**Chapter 10 Thinking in Objects**

*Section 10.2 Immutable Objects and Classes*

***10.1***  Which of the following statements are true about an immutable object?

 A. The contents of an immutable object cannot be modified.

 B. All properties of an immutable object must be private.

 C. All properties of an immutable object must be of primitive types.

 D. An object type property in an immutable object must also be immutable.

 E. An immutable object contains no mutator methods.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.3 Scope of Variables*

***10.2***  What is the printout for the first statement in the main method?  
public class Foo {  
  static int i = 0;  
  static int j = 0;  
  
  public static void main(String[] args) {  
    int i = 2;  
    int k = 3;  
    {  
      int j = 3;  
      System.out.println("i + j is " + i + j);  
    }  
  
    k = i + j;  
    System.out.println("k is " + k);  
    System.out.println("j is " + j);  
  }  
}

 A. i + j is 5

 B. i + j is 6

 C. i + j is 22

 D. i + j is 23

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.3***  What will be displayed from the statement System.out.println("k is " + k) in the main method?  
public class Foo {  
  static int i = 0;  
  static int j = 0;  
  
  public static void main(String[] args) {  
    int i = 2;  
    int k = 3;  
    {  
      int j = 3;  
      System.out.println("i + j is " + i + j);  
    }  
  
    k = i + j;  
    System.out.println("k is " + k);  
    System.out.println("j is " + j);  
  }  
}

 A. k is 0

 B. k is 1

 C. k is 2

 D. k is 3

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.4***  What will be displayed from the statement System.out.println("j is " + j) in the main method?  
public class Foo {  
  static int i = 0;  
  static int j = 0;  
  
  public static void main(String[] args) {  
    int i = 2;  
    int k = 3;  
    {  
      int j = 3;  
      System.out.println("i + j is " + i + j);  
    }  
  
    k = i + j;  
    System.out.println("k is " + k);  
    System.out.println("j is " + j);  
  }  
}

 A. j is 0

 B. j is 1

 C. j is 2

 D. j is 3

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.5***  You can declare two variables with the same name in \_\_\_\_\_\_\_\_\_\_.

 A. a method one as a formal parameter and the other as a local variable

 B. a block

 C. two nested blocks in a method (two nested blocks means one being inside the other)

 D. different methods in a class

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.4 The this Keyword*

***10.6***  Analyze the following code:  
  
class Circle {  
  private double radius;  
    
  public Circle(double radius) {  
    radius = radius;  
  }  
}

 A. The program has a compilation error because it does not have a main method.

 B. The program will compile, but you cannot create an object of Circle with a specified radius. The object will always have radius 0.

 C. The program has a compilation error because you cannot assign radius to radius.

 D. The program does not compile because Circle does not have a default constructor.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.7***  Analyze the following code:  
  
class Test {  
  private double i;  
    
  public Test(double i) {  
    this.t();  
    this.i = i;  
  }  
  
  public Test() {  
    System.out.println("Default constructor");  
    this(1);  
  }  
  
  public void t() {  
    System.out.println("Invoking t");  
  }  
}

 A. this.t() may be replaced by t().

 B. this.i may be replaced by i.

 C. this(1) must be called before System.out.println("Default constructor").

 D. this(1) must be replaced by this(1.0).

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.8***  Which of the following can be placed in the blank line in the following code?  
public class Test {  
  private int id;  
    
  public void m1() {  
    \_\_\_\_\_.id = 45;  
  }  
}

 A. this

 B. Test

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.9***  Analyze the following code:  
public class Test {  
  public static void main(String[] args) {  
    String firstName = "John";  
    Name name = new Name(firstName, 'F', "Smith");  
    firstName = "Peter";  
    name.lastName = "Pan";  
    System.out.println(name.firstName + " " + name.lastName);  
  }  
}  
  
class Name {  
  String firstName;  
  char mi;  
  String lastName;  
    
  public Name(String firstName, char mi, String lastName) {  
    this.firstName = firstName;  
    this.mi = mi;  
    this.lastName = lastName;  
  }  
}

 A. The program displays Peter Pan.

 B. The program displays John Pan.

 C. The program displays Peter Smith.

 D. The program displays John Smith.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.10***  Analyze the following code:  
public class Test {  
  public static void main(String[] args) {  
    MyDate birthDate = new MyDate(1990, 5, 6);  
    Name name = new Name("John", 'F', "Smith", birthDate);  
    birthDate = new MyDate(1991, 1, 1);  
    birthDate.year = 1992;  
    System.out.println(name.birthDate.year);  
  }  
}  
  
class MyDate {  
  int year;  
  int month;  
  int day;  
    
  public MyDate(int year, int month, int day) {  
    this.year = year;  
    this.month = month;  
    this.day = day;  
  }  
}  
  
class Name {  
  String firstName;  
  char mi;  
  String lastName;  
  MyDate birthDate;  
    
  public Name(String firstName, char mi, String lastName, MyDate birthDate) {  
    this.firstName = firstName;  
    this.mi = mi;  
    this.lastName = lastName;  
    this.birthDate = birthDate;  
  }  
}

 A. The program displays 1990.

 B. The program displays 1991.

 C. The program displays 1992.

 D. The program displays no thing.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.7 Object Composition*

***10.11***  \_\_\_\_\_\_\_\_\_\_\_ is attached to the class of the composing class to denote the aggregation relationship with the composed object.

 A. An empty diamond

 B. A solid diamond

 C. An empty oval

 D. A solid oval

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.12***  An aggregation relationship is usually represented as \_\_\_\_\_\_\_\_\_\_ in \_\_\_\_\_\_\_\_\_\_\_.

 A. a data field/the aggregating class

 B. a data field/the aggregated class

 C. a method/the aggregating class

 D. a method/the aggregated class

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.11 Class Design Guidelines*

***10.13***  Which of the following statements are true?

 A. A class should describe a single entity and all the class operations should logically fit together to support a coherent purpose.

 B. A class should always contain a no-arg constructor.

 C. The constructors must always be public.

 D. The constructors may be protected.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.14***  Which of the following is poor design?

 A. A data field is derived from other data fields in the same class.

 B. A method must be invoked after/before invoking another method in the same class.

 C. A method is an instance method, but it does not reference any instance data fields or invoke instance methods.

 D. A parameter is passed from a constructor to initialize a static data field.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.12 Processing Primitive Data Type Values as Objects*

***10.15***  Which of the following statements will convert a string s into i of int type?

 A. i = Integer.parseInt(s);

 B. i = (new Integer(s)).intValue();

 C. i = Integer.valueOf(s).intValue();

 D. i = Integer.valueOf(s);

 E. i = (int)(Double.parseDouble(s));

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.16***  Which of the following statements will convert a string s into a double value d?

 A. d = Double.parseDouble(s);

 B. d = (new Double(s)).doubleValue();

 C. d = Double.valueOf(s).doubleValue();

 D. All of the above.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.17***  Which of the following statements convert a double value d into a string s?

 A. s = (new Double(d)).toString();

 B. s = (Double.valueOf(s)).toString();

 C. s = new Double(d).stringOf();

 D. s = String.stringOf(d);

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.18***  Which of the following statements is correct?

 A. Integer.parseInt("12", 2);

 B. Integer.parseInt(100);

 C. Integer.parseInt("100");

 D. Integer.parseInt(100, 16);

 E. Integer.parseInt("345", 8);

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.19***  What is the output of Integer.parseInt("10", 2)?

 A. 1;

 B. 2;

 C. 10;

 D. Invalid statement;

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.13 Automatic Conversion Between Primitive Types and Wrapper Class Types*

***10.20***  In JDK 1.5, you may directly assign a primitive data type value to a wrapper object. This is called \_\_\_\_\_\_\_\_\_\_\_\_\_\_.

 A. auto boxing

 B. auto unboxing

 C. auto conversion

 D. auto casting

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.21***  In JDK 1.5, analyze the following code.  
  
Line 1: Integer[] intArray = {1, 2, 3};  
Line 2: int i = intArray[0] + intArray[1];  
Line 3: int j = i + intArray[2];  
Line 4: double d = intArray[0];

 A. It is OK to assign 1, 2, 3 to an array of Integer objects in JDK 1.5.

 B. It is OK to automatically convert an Integer object to an int value in Line 2.

 C. It is OK to mix an int value with an Integer object in an expression in Line 3.

 D. Line 4 is OK. An int value from intArray[0] object is assigned to a double variable d.

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

*Section 10.14 The BigInteger and BigDecimal Classes*

***10.22***  To create an instance of BigInteger for 454, use

 A. BigInteger(454);

 B. new BigInteger(454);

 C. BigInteger("454");

 D. new BigInteger("454");

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.23***  To create an instance of BigDecimal for 454.45, use

 A. BigInteger(454.45);

 B. new BigInteger(454.45);

 C. BigInteger("454.45");

 D. new BigDecimal("454.45");

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.24***  BigInteger and BigDecimal are immutable

 A. true

 B. false

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.25***  To add BigInteger b1 to b2, you write \_\_\_\_\_\_\_\_\_.

 A. b1.add(b2);

 B. b2.add(b1);

 C. b2 = b1.add(b2);

 D. b2 = b2.add(b1);

 E. b1 = b2.add(b1);

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.26***  What is the output of the following code?  
  
public class Test {  
  public static void main(String[] args) {  
    java.math.BigInteger x = new java.math.BigInteger("3");  
    java.math.BigInteger y = new java.math.BigInteger("7");  
    x.add(y);  
    System.out.println(x);  
  }  
}

 A. 3

 B. 4

 C. 10

 D. 11

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.27***  To divide BigDecimal b1 by b2 and assign the result to b1, you write \_\_\_\_\_\_\_\_\_.

 A. b1.divide(b2);

 B. b2.divide(b1);

 C. b1 = b1.divide(b2);

 D. b1 = b2.divide(b1);

 E. b1 = b2.divide(b1);

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.28***  Which of the following classes are immutable?

 A. Integer

 B. Double

 C. BigInteger

 D. BigDecimal

 E. String

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg

***10.29***  Which of the following statements are correct?

 A. new java.math.BigInteger("343");

 B. new java.math.BigDecimal("343.445");

 C. new java.math.BigInteger(343);

 D. new java.math.BigDecimal(343.445);

Your answer is correct http://www.cs.armstrong.edu/liang/image/correct.jpg