Blue Pelican Java

Exercise, Quiz, & Test Keys

by Charles E. Cook

Version 7.0.1A

This page is intentionally left blank.

Keys for Quizzes/Exercises/Projects

The short quizzes for each lesson in this section are not comprehensive and not very difficult. Normally, only basic, superficial questions are asked. The general philosophy here is for the specter of a quiz to always be hanging over the student where he knows he must quickly acquire a general working knowledge of the subject but at the same time knows he will not be asked in-depth or tricky questions. It is hoped that this gentle, but persistent pressure, will encourage the student to keep current with his studies and be rewarded with a frequent "100" on these little quizzes. It is suggested that a quiz be given **the day after** a new lesson is introduced.

Quiz on Lesson 11

In the questions below, consider the following code:

```
int sum = 0;
for(int j = 0; j < 3; j++)
{
        sum = sum + 2;
}
System.out.println(sum);
```

1. Identify the control expression.

2. Identify the step expression.

3. Identify the initializing expression.

4. How many times does the loop iterate (repeat)?

5. What is printed?

Key to Quiz on Lesson 11

In the questions below, consider the following code:

```
int sum = 0;
for(int j = 0; j < 3; j++)
{
        sum = sum + 2;
}
System.out.println(sum);
```

1. Identify the control expression.
   j < 3

2. Identify the step expression.
   j++

3. Identify the initializing expression.
   int j = 0

4. How many times does the loop iterate (repeat)?
   3

5. What is printed?
   6

Key to Exercise on Lesson 11

In each problem below state what is printed unless directed otherwise.

1. int j = 0;
    for (int g = 0; g <5; g++)
            j++;
    System.out.println(j); //**5**

2. int s = 1;
    for (int j = 3; j >= 0; j--)
    {
            s = s + j;
    }
    System.out.println(s); //**7**

3. int p = 6;
    int m = 20, j;
    for (j = 1; j < p; j++); //Notice the semicolon on this line
    {
            m = m + j * j;
    }
    System.out.println(m); //**56….notice the ";" after the parenthesis above**

4. double a = 1.0;
    for (int j = 0; j < 9; j++)
    {
            a*=3;
    }
    System.out.println(j); //**won't compile….j is undefined outside the loop.**

5. for (int iMus = 0; iMus < 10; iMus++)
    {
            int b = 19 + iMus;
    }
    System.out.println(b); //**won't compile since the scope of b is limited to inside the loop**

6. double d = 100.01;
    int b = 0;
    for (int iMus = 0; iMus < 10; iMus++)
            b = 19 + iMus;
            d++;
    System.out.println(d); //**101.01…d++ is not inside the loop. Notice, no braces.**

7. Write a for-loop that will print the numbers 3, 6, 12, and 24
            for (n = 3; n <= 24; n = n* 2)
            {
      System.out.println(n);
            }

8. Write a for-loop that will print the numbers 24, 12, 6, 3

for (n = 24; n >= 3; n = n / 2)
{
   System.out.println(n);
}

9. int k = 0;
   for(j = 0; j <= 10; j++)
   {
         if (j == 5)
         {
               break;
         }
         else
         {
               k++;
         }
   }
   System.out.println(k); //**5**

10. What is the name of the part of the parenthesis of a for-loop that terminates the loop?
   The control expression

11. What is the value of *j* for each iteration of the following loop?
   int i, j;
   for(i = 10; i <= 100; i = i+ 10)
         j = i / 2;
   5, 10, 15, 20, 25, 30, 35, 40, 45, 50

12. What is the value of *r* after the following statements have executed?
   int r, j;
   for (j = 1; j < 10; j = j * 2)
   r = 2 * j;
   16

13. What is the worst sin you can commit with a for-loop (or any loop for that matter)?
   Causing it to be an endless loop

14. How many times does the following loop iterate?
   for (p = 9; p <= 145; p++)
   {
         . . .
   }
   137

Project… Name Reversal, Key

```
import java.io.*;
import java.util.*;
public class Tester
{
   public static void main(String args[])
   {
      Scanner kbReader = new Scanner(System.in);
      System.out.print("Please enter your name. ");
      String name = kbReader.nextLine( );

      int strLen = name.length( );
      String reversedName = "";
      for(int j = strLen - 1; j >= 0; j--)
      {
         String letter = name.substring(j, j + 1);  //picks up just one letter at a time
         reversedName = reversedName + letter;
      }
      System.out.println(reversedName);
   }
}
```

It appears that the first time through the loop that the *j + 1* parameter of *substring* is illegal. It is indeed larger than the index of the last letter; however, when this happens, the *substring* method interprets this as us wanting to just go all the way to the end of the *String*.

■■

There is an easier and more conventional way to do this program…using the *charAt( )* method. We will learn about this method in a later lesson. Below is the same program using *charAt( )*.

```
import java.io.*;
import java.util.*;
public class Tester
{
   public static void main(String args[])
   {
      Scanner kbReader = new Scanner(System.in);
      System.out.print("Please enter your name. ");
      String name = kbReader.nextLine( );

      int strLen = name.length( );
      String reversedName = "";
      for(int j = strLen - 1; j >= 0; j--)
      {
         char letter = name.charAt(j);  //picks up just one letter at a time
         reversedName = reversedName + letter;
      }
      System.out.println(reversedName);
   }
}
```

Key for for-loop… Contest Type Problems

1. B

2. C

3. A

4. E

5. D

6. A

7. D

Quiz on Lesson 39

1. What is the value of 5! (five factorial)?

2. What is returned by *method(8)*?

```
public static int method(int n)
{
        if(n = = 5)
        {
                return 20;
        }
        else
        {
                return n + method(n - 1);
        }
}
```

3. What is returned by *method(1)*?

```
public static int method(int n)
{
        if(n > 5)
        {
                return n - 1;
        }
        else
        {
                return n * method(n + 2);
        }
}
```

Key to Quiz on Lesson 39

1. What is the value of 5! (five factorial)?
   120

2. What is returned by *method(8)*?
   ```
   public static int method(int n)
   {
           if(n = = 5)
           {
                   return 20;
           }
           else
           {
                   return n + method(n - 1);
           }
   }
   ```
   $8 + 7 + 6 + 20 = $ **41**

3. What is returned by *method(1)*?
   ```
   public static int method(int n)
   {
           if(n > 5)
           {
                   return n - 1;
           }
           else
           {
                   return n * method(n + 2);
           }
   }
   ```
   $1 * 3 * 5 * 6 = $ **90**

Key to Exercises on Lesson 39

In each of the following recursion problems, state what's printed.

1. System.out.println(rig(4));

```
public static int rig(int n)
{
        if ( (n = = 0) )
        {
                return 5;
        }
        else if ( n = = 1)
        {
                return 8;
        }
        else
        {
                return rig(n – 1) - rig(n – 2);
        }
}
```
0	1	2	3	4
5	8	3	-5	**-8**

2. System.out.println(mm(6)); // 6 + 5 + 4 + 3 + 2 + 1 + 10 = **31**

```
public static int mm(int n)
{
        if (n<=0)
                return 10;
        else
                return n + mm(n-1);
}
```

3. System.out.println(adrml(5)); // 5 * 3 * 1 = **15**

```
public static int adrml(int n)
{
        if (n<=1)
                return n;
        else
                return n * adrml(n-2);
}
```

4. System.out.println(bud(1)); // 1 + 2 + 3 + 4 + 5 + 4 = **19**

```
public static int bud(int n)
{
        if (n>5)
                return n - 2;
        else
                return n + bud(n +1);
}
```

5. System.out.println(zing(0)); // 0 + 6 + 24 + 24 = **54**

```
public static int zing(int n)
{
        if (n > 10)
                return n - 2;
        else
        {
                n = n * 3;
                return n + zing(n + 2);
        }
}
```

6. crch(12);

```
public static void crch(int n)
{
        if (n <= 0)
                System.out.print(n);
        else
        {
                crch(n / 3);
                System.out.print("," + n);
        }
} //0,1,4,12
```

7. elvis(11); //**3>>4>>7>>10**

```
public static void elvis(int n)
{
        if (n <= 3)
                System.out.print(n + 1);
        else
        {
                elvis(n-3);
                System.out.print(">>" + (n – 1));
        }
}
```

8. sal(5);

    ```
    public static int sal(int n)
    {
            if (n == 2)
            {
                    return 100;
            }
            else if (n == 3)
            {
                    return 200;
            }
            else
            {
                    return  (2 * sal(n - 1) +  sal(n - 2) + 1);
            }
    }
    ```

2	3	4	5
100	200	501	**1203**

9. puf(4);

    ```
    public void puf(int n)
    {
            if(n == 1)
            {
                    System.out.print("x");
            }
            else if( n%2 == 0) //n is even
            {
                    System.out.print("{");
                    puf(n-1);
                    System.out.print("}");
            }
            else  //n is odd
            {
                    System.out.print("<");
                    puf(n-1);
                    System.out.print(">");
            }
    }
    ```

1	2	3	4
x	{x}	<{x}>	**{<{x}>}**

10. bc(6, 2);

```
public static void bc(int p, int q)
{
        if (p/q == 0)
        {
                System.out.println(p + q + 1);
        }
        else
        {
                System.out.println(p);
                 bc(p/q, q);
        }
}
```

6, 2	3, 2	1, 2	
6	3	4	**6**
			3
			4

Project… Fibonacci, Key

```
public class ModFib
{
      public static int modFibonacci(int n)
      {
              if(n == 0)
              {
                 return 3;
              }
              else if(n == 1)
              {
                 return 5;
              }
              else if(n == 2)
              {
                 return 8;
              }
              else
              {
                 return modFibonacci(n-1) + modFibonacci(n-2) + modFibonacci(n-3);
              }
      }
}
```