You need to refactor the following code without changing its functionality. Line numbers are included for reference only.

```
01 displayMenu();
02 for ( ; ; )
03 {
04    boolean b = makeNewTransaction();
05    if (!b)
06       break;
07    displayMenu();
08 }
```

Complete the code by moving the appropriate code segments from the list on the left to the correct positions on the right. You may use each code segment once, more than once, or not at all.

Code Segments

```
} break;
```

```
do
```

```
} do;
```

```
while (makeNewTransaction())
```

```
} while (makeNewTransaction());
```

```
}
```

Answer Area

```
{

displayMenu();
```

Go To Summary   Unmark for Review ✔   Mark for Feedback   Tools ▾

Back   Next

Which code segment maintains the original array elements when exiting the loop?

A.
```
for(int i = 0; i < intArray.length; i++)
{
    intArray[i] *= 2;
}
```

B.
```
for(Integer i : intArray)
{
    i *= 2;
}
```
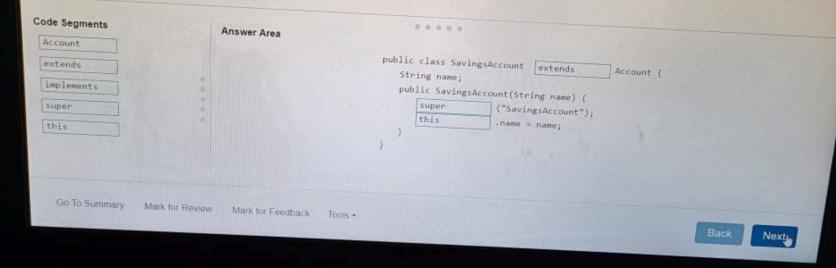
C.
```
int count = 0;
do
{
    intArray[count] *= 2;
    count++;
}
while(count < intArray.length);
```

D.
```
int count = 0;
while(count < intArray.length)
{
    intArray[count] *= 2;
    count++;
}
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You have a Java class named Account. The constructor of the Account class accepts a String object.

You need to create a class named SavingsAccount that inherits from the Account class. The constructor of the SavingsAccount class perform the following tasks:

- Accept a String parameter named name.
- Pass a String value SavingsAccount to the constructor of the Account class.
- Initialize the class member name with the value of the constructor parameter name.

Complete the code by moving the appropriate code segments from the list on the left to the correct positions on the right. You may use each code segment once, more than once, or not at all.

Note: You will receive partial credit for each correct selection.

Code Segments

Account

extends

implements

super

this

Answer Area

```
public class SavingsAccount  extends   Account {
    String name;
    public SavingsAccount(String name) {
        super      ("SavingsAccount");
        this       .name = name;
    }
}
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Review the following code:

```
public class Handler {

    static int[] a;
    static {
        a[0]=2;
    }

    public static void main( String[] args ) {
        try {
            int number = a[0]++/3;
            int result = a[0]/number;
            System.out.println("The final result is: " + result);
        } catch(Exception e) {
            if(e instanceof ArithmeticException) {
                System.out.println("The exception name is: " +e.getClass().
            } else {
                System.out.println("The exception name is: " +e.getClass().getC
            }
        }
    }
}
```

What is the output when you run the program?

○ A. The final result is: 2

◉ B. java.lang.ExceptionInInitializerError

○ C. The exception name is: java.lang.RuntimeException

○ D. The exception name is: java.lang.ArithmeticException

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

The Movie class includes a method isInMovie that is accessible only from within the class. The isInMovie method accepts the name of an actor to compare against a list of actors.

Complete the code by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

```
private   ▼     bool  ▼     isInMovie  (Movie actor) ▼      {
    for (int i = 0; i < numActors; i++) {
        if (actors[i] .= actor) return true;     ▼
    }
    return false;  ▼
}
```

You are developing an application that reads and writes files. The application must perform the following tasks:

- Display exceptions that are due to file handling errors.
- Display the stack information if any other exception occurs.

Complete the code by selecting the correct code segment from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

```
try
{
    /* add logic */
}
catch ( [IOException          ▼] e1)
{
    System.out.println(e1. [printStackTrace()  ▼]);
}
catch ( [Exception          ▼] e2)
{
    System.out.println(e2. [printStackTrace()  ▼]);
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You work as a Java programmer for a bank.

Evaluate the following class. Line numbers are included for reference only.

```
01 public class Account {
02    protected int balance;
03    public Account() {
04       balance = 0;
05    }
06    public Account(int amount) {
07       balance = amount;
08    }
09 }
```

Answer Area

	True	False
The Account class has a single constructor.	○	●
Other classes can inherit the Account class.	●	○
Line 07 is equivalent to this.balance = amount;	●	○

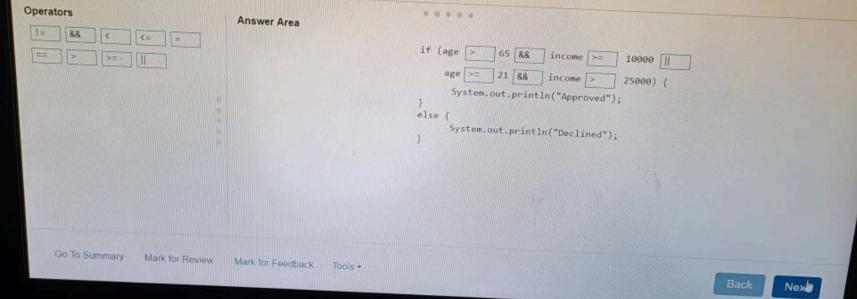Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

The constructor method for the Rectangle class takes integer arguments named length and width. length and width are declared in main and assigned values from user input.

You need to create an instance of the Rectangle class named rect1.

Complete the code by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

Rectangle ▼ rect1 ▼ = new ▼ Rectangle(length, width) ▼ ;

Back   Next

A company hires you to write a Java program to manage credit account openings. To open a new account, a person must meet one the following requirements:

- Be over 65 years old and have a minimum annual income of 10,000.
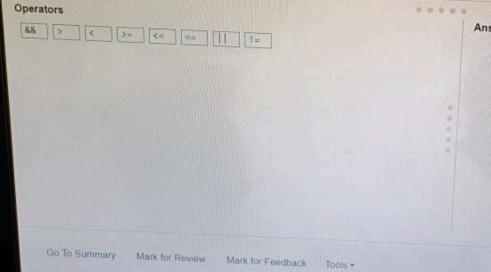- Be at least 21 and have an annual income greater than 25,000.

Complete the code by moving the appropriate operators from the list on the left to the correct positions on the right. You may use each operator once, more than once, or not at all. You might need to drag the split bar between panes or scroll to view content.

Operators

!=	&&	<	<=	=
==	>	>=	||	

Answer Area

```
if (age >    65 &&  income >=  10000 ||
    age >=  21 &&  income >   25000) {
        System.out.println("Approved");
}
else {
        System.out.println("Declined");
}
```

You need to create an if statement that evaluates to true when both of the following conditions are met:

- sum is greater than or equal to num
- cnt is smaller than num

Complete the code by moving the appropriate operators from the list on the left to the correct positions on the right. You may use each operator once, more than once, or not at all.

Operators

&&	>	<	>=	<=	==	||	!=

Answer Area

```
if (num <= sum && cnt < num)
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Evaluate the following code segment:

```
int x = 50;
x += 100 % 5 + 10 * 2;
```

What is the final value of x?

- A. 0
- B. 10
- C. 20
- D. 50
- E. 70
- F. 110

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Evaluate the following exception:

```
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 5
    at itsjava.Program.arrayDisplay(Program.java:31)
    at itsjava.Program.beginProcess(Program.java:23)
    at itsjava.Program.main(Program.java:18)
```

For each statement, select **True** or **False**

Note: You will receive partial credit for each correct selection.

Answer Area

	True	False
The root cause of the exception is in the method `beginProcess()`	●	○
An error occurred on line 31.	●	○
Three methods were invoked before the error occurred.	●	○
The stack trace indicates that the exception was caused by a syntax error.	○	●

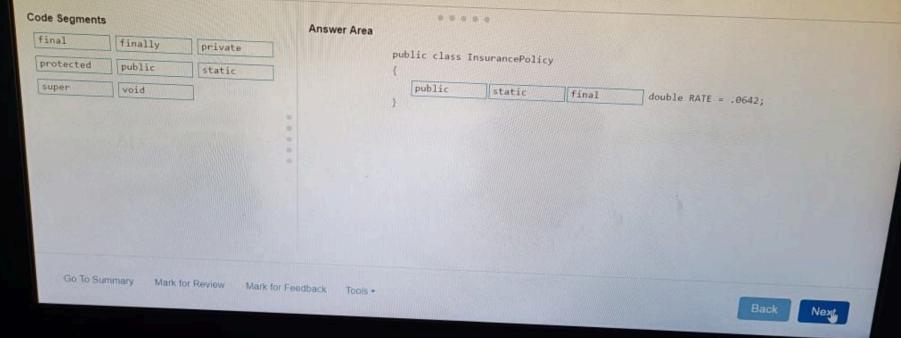Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

The following Java method calculates scholarship award amounts based on grade point average (gpa). Line numbers are included for reference only.

```
01  static double calculateAward(double gpa, int satScore, int actScore)
02  {
03      double award = 0;
04      if (gpa >= 3.7 && (satScore >= 1200 || actScore >= 26))
05      {
06          award = 30000;
07      }
08      else if (gpa >= 3.0 || satScore >= 1200 || actScore >= 26)
09      {
10          award = 15000;
11      }
```

Answer Area

What is the return value of `calculateAward(3.2, 1100, 28)`?   [15000 ▼]

What is the return value of `calculateAward(2.7, 1500, 30)`?   [15000 ▼]

What is the return value of `calculateAward(3.7, 1300, 23)`?   [30000 ▼]

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You have a Java class named InsurancePolicy.

You need to define a constant data member named RATE. The data member must be accessible by any class without instantiating the InsurancePolicy class.

Complete the code by moving the appropriate code segments from the list on the left to the correct positions on the right. You may use each code segment once, more than once, or not at all.

Code Segments

final	finally	private
protected	public	static
super	void	

Answer Area

```
public class InsurancePolicy
{
    public    static    final    double RATE = .0642;
}
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Evaluate the following code segment. Line numbers are included for reference only.

```
01 public static void main(String[] args) {
02    int anum = 55;
03    for (int cnt = 0; cnt < 10; cnt++) {
04       add(anum);
05    }
06    System.out.println(anum);
07 }
08
09 public static void add(int anum) {
10    anum++;
11 }
```

Answer Area

When cnt is 9, what is the value of anum after line 04 runs?   [55   ▼]

When cnt is 7, what is the value of anum after line 10 runs?   [56   ▼]

What is the value of cnt at line 06?   [10   ▼]

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Evaluate the following code segment. Line numbers are included for reference only.

```
01 byte value1 = 127;
02 value1++;
03 System.out.println(value1);
04 System.out.println(1.0 / 3.0);
05 System.out.println(1.0f / 3.0f);
```

Answer the questions by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

. . . . .

What is the output of line 03? | -128   ▼ |

What is the output of line 04? | 0.3333333333333333   ▼ |

What is the output of line 05? | 0.33333334   ▼ |

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Evaluate the following code segment. Line numbers are included for reference only.

```
01 String s1 = "Hello World";
02 String s2 = "Hello World";
03 String s3 = s2;
04
05
```

For each statement, select **True** or **False**

Note: You will receive partial credit for each correct selection.

Answer Area

	True	False
s1 and s2 refer to the same object in memory.	●	○
s2 and s3 refer to the same object in memory.	●	○
A different string can be assigned to s1 on line 04.	●	○
A different string can be assigned to s2 on line 05	●	○

Back   Next

You are writing a Java console program that accepts command-line arguments.

The program must display each command-line argument on a separate line.

Complete the code by selecting the correct code segment from each drop-down list.

Answer Area

• • • • •

```
public static void main(String[] args) {
    for        ▼  (String arg              ▼  args)
    {
        System.out.println( arg[0]    ▼ );
    }
}
```

You need to create an int array named numbers that is initialized with num1, num2, and num3.

You have the following code:

```
int num1 = 10;
int num2 = 20;
int num3 = 30;
```

Complete the code by moving the appropriate code segments from the list on the left to the correct positions on the right. You may use each code segment once, more than once, or not at all.

Code Segments

(num1, num2, num3)

[num1, num2, num3]

{num1, num2, num3}

int

int[]

new int

new int[]

Answer Area

int[]   numbers =   new int[]   {num1, num2, num3}   ;

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You work as a Java programmer. Your colleague creates the following class. Line numbers are included for reference only.

```
01 public class Rectangle {
02     private int width;
03     private int length;
04
05     Rectangle(int width, int length) {
06         this.width = width;
07         this.length = length;
08     }
09
10     public int area() {
11         return this.width * this.length;
12     }
13
14     public int getWidth() {
15         return width;
16     }
17
18     public int getLength() {
19         return length;
20     }
21 }
```

You need to write code to test the Rectangle class.

Complete the code by selecting the correct code segment from each drop-down list.

• • • • •

Answer Area

```
int areaNum;
Rectangle rect = new Rectangle(20, 40);  ▼
areaNum = rect area();  ▼
System.out.printf("Width = %d Length = %d\n", rect.getWidth(), rect.getLength()
System.out.printf("Area is correct %b\n", areaNum == 800);
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

What happens when the following code segment runs?

```
double dNum = 2.667;
int iNum = 0;
iNum = (int)dNum;
```

A   iNum has a value of 0

B   iNum has a value of 2

C   iNum has a value of 3

D   An exception is thrown

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

The Book.java program includes the following private data members:

- string title;
- string author;
- int year;
- int isbn;

The default constructor assigns empty strings and zeros accordingly, while the overloaded constructor takes arguments in the order listed above.

You need to complete the code to instantiate an empty textbook object for the user to update in a later function.

Complete the code by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

```
public static void main(String args[]){
    Book        ▼  book        ▼  =  new        ▼  Book()        ▼  ;
}
```

Back   Next

Evaluate the following code segment. Line numbers are included for reference only.

```
01 public static void main(String[] args) {
02    double pi = Math.PI; //3.141593
03    System.out.format("Pi is %.3f%n", pi);
04    System.out.format("Pi is %.0f%n", pi);
05    System.out.format("Pi is %09f%n", pi);
06 }
```

Answer the questions by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

· · · · ·

What is the output of line 03?	Pi is 3.142 ▼
What is the output of line 04?	Pi is 3 ▼
What is the output of line 05?	Pi is 0000000003.141593 ▼

Go To Summary   Mark for Review   Mark for Feedback   Tools ▼

Back   Next

You are writing a Java method named `countdown`. The method must perform the following tasks:

- Accept an `int` parameter named `start`.
- Display all numbers from `start` to zero in decrements of one.

Complete the code by selecting the correct code segment from each drop-down list.

Answer Area

• • • • •

```java
public static void countdown(int start) {
    for ( int i = start;   ▼   i >= 0   ▼   --i   ▼  ) {
        System.out.println(i);
    }
}
```

Evaluate the following code segment. Line numbers are included for reference only.

```
01 public static void method1() {
02      int x = 1;
03      int y = 1;
04
05      for (int i = 0; i < 10; i++) {
06          x += i;
07      }
08      for (int i = 0; i < 10; i++) {
09          y += i;
10      }
11 }
12
13 public static void method2() {
14      int i = 1;
15      int sum = 0;
16
17      for (int i = 0; i < 10; i++) {
18          sum += i;
19      }
20 }
```

For each statement, select **True** or **False**.

Answer Area

	True	False
The first method causes a compilation error because the variable i is declared in two non-nested blocks.	○	◉
The second method causes a compilation error because the variable i is declared in two nested blocks.	○	◉

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You are writing a Java method. The method must perform the following tasks:

- Accept an int parameter named age.
- Assign the senior classification if age is 65 or higher.
- Assign the adult classification if age is 20 or higher, but lower than 65.
- Otherwise, assign the youth classification.

Complete the code by selecting the correct code segment from each drop-down list.

Answer Area

```
public static String ageClassification(int age) {
    String classification;
    if (age >= 65)                ▼
        classification = "senior"
    else if (age >= 20)           ▼
        classification = "adult";
    else                          ▼
        classification = "youth";
    return classification;
}
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Nex

Review the following class definition

```
public class Box
{
   protected short minBoxWidth;
   protected short maxBoxWidth;
}
```

Which class(es) can access the minBoxWidth and maxBoxWidth data members?

A. Only the Box class

B. Only classes in the same package and classes that inherit the Box class.

C. Only classes that do not inherit from the Box class

D. All classes

Go To Summary   Mark for Review   Mark for Feedback   Tools •

Back   Next

You need to determine what happens when the following code runs. Line numbers are included for reference only.

```
01 char data1 = 65;
02 System.out.println(data1);
03
04 long data2 = 65;
05 System.out.println(data2);
06
07 float data3 = new Float("-65.0");
08 System.out.println(data3);
09
10 short data4 = new Short("65.0");
11 System.out.println(data4);
```

Answer Area

. . . . .

What happens when lines 01 and 02 are run?

The number 65 is displayed ▾

What happens when lines 04 and 05 are run?

The number 65 is displayed ▾

What happens when lines 07 and 08 are run?

An exception is thrown ▾

What happens when lines 10 and 11 are run?

An exception is thrown. ▾

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Which two code segments show the correct syntax for a multi-line comment? (Choose 2.)

☐ A.
```
//* This is a multi-
   line comment
//*
```

☒ B.
```
/* This is a multi-
   line comment
*/
```

☐ C.
```
/*//
      This is a multi-
      line comment
*/
```

☐ D.
```
//* This is a multi-
   line comment
*//
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You write the following Java program. Line numbers are included for reference only.

```
01 class Pickle {
02    boolean isPreserved = false;
03    private boolean isCreated = false;
04
05    void preserve() {
06       isPreserved = true;
07    }
08
09    public static void main(String[] args)
10    {
11       Pickle pickle = new pickle();
12       isCreated = true;
13       pickle.preserve;
14    }
15 }
```

You encounter error messages when you attempt to compile the program.

You need to ensure that the program compiles successfully.

Correct the code by selecting the correct code segment from each drop-down list.

Answer Area

```
class Pickle {
    boolean isPreserved = false;
    private boolean isCreated = false;

    void preserve() {
        isPreserved = true;
    }

    public static void main(String[] args)
    {
        Pickle pickle = new Pickle();   ▼
        pickle.isCreated = true         ▼
        pickle.preserve();              ▼
    }
}
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▼

Back   Next

You initialize the following variables:

```
int a = 5;
double b = 3.5;
int c = 33;
float d = 0.5f;
short e = 22;
```

Identify the value of each code segment by moving the appropriate values from the list on the left to the corresponding code segments on the right. You may use each value once, more than once, or not at all.

Note: You will receive partial credit for each correct selection.

Values

0.175	0.5
0.7	4.0
12	22
27	33

Answer Area

Code segment	Value
a += 4 % 2 + e;	27
b /= 10 * 2;	0.7
c %= e * 2 + 1;	33
d *= 2 + 6 % 7;	4.0

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You need to always print every element in the two-dimensional array.

Complete the code by selecting the correct option from the drop-down list.

Answer Area

• • • • •

```
for(int row=0; row < intArray.length; row++)
    for(int column = 0; column < intArray[row].length; column++)
        System.out.println(intArray[row][column]);
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Review the following class definition:

```
class Logger
{

    public void logError(String message)
    {

    }
}
```

What can invoke the logError method?

- ● A. Only code in all classes in the same package as the Logger class.

- ○ B. Only the Logger class.

- ○ C. Only the Logger class and classes in the same package that inherit from it.

- ○ D. All classes in all packages.

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You are interviewing for a job as a Java developer. You need to demonstrate your understanding of the main method.

For each statement, select **True** or **False**

Note: You will receive partial credit for each correct selection.

Answer Area

. . . . .

	True	False
The main method parameter args is an array of type String.	●	○
A Java application can accept only one argument from the command line.	○	●
The main method must be static because it is run without instantiating an instance of the class.	●	○

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

Review the following code:

```
public class Logger {
    public static void main(String[] args)
    {
        int[][] arr = { { 9, 15 }, { 18, 20, 40 } };
        for (int i = 2; i >= 0; i--) {
            for (int j = 2; j >= 0; j--) {
                System.out.print(arr[i][j] + " ");
            }
        }
    }
}
```

What is the output when you run the program?

A. `20 18 15 9`

B. `40 15 20 9 18`

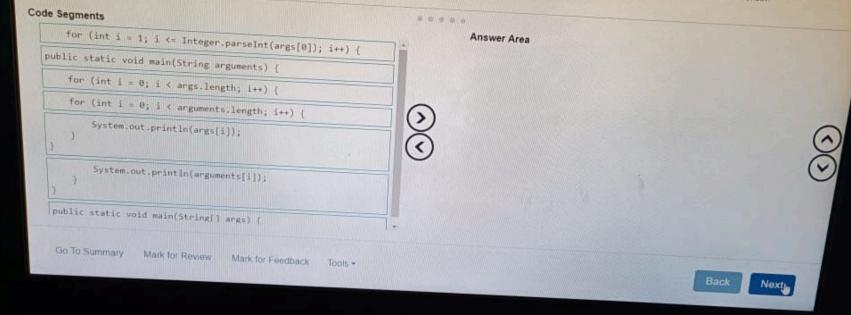C. `ArrayIndexOutOfBounds Exception`

D. `Compilation error`

Go To Summary   Mark for Review   Mark for Feedback   Tools

Back   Next

Complete the sentences about working with branching statements in Java by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

• • • • •

To terminate the inner loop of a nested for loop, use a [labeled break ▼] statement.

To terminate the outer loop of a nested for loop, use a [return ▼] statement.

To jump to the next iteration of an inner loop in a nested loop, use a [continue ▼] statement.

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You are interviewing for a job. The hiring manager asks you to create a simple console program. The program must perform the following tasks:

- Accept multiple arguments from the command line.
- Write the arguments to the screen in the same order the user enters them on the command line.

Which three code segments should you use in sequence to develop the solution? Move the three code segments to the answer area and place them in the correct order.

Code Segments

```
    for (int i = 1; i <= Integer.parseInt(args[0]); i++) {
public static void main(String arguments) {
    for (int i = 0; i < args.length; i++) {
    for (int i = 0; i < arguments.length; i++) {
        System.out.println(args[i]);
    }
}

        System.out.println(arguments[i]);
    }
}
public static void main(String[] args) {
```

Answer Area

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next

You are creating a class named Phone that must meet the following requirements:

- Include a method named dialNumber that cannot be overridden by derived classes.
- Only classes in the same package as the Phone class can call the dialNumber method.

Complete the code by selecting the correct option from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

```
class Phone {
    protected    ▼   final      ▼  void dialNumber(String phoneNumber) {
    }
}
```

Go To Summary   Mark for Review   Mark for Feedback   Tools ▼

Back   Next

You are creating a Java class to implement an Integer stack based on an ArrayList.

You need to implement the following two methods:

- The pop method removes an Integer from the beginning of the ArrayList and returns the removed Integer.
- The push method adds an Integer to the beginning of the ArrayList.

Complete the code by selecting the correct code segment from each drop-down list.

Note: You will receive partial credit for each correct selection.

Answer Area

```
public static Integer pop(ArrayList<Integer> stack) {
    int index = 0
    return stack.remove(index);
}
public static void push(ArrayList<Integer> stack, Integer item) {
    int index = 0
    stack.add(index, item);
}
```

Back   Nex

Evaluate the following Java program. Line numbers are included for reference only.

```
01 public static void main(String[] args)
02 {
03    int x = 5;
04    int y = 7;
05    String value1 = "x + y = " + x + y;
06    System.out.println(value1);
07    String value2 = "x + y = " + (x + y);
08    System.out.println(value2);
09 }
```

Answer the questions by selecting the correct option from each drop-down list.

Answer Area

What is the output of line 06?   [x + y = 57   ▼]

What is the output of line 08?   [x + y = 12   ▼]

Go To Summary   Mark for Review   Mark for Feedback   Tools ▾

Back   Next