore, `average` gets the value `0.0`. It is output as `0.0` instead of `0` because `average` is a `double`, which outputs at least one decimal digit unless specified otherwise using the `DecimalFormat` class.

34. In order to create a constant, which of the following Java reserved words is used?

    a. `private`
    b. `static`
    c. `int`
    d. `final`
    e. `class`

    ANS: D

The reserved word `final` indicates that this is the final value that will be stored in this variable, thus making it unchangeable, or constant. While constants can be of type `int`, constants can be of any other type as well. It is the `final` reserved word that makes the value unchangeable.

35. Given three `int` variables with the values `a = 5`, `b = 7`, and `c = 12`, what is the value of `z` after the following statement is executed?

```
int z = (a * b - c) / a;
```

a. `0`
b. `4`
c. `5`
d. `-5`
e. `23`

ANS: B
`(a * b - c) / a = (5 * 7 - 12) / 5 = (35 - 12) / 5 = 23 / 5`, and since `23` and `5` are `int` values, the division is performed as an `int` division, or `23 / 5 = 4`.

36. Java is a strongly typed language. What is meant by "strongly typed"?

a. Every variable must have an associated type before it can be used.
b. Variables can be used without declaring their types.
c. Every variable has a single type associated with it throughout its existence in the program and the variable can only store values of that type.
d. Variables are allowed to change type during their existence in a program so long as the value a variable currently stores is of the type it currently declared to be.
e. Variables are allowed to change types during their existence in a program but only if the change is to a narrower type.

ANS: C
Strong typing is a property of a programming language whereby the variable's type does not change during the variable's existence, and any value stored in that variable is of that type. The reason that strong typing is important is it guarantees that a program that was successfully compiled will not have run-time errors associated with the misuse of types for the variables declared.

37. As presented in the Software Failure section of the text, the root cause of the Mars Climate Orbiter problem was

a. the cost of the project
b. an inability to track the orbiter at long distances
c. atmospheric friction
d. a communication issue between subsystems
e. None of these

ANS: D
The cause was determined to be an embarrassing communication issue between subsystems. The navigation subsystem used imperial units of measure (pound-force) and the spacecraft's software itself expected data in metric units (newtons).

PROBLEM

1. Write a set of instructions to prompt the user for an `int` value and input it using the `Scanner` class into the variable `x` and prompt the user for a `float` value and input it using the `Scanner` class into the variable `y`.

   ANS:
   ```
   Scanner scan = Scanner.create(System.in);
   System.out.println("Enter an integer");
   int x = scan.nextInt();
   System.out.println("Enter a float");
   float y = scan.nextFloat();
   ```

2. Provide an example of how you might use a `boolean`, a `float`, a `char`, a `String`, and an `int`.

   ANS:
   Answers may vary. Samples:

   `boolean` to store whether a person is of voting age or not
   `float`: to store someone's GPA
   `char`: to store someone's middle initial
   `String`: to store someone's social security number
   `int`: to store someone's age

3. Explain what the following statement computes:

   ```
   int z = (int)Math.ceil(Math.sqrt(x)/Math.sqrt(y));
   ```

   ANS:
   The integer value which bounds `sqrt(x)/sqrt(y)` where bounds means it is equal to the result of the division or the next `int` larger than the result of the division if the result has a remainder.

4. Given four int values, `x1`, `x2`, `y1`, `y2`, write the code to compute the distance between the two points `(x1, y1)` and `(x2, y2)`, storing the result in the `double distance`.

   ANS:
   ```
   double distance = Math.sqrt(Math.pow(x1 - y1, 2) + Math.pow(x2 -
   y2,2));
   ```

5. Write an assignment statement to compute the gas mileage of a car where the `int` values `milesTraveled` and `gallonsNeeded` have already been input. The variable `gasMileage` needs to be declared and should be a `double`.

   ANS:
   ```
   double gasMileage = (double)milesTraveled/gallonsNeeded;
   ```
   The reason that the value `milesTraveled` is cast as a `double` is to ensure that the division is performed as a `double` and not an `int`.

6. Write the `paint` method for an applet so that it contains 4 concentric circles (each circle is inside the previous circle), where the largest circle is bounded by a 400x400 box and the smallest is bounded by a 100x100 box. Each circle is centered on an applet that is 400x400. Make each circle a different color of your choice.

   ANS:
   ```
   public void paint(Graphics page)
   {
       page.setColor(Color.white);
       page.fillOval(0, 0, 400, 400);
       page.setColor(Color.yellow);
       page.fillOval(50, 50, 300, 300);
       page.setColor(Color.orange);
       page.fillOval(100, 100, 200, 200);
       page.setColor(Color.red);
       page.fillOval(150, 150, 100, 100);
   }
   ```

7. Given two points in an applet represented by the four `int` variables `x1`, `y1`, `x2` and `y2`, write a `paint` method to draw a line between the two points and write the location of the two points next to the two points.

   ANS:
   ```
   public void paint(Graphics page)
   {
       page.setColor(Color.blue);
       page.drawLine(x1, y1, x2, y2);
       page.drawString("" + x1 + ", " + y1, x1, y1);
       page.drawString("" + x2 + ", " + y2, x2, y2);
   }
   ```

8. An employer has decided to award a weekly pay raise to all employees by taking the square root of the difference between his weight and the employee's weight. For instance, an employee who weighs 16 pounds less than the employer will get a $4 per week raise. The raise should be in whole dollars (an `int`). Assume that `employerWeight` and `employeeWeight` are both `int` variables that have been input. Write an assignment statement to compute the `int` value for raise.

   ANS:
   ```
   raise = (int) Math.sqrt(Math.abs(employerWeight - employeeWeight));
   ```

   The absolute value of the difference must be taken in case the employee weighs more than the employer, which would then result in an attempt to take the square root of a negative number. So `Math.abs` is used. After computing the absolute value of the difference, the square root is taken and the result (a `double`) is cast as an `int` to be stored in `raise`.

9. Using the various `String` methods, manipulate a `String` called `current` to be the last character of `current` followed by the remainder of its characters in order, placing the result in a `String` called `rearranged`.

   ANS:

```
public class Compute14
{
    public static void main(String[] args)
    {
        Scanner scan = Scanner.create(System.in);
        System.out.println("Enter an integer value");
        int x = scan.nextInt();
        int twoToTheX = (int) Math.pow(2, x);
        int xToThe10th = (int) Math.pow(x, 10);
        System.out.println("The results are " + twoToTheX +
                           " and " + xToThe10th);
    }
}
```

14. What is wrong with the following assignment statement? Assume `x` and `y` are both `String` objects.

```
String z = x.equals(y);
```

ANS:
The equals method returns a `boolean` value, not a `String`. The values `true` and `false` are not the same as the values `"true"` and `"false`.

15. Write a program that will input some number of cents (less than 100) and output the number of quarters, dimes, nickels and pennies needed to add up to that amount.

ANS:

```
import java.util.Scanner;
public class Change
{
   public static void main(String[] args)
   {
     Scanner scan = Scanner.create(System.in);
     System.out.println("Enter the amount of change");
     int amount = scan.nextInt();
     System.out.println("The change for " + amount " cents
                                    is: ");
     int quarters = amount / 25;
     System.out.println("      " + quarters + " quarters");
     amount = amount - quarters * 25;
     int dimes = amount / 10;
     System.out.println("      " + dimes + " dimes");
     amount = amount - dimes * 10;
     int nickels = amount / 5;
     System.out.println("      " + nickels + " nickels");
     amount = amount - nickels * 5;
     int pennies = amount;
     System.out.println("      " + pennies + " pennies");
   }
}
```

16. Write an output statement which will output the following characters exactly as shown:

```
/ ' \" / ' \
```

ANS:
```
System.out.println("/ \ ' \ \ \" / \ ' \ \ ");
```

17. Provide three examples of code using assignment statements where one assignment statement would result in a syntax error, one would result in a logical error, and one would result in a run-time error.

ANS:
Answers will vary. Examples:
```
//  syntax error caused by right side providing a double and the
left side wants an int:
int x = 5.0 / 2.0;

// logical error caused by not casting the division as a double:
double x = (intValue1 + intValue2)  / 2;

// run-time error caused by division by 0 when y - y is evaluated:
int x = 5 / (y - y);
```

18. What are the syntax errors from the following program?
```
public class Enigma
{
   public static void main(String[] args)
   {
      System.out.println("Input a String");
      String x = scan.nextString();
      int size = x.length;
      char last = x.charAt(size);
      System.out.println("The last character in your string
                              ", x, " is ", last);
   }
}
```

ANS:
There are 3 syntax errors. First, the `Scanner` class has not been imported so that the `scan.nextString()` will yield a syntax error. Second, the statement `x.length` requires parentheses as `length` is a method of the class `String`, so this is a message passed to `x`. Finally, the `System.out.println` statement is not valid because of the use of a comma instead of + to concatenate the various parts of the `String`. Note that in spite of these syntax errors, there is a run-time error. This error would arise because `size` will store the length of the `String`, but the last character will be at location `size-1` instead of `size`. Thus, the statement `char last = x.charAt(size);` will result in a run-time error because `size` is beyond the bounds of the `String x`.