

Sun Certified Java Programmer (SCJP) Mock Exams

<http://www.JavaChamp.com>

Open Certification Plattform

Authors: N. Ibrahim, Y. Ibrahim

Copyright (c) 2009-2010

Introducing JavaChamp.com Website

JavaChamp.com is an Open Certification Platform.

What does this mean?

JavaChamp is the best place to learn, share, and certify your professional skills.

We help you develop yourself in the field of computer science and programming

Here are the most significant features offered by JavaChamp:

Online Exams

Start Online Certification Exams in SCJP, SCEA, EJB, JMS, JPA and more...

Top quality mock exams for SCJP, SCEA, EJB, JMS, JPA. Start Express or topic-wise customized exam.

- * We offer you unlimited free mock exams
- * Exams cover subjects like SCJP, SCEA, EJB, JMS, JPA,..
- * You can take as many exams as you want and at any time and for no charges
- * Each exam contains 20 multiple choice questions
- * You can save the exams taken in your exams history
- * Your exams history saves the exams you took, the scores you got, time took you to finish the exam, date of examination and also saves your answers to the questions for later revision
- * You can re-take the same exam to monitor your progress
- * Your exams history helps the system to offer you variant new questions every time you take a new exam, therefore we encourage you to register and maintain an exams history

Network

Find guidance through the maze, meet Study-Mates, Coaches or Trainees...

Studying together is fun, productive and helps you in building your professional network and collecting leads

Bookshelf

JavaChamp Bookshelf full of PDF eBooks...

Download PDF books with a selected sample of the JavaChamp question bank in SCJP, SCEA, EJB, JMS and more or read it online

JavaChamp Profile

You may publish your profile and connect to your colleagues and friends.

Content Channel

Be an Author and get recognition, leads, and more...

Contributing to the JavaChamp question bank will earn your recognition of your professional skills, expands your network, introduce you to potential leads

Join Us

Join the fast growing JavaChamp Community now.

JavaChamp Community is young and very dynamic, we would be thrilled to welcome you on board :o)



→ Stop

★ About us

JavaChamp.com is a Java Open Certification Plattform!

- It provides unique services to help you develop yourself in the field of computer science and programming
- It provides a web interface for the Java community to learn, interact and certify its acquired java experience.
- JavaChamp.com helps the Java developers to achieve the

★ Why to register in our community?

Registration is optional and free but brings many advantages

- By registering in JavaChamp.com you can save and keep track of your exams for later revision and hence to help you monitor your progress.

CHECK OUR eBook

JavaChamp.com's eBook encompasses a large number of multiple questions, which cover the subjects java, C and assembly. The book doesn't ...

[More](#) →



Home

Take Exam

Log in

Register

Book

Search



Exam in Session

Question 3 / 20

What is the expected output?

```
01. public class OuterTest {
02.
03.     public static void main(String args[]) {
04.         Airplane.BlackBox box = new Airplane().new BlackBox(); // line 1
05.         box.printVariables();
06.
07.     }
08. }
09.
10. class Airplane {
11.     String code = "11";
12.
13.     class BlackBox {
14.         String code = "22";
15.
16.         public void printVariables() {
17.             System.out.print(code);
18.             System.out.print(Airplane.this.code); // line 20
19.
20.         }
21.     }
22. }
23. }
```

- ☐ Compile error because of line 1 (incorrect instantiation)
- ☐ Compile error because of line 20 (can't access Airplane's variables)
- ☐ 2222
- ☐ 1111
- ☐ 2211

Back Next

Finish and evaluate Abort Exam

CHECK OUR eBook

JavaChamp.com's eBook encompasses a large number of multiple questions, which cover the subjects java,C and assembly. The book doesn't ...

More →

Copyright

Copyright 2009 JavaChamp.com

Online version published by JavaChamp.com Germany.

DISCLAIMER

All services and content of JavaChamp.com are provided under JavaChamp.com terms of use on an "as is" basis, without warranty of any kind, either expressed or implied, including, without limitation, warranties that the provided services and content are free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the provided services and content is with you. In no event shall JavaChamp.com be liable for any damages whatsoever arising out of or in connection with the use or performance of the services. Should any provided services and content prove defective in any respect, you (not the initial developer, author or any other contributor) assume the cost of any necessary servicing, repair or correction. This disclaimer of warranty constitutes an essential part of these "terms of use". No use of any services and content of JavaChamp.com is authorized hereunder except under this disclaimer.

The detailed "terms of use" of JavaChamp.com can be found under:

<http://www.javachamp.com/public/termsOfUse.xhtml>

This work is licensed under a Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 license.

The full license legal code can be found under:

<http://creativecommons.org/licenses/by-nc-nd/3.0/legalcode>

And a human-readable summary of the this license can be found under:

<http://creativecommons.org/licenses/by-nc-nd/3.0/>

According to the Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 license You agree to the following:

You are free to share, copy, distribute and transmit the work under the following conditions:

- You must attribute the work to JavaChamp.com with a link to <http://www.javachamp.com>.
- You may not use this work for commercial purposes.
- You may not alter, transform, or build upon this work.

Table of Contents

1. Java Programming Language

- Strings
- Constructors
- Inner class
- Flow Control
- Declarations and Access Control
- Interfaces and Abstract Classes
- Exceptions
- Inheritance
- Arrays
- Operators
- Variables and Methods
- Wrappers and Autoboxing
- Garbage Collection
- Overriding and Overloading
- Collections
- Generics
- Formatting
- I/O
- Threads
- Enums
- Data Types
- Static and init blocks
- Serialization
- Typecasting

1. Chapter: Java Programming Language

Chapter Description and Objectives

1. Strings

Exam Category Description and Objectives

1.1.1. how to split String in java?

Author: [Java Champ](#)

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        String stmt = "JavaChamp is here to help you";  
        for (String token : stmt.split("//s")) {  
            System.out.print(token + " ");  
        }  
    }  
}
```

Please choose only one answer:

- JavaChamp is here to help you
- JavaChamp i here to help you
- No output is produced
- Compilation error

Check the answer of this question online on [JavaChamp.com](#): [how to split String in java?](#)

1.1.2. How to use java.util.Scanner?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.util.Scanner;

public class Tester {

    public static void main(String[] args) {
        Scanner sc = new Scanner("javachamp 2009, true 239");
        while (sc.hasNext()) {

            if (sc.hasNextBoolean())
                System.out.print("Boolean");
            if (sc.hasNextInt())
                System.out.print("Int");
            sc.next();
        }
    }
}
```

Please choose only one answer:

- IntBooleanInt
- BooleanInt
- IntInt
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to use java.util.Scanner?](#)

1.1.3. What are common methods between String and StringBuffer?

Author: [Java Champ](#)

Select the common methods, which are defined for both type String and type StringBuffer ?

Please choose all the answers that apply:

- toString()
- length()
- append(String)
- trim()
- equals(Object)

Check the answer of this question online on [JavaChamp.com](#): [What are common methods between String and StringBuffer?](#)

1.1.4. Java objects comparison

Author: Yasser Ibrahim

What is the expected output?

```
public static void main(String[] args) {  
    boolean stmt1 = "champ" == "champ";  
    boolean stmt2 = new String("champ") == "champ";  
    boolean stmt3 = new String("champ") == new String("champ");  
    System.out.println(stmt1 && stmt2 || stmt3);  
}
```

Please choose only one answer:

- true
- false

Check the answer of this question online on JavaChamp.com: [Java objects comparison](#)

1.1.5. Comparing StringBuffer using equals in java

Author: [Java Champ](#)

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    StringBuffer buffer1 = new StringBuffer("javachamp");  
    StringBuffer buffer2 = new StringBuffer(buffer1);  
    if (buffer1.equals(buffer2))  
        System.out.println("true");  
    else  
        System.out.println("false");  
}
```

Please choose only one answer:

- true
- false

Check the answer of this question online on [JavaChamp.com](#): [Comparing StringBuffer using equals in java](#)

1.1.6. Comparing String and StringBuffer in java

Author: Java Champ

Which of the statements will evaluate to true?

```
public class Tester {  
    public static void main(String[] args) {  
        StringBuffer sb1 = new StringBuffer(&quot;javachamp&quot;);  
        StringBuffer sb2 = new StringBuffer(&quot;javachamp&quot;);  
  
        boolean stmt1 =sb1.equals(sb2) ;  
        boolean stmt2 = sb1 == sb2;  
  
        String s1 = new String(&quot;javachamp&quot;);  
        String s2 = new String(&quot;javachamp&quot;);  
  
        boolean stmt3 = s1.equals(s2);  
        boolean stmt4 = s1 == s2;  
    }  
}
```

Please choose only one answer:

- stmt1
- stmt2
- stmt3
- stmt4

Check the answer of this question online on JavaChamp.com: [Comparing String and StringBuffer in java](#)

1.1.7. What is the difference between String and StringBuffer?

Author: Java Champ

Which of the statements would evaluate to true?

```
public class Tester {  
    public static void main(String[] args) {  
        StringBuffer sb = new StringBuffer("javachamp");  
        String s = new String("javachamp");  
        boolean stmt1 = s.equals(sb) ;  
        boolean stmt2 = sb.equals(s) ;  
        boolean stmt3 = sb.toString() == s ;  
        boolean stmt4 = sb.toString().equals(s) ;  
        boolean stmt5 = s.equals(sb.toString()) ;  
    }  
}
```

Please choose all the answers that apply:

- stmt1
- stmt2
- stmt3
- stmt4
- stmt5

Check the answer of this question online on JavaChamp.com: [What is the difference between String and StringBuffer?](#)

1.1.8. What are java StringBuffer methods?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        String str = "java";  
        StringBuffer sb = new StringBuffer("javachamp");  
        sb.insert(9, ".com");  
        str.concat("champ");  
        if (sb.length() < 6 || str.equals("javachamp")) {  
            System.out.print(sb);  
        }  
        sb.delete(2, 7);  
        System.out.print(sb);  
    }  
}
```

Please choose only one answer:

- javachamp.comjump.com
- jump
- jump.com
- javachampjump
- jap

Check the answer of this question online on JavaChamp.com: [What are java StringBuffer methods?](#)

1.1.9. Comparing Strings for equality in java

Author: Yasser Ibrahim

What is the result of compiling and running the following program?

```
public class Tester {  
    public static void main(String[] args) {  
        String a = "javachamp";  
        String b = "javachamp";  
        String c = new String("javachamp");  
  
        System.out.print(a==b);  
        System.out.print(a==c);  
        System.out.print(b.equals(c));  
        System.out.print(b.equals(a));  
    }  
}
```

Please choose only one answer:

- Compilation error
- falsefalsetrue
- true>true>true
- truefalsetrue

Check the answer of this question online on JavaChamp.com: [Comparing Strings for equality in java](#)

1.1.10. comparing null to String

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    String s1 = null;  
    String s2 = null;  
  
    if (s1 == s2)  
        System.out.print("A");  
  
    if (s1.equals(s2))  
        System.out.print("B");  
}
```

Please choose only one answer:

- "AB" will be printed
- "A" will be printed followed by a NullPointerException thrown
- "B" will be printed
- No output is produced

Check the answer of this question online on JavaChamp.com: [comparing null to String](#)

1.1.11. How to use StringBuffer in java?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        String s = "";  
        Integer x = 5;  
        StringBuffer sb = new StringBuffer();  
        if (x < 15)  
            s.concat("javachamp");  
        else  
            sb.append("javachamp");  
        System.out.print(s + sb);  
    }  
}
```

Please choose only one answer:

- javachamp
- javachamp javachamp
- No output is produced
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to use StringBuffer in java?](#)

1.1.12. How to parse String in java?

Author: Java Champ

Given the following code, what is the expected output?

```
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Tester {

    public static void main(String[] args) {
        Pattern p = Pattern.compile("[a-f]\\d+");
        Matcher m = p.matcher("ab34ef0");
        while(m.find()) {
            System.out.print(m.start()+" "+m.group());
        }
    }
}
```

Please choose only one answer:

- 0 ab345 f0
- 0 ab344 ef0
- 1 b35 f0
- 1 b345 f0

Check the answer of this question online on JavaChamp.com: [How to parse String in java?](#)

1.1.13. Methods of java regex Pattern?

Author: Yasser Ibrahim

Which of the following methods can be invoked by an object of Pattern class?

Please choose all the answers that apply:

- compile
- matches
- group
- toString

Check the answer of this question online on JavaChamp.com: [Methods of java regex Pattern?](#)

1.1.14. Concatenating String and null

Author: Yasser Ibrahim

What is the expected output?

```
public static void main(String args []) {  
    String stmt = null;  
        System.out.print(null+stmt);  
        System.out.print(stmt+null);  
}
```

Please choose only one answer:

- RuntimeException is thrown because of the first print statement
- RuntimeException is thrown because of the second print statement
- nullnullnullnull
- nullnull
- compilation error

Check the answer of this question online on JavaChamp.com: [Concatenating String and null](#)

1.1.15. How to use java StringBuffer?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        String s = "";  
        Integer x = 5;  
        StringBuffer sb = "";  
        if (x < 0)  
            s.concat("javachamp");  
        else  
            sb.append("javachamp");  
        System.out.print(s + sb);  
    }  
}
```

Please choose only one answer:

- javachamp
- javachamp javachamp
- No output is produced
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to use java StringBuffer?](#)

1.1.16. How to compare java objects?

Author: [Java Champ](#)

What is the expected output?

```
public static void main(String[] args) {  
    boolean stmt1 = "champ" == "champ";  
    boolean stmt2 = new String("champ").equals(new String("champ"));  
    boolean stmt3 = "champ".toString()=="champ";  
  
    System.out.println(stmt1 && stmt2 && stmt3);  
}
```

Please choose only one answer:

- true
- false

Check the answer of this question online on JavaChamp.com: [How to compare java objects?](#)

2. Constructors

Exam Category Description and Objectives

1.2.1. Can java constructors be marked static?

Author: Java Champ

What is the expected output?

```
public class Profile {  
    private Profile(int w) { // line 1  
        System.out.println(w);  
    }  
  
    public static Profile() { // line 5  
        System.out.println(10);  
    }  
  
    public static void main(String args[]) {  
        Profile obj = new Profile(50);  
    }  
}
```

Please choose only one answer:

- Won't compile because of line (1) – constructor can't be private
- 10
- 50
- 50
- Won't compile because of line (5) – constructor can't be static

Check the answer of this question online on JavaChamp.com: [Can java constructors be marked static?](#)

1.2.2. What are java constructors legal modifiers?

Author: [Java Champ](#)

What are the legal modifiers which the constructor can be declared with?

Please choose all the answers that apply:

- public
- protected
- private
- final
- static
- abstract

Check the answer of this question online on JavaChamp.com: [What are java constructors legal modifiers?](#)

1.2.3. When can call super() in java constructors?

Author: Yasser Ibrahim

What is the expected output?

```
class Plant {
    Plant() {
        System.out.println("Plant created");
    }
}

class Tree extends Plant {
    Tree() {
        System.out.println("Tree created");
        super();
    }
}

public class Test {
    public static void main(String args[]) {
        Tree tree = new Tree();
    }
}
```

Please choose only one answer:

- Plant created
Tree created
- Tree created
Plant created
- RuntimeException
- Compilation error

Check the answer of this question online on JavaChamp.com: [When can call super\(\) in java constructors?](#)

1.2.4. Can java constructors be declared static?

Author: Yasser Ibrahim

Can a constructor be declared static?

Please choose only one answer:

- Yes
- No

Check the answer of this question online on JavaChamp.com: [Can java constructors be declared static?](#)

1.2.5. Can java constructors be declared final?

Author: Java Champ

What is the expected output?

```
public class Profile {  
    private Profile(int w) { // line 1  
        System.out.println(w);  
    }  
  
    public final Profile() { // line 5  
        System.out.println(10);  
    }  
  
    public static void main(String args[]) {  
        Profile obj = new Profile(50);  
    }  
}
```

Please choose only one answer:

- Won't compile because of line (1) – constructor can't be private
- Won't compile because of line (5) – constructor can't be final
- 50
- 10
- 50

Check the answer of this question online on JavaChamp.com: [Can java constructors be declared final?](#)

1.2.6. How to call this() in java constructor?

Author: Yasser Ibrahim

The following code contains one compilation error, where could it be?

```
public class Tester {  
    Tester() { } // line 1  
    static void Tester() { this(); } // line 2  
  
    public static void main(String[] args) { // line 3  
        Tester(); // line 4  
    }  
}
```

Please choose only one answer:

- At line 1, constructor Tester must be marked public like its class
- At line 2, constructor call "this()" can only be called inside constructors
- At line 3, compilation error, ambiguity problem, compiler can't determine whether a constructor Tester or method Tester is called

Check the answer of this question online on JavaChamp.com: [How to call this\(\) in java constructor?](#)

1.2.7. How to call super() in java constructor?

Author: Java Champ

What is the expected output?

```
import java.io.IOException;

class AirPlane {
    public AirPlane() throws IOException {
        System.out.print("AirPlane");
        throw new IOException();
    }
}

class AirJet extends AirPlane {
    public AirJet() throws IOException {
        try {
            super();
        } catch (IOException e) {
            System.out.print("IOException is thrown in AirJet");
        }
    }
}

public class Tester {
    public static void main(String args[]) {
        try {
            new AirJet();
        } catch (IOException e) {
            System.out.print("IOException is thrown in Tester");
        }
    }
}
```

Please choose only one answer:

- "AirPlaneIOException is thrown in AirJet" will be printed
- "AirPlaneIOException is thrown in AirJetIOException is thrown in Tester" will be printed
- "AirPlaneIOException is thrown in Tester" will be printed
- Compile error

Check the answer of this question online on JavaChamp.com: [How to call super\(\) in java constructor?](#)

3. Inner class

Exam Category Description and Objectives

1.3.1. How to instantiate an inner class inside main()?

Author: Java Champ

Which of the instantiations for a BlackBox are correct and compile-free?

```
public class OuterTest {  
    public static void main(String args[]) {  
        // instantiation 1  
        Airplane airplane = new Airplane();  
        Airplane.BlackBox box1 = airplane.new BlackBox();  
  
        // instantiation 2  
        Airplane.BlackBox box2 = new Airplane().new BlackBox();  
  
        // instantiation 3  
        Airplane airplane3 = new Airplane();  
        BlackBox box3 = airplane3.new BlackBox();  
    }  
}  
  
class Airplane {  
    class BlackBox {  
    }  
}
```

Please choose all the answers that apply:

- Instantiation 1
- Instantiation 2
- Instantiation 3

Check the answer of this question online on JavaChamp.com: [How to instantiate an inner class inside main\(\)?](#)

1.3.2. How java inner class access outer class?

Author: Java Champ

What statement,inserted independently at // insert statement here, would produce output "Body" ?

```
public class Body {  
    String name = "Body";  
  
    public static void main(String[] args) {  
        System.out.println(new Body().new Heart().getPName());  
    }  
  
    class Heart {  
        String name = "Heart";  
  
        String getPName() {  
            // insert statement here  
        }  
    }  
}
```

Please choose only one answer:

- return Body.name;
- return Body.this.name;
- return super.name;
- return this.super.name;

Check the answer of this question online on JavaChamp.com: [How java inner class access outer class?](#)

1.3.3. How to access an outer class member from inside its inner class?

Author: Java Champ

What is the expected output?

```
public class OuterTest {  
    public static void main(String args[]) {  
        Airplane airplane = new Airplane(); //line 5  
        Airplane.BlackBox box = airplane.new BlackBox(); // line 6  
        box.printVariables();  
    }  
}  
  
class Airplane {  
    String code = "11";  
  
    class BlackBox {  
        String code = "22";  
  
        public void printVariables() {  
            System.out.print(code); // line 19  
            System.out.print(Airplane.this.code); // line 20  
        }  
    }  
}
```

Please choose only one answer:

- Compile error because of lines 5-6 (incorrect instantiation)
- Compile error because of line 20 (can't access Airplane's variables)
- 2222
- 1111
- 2211

Check the answer of this question online on JavaChamp.com: [How to access an outer class member from inside its inner class?](#)

1.3.4. How to access a member of an outer class from inside its inner class?

Author: Java Champ

What is the expected output?

```
public class OuterTest {  
    public static void main(String args[]) {  
        Airplane.BlackBox box = new Airplane().new BlackBox(); // line 1  
        box.printVariables();  
    }  
}  
  
class Airplane {  
    String code = "11";  
  
    class BlackBox {  
        String code = "22";  
  
        public void printVariables() {  
            System.out.print(code);  
            System.out.print(Airplane.this.code); // line 20  
        }  
    }  
}
```

Please choose only one answer:

- Compile error because of line 1 (incorrect instantiation)
- Compile error because of line 20 (can't access Airplane's variables)
- 2222
- 1111
- 2211

Check the answer of this question online on JavaChamp.com: [How to access a member of an outer class from inside its inner class?](#)

1.3.5. How to instantiate an inner class?

Author: Java Champ

Which of the instantiations for a BlackBox are correct and compile-free?

```
public class AirJet {  
    public static void main(String[] args) {  
        // instantiation 1  
        Engine engine1 = new AirJet().new Engine();  
  
        // instantiation 2  
        AirJet.Engine engine2 = new AirJet().new Engine();  
  
        // instantiation 3  
        AirJet airjet = new AirJet();  
        Engine engine3 = airjet.new Engine();  
  
        // instantiation 4  
        Engine engine4 = new Engine();  
    }  
    class Engine{  
    }  
}
```

Please choose all the answers that apply:

- Instantiation 1
- Instantiation 2
- Instantiation 3
- Instantiation 4

Check the answer of this question online on JavaChamp.com: [How to instantiate an inner class?](#)

1.3.6. How to declare java inner class inside a method?

Author: Yasser Ibrahim

Are the lines 1 and 2 correct and free of compilation errors?

```
public class AirPlane {  
    public void fly(int speed) {  
        final int e = 1;  
        class FlyingEquation {  
            {  
                System.out.println(e); // line 1  
                System.out.println(speed); // line 2  
            }  
        }  
    }  
}
```

Please choose only one answer:

- Both statements at lines 1 and 2 are correct
- Both statements at lines 1 and 2 are incorrect and cause compilation errors
- Compilation error at line 1 , inner class can't access outer class local variables
- Compilation error at line 2, inner class can't access method's arguments

Check the answer of this question online on JavaChamp.com: [How to declare java inner class inside a method?](#)

4. Flow Control

Exam Category Description and Objectives

1.4.1. How to use break and continue in java for loop?

Author: Java Champ

What is the expected output?

```
public class LoopTest {  
    public static void goLoop(int a) {  
  
        start: for (int i = 1; i < 2; i++) {  
            for (int j = 1; j < 2; j++) {  
                if (a > 5) {  
                    break start;  
                }  
                System.out.print(i + j);  
            }  
        }  
    }  
}  
  
public static void main(String args[]) {  
    LoopTest.goLoop(15);  
}  
}
```

Please choose only one answer:

- 2
- No output is produced
- Compile Error
- Runtime exception is thrown
- 24

Check the answer of this question online on JavaChamp.com: [How to use break and continue in java for loop?](#)

1.4.2. How to write java switch statement?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int j = 10;  
    switch (1) {  
        case 20:  
            j += 1;  
        case 40:  
            j += 2;  
        default:  
            j += 3;  
        case 0:  
            j += 4;  
    }  
    System.out.print(j);  
}
```

Please choose only one answer:

- Compile error, can't use constant 1 in the switch, must be final variable
- Compile error, default must be the last statement after all the cases
- Compile error, must have break statement within each case
- 17
- 13

Check the answer of this question online on JavaChamp.com: [How to write java switch statement?](#)

1.4.3. do while loop syntax in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
  
        do {  
            System.out.print(""inside do"");  
        } while (false);  
  
        while (false) {  
            System.out.print(""inside while"");  
        }  
        System.out.print(""outside"");  
    }  
}
```

Please choose only one answer:

- inside dooutside
- outside
- compilation error because of the second loop (while statement)

Check the answer of this question online on JavaChamp.com: [do while loop syntax in java](#)

1.4.4. Java switch control example

Author: Abhishek Sawant

What is the output of this code?

```
public class SwitchTest{
enum Color {RED, GREEN, BLUE};
public static void main(String []args){
    Color clr = Color.GREEN;
    long x = 2;

    switch(x){ // #1
    case 1:
        System.out.println(&quot;1&quot;);
        break;

    case 2:
        System.out.println(&quot;2&quot;);

    case 3:
        System.out.println(&quot;3&quot;);
        break;

    default:
        System.out.println(&quot;default&quot;);
        break;
    }

    switch(clr){ // #2
    case RED:
        System.out.println(&quot;Red&quot;);
        break;

    case GREEN:
        System.out.println(&quot;Green&quot;);
        break;

    case BLUE:
        System.out.println(&quot;Blue&quot;);
        break;
    }
}
```

Please choose only one answer:

- a) 2 Green
- b) 2 3 Green
- c) 2 3
- d) Compilation error at #1
- e) Compilation error at #2
- f) Runtime exception at #2

Check the answer of this question online on JavaChamp.com: [java switch control example](#)

1.4.5. How to write If-statement in java?

Author: Java Champ

Given the following code, which of if-statements declared is correctly written and causes no compilation error?

```
public class Tester {  
    public static void main(String[] args) {  
        int x = 9;  
  
        // stmt 1  
        if (x>0){}  
  
        // stmt 2  
        if (x=9){} else {}  
  
        // stmt 3  
        if (true){  
        } else if (x<0){  
        }  
  
        // stmt 4  
        if (true){  
        } else (x<0){  
        } else {  
        }  
  
        // stmt 5  
        if(true){  
            if (false){  
            }  
        } else {  
        }  
        } else {  
            if (true){  
            }  
        }  
    }  
}
```

Please choose all the answers that apply:

- stmt 1
- stmt 2
- stmt 3
- stmt 4
- stmt 5

1.4.6. How to write do while statement in java?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
  
    boolean flag = false;  
    int x = 0;  
    do {  
        x++;  
    } while (flag!=flag);  
    System.out.println(x);  
  
}
```

Please choose only one answer:

- 0
- 1
- 2
- 3
- Compilation error
- The loop is infinite and will cause the program to break

Check the answer of this question online on JavaChamp.com: [How to write do while statement in java?](#)

1.4.7. How to write java labeled for loop?

Author: Java Champ

What is the expected output?

```
public class LoopTest {
    public static void goLoop(int a) {

        start: for (int i = 1; i < 2; i++) {
            for (int j = 1; j < 2; j++) {
                if (a < 5) {
                    break;
                }
                System.out.print(i + j);

            }
            if (a > 5) {
                break start;
            }

        }

    }

    public static void main(String args[]) {
        LoopTest.goLoop(15);
    }
}
```

Please choose only one answer:

- 2
- No output is produced
- Compile Error
- Runtime exception is thrown
- 24

Check the answer of this question online on JavaChamp.com: [How to write java labeled for loop?](#)

1.4.8. How to use break and continue in java labeled for loop?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class LoopTest {
    public static void goLoop(int a) {

        outside: for (int i = 0; i < 3; i++) {
            inside: for (int j = 1; j < 3; j++) {
                System.out.print(j);
                if (a > 5) {
                    continue outside;
                }
            }
        }

    }

    public static void main(String args[]) {
        LoopTest.goLoop(15);
    }
}
```

Please choose only one answer:

- 121212
- No output is produced
- Compile Error
- Runtime exception is thrown
- 111

Check the answer of this question online on JavaChamp.com: [How to use break and continue in java labeled for loop?](https://www.javachamp.com/question/1.4.8-how-to-use-break-and-continue-in-java-labeled-for-loop/)

1.4.9. How to use java main method argument?

Author: Java Champ

What is the result of running the following program (which was compiled fine) given the command line:

java Tester one two

```
public class Tester {  
    public static void main(String[] args) {  
        if (args.length > 0)  
            for (String str : args)  
                System.out.print(str);  
                System.out.print("&quot;--&quot;");  
    }  
}
```

Please choose only one answer:

- one--two--
- onetwo
- onetwo--
- Tester--one--two--
- Testeronetwo--

Check the answer of this question online on JavaChamp.com: [How to use java main method argument?](#)

1.4.10. How to use break and continue loop control commands in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class LoopTest {
    public static void goLoop(int a) {

        start; for (int i = 1; i < 2; i++) {
            for (int j = 1; j < 2; j++) {
                if (a > 5) {
                    break start;
                }
                System.out.print(i + j);
            }
        }

    }

    public static void main(String args[]) {
        LoopTest.goLoop(15);
    }
}
```

Please choose only one answer:

- 2
- No output is produced
- Compile Error
- Runtime exception is thrown
- 24

Check the answer of this question online on JavaChamp.com: [How to use break and continue loop control commands in java?](https://www.javachamp.com/question/1410/how-to-use-break-and-continue-loop-control-commands-in-java/)

1.4.11. switch in java(core concept)

Author: pankaj gupta

what will be the output of compiling and running this code:

```
class TestByte{
public static void main(String arg[])
{
final Integer a=10;
int b=2;
switch(b)
{
case 1: System.out.println(&quot;1&quot;);
break;
case 2: System.out.println(&quot;2&quot;);
break;
case a: System.out.println(&quot;3&quot;);
break;
}
}
```

Please choose only one answer:

- compiler error
- prints 2
- Runtime exception is generated.
- prints 3

Check the answer of this question online on JavaChamp.com: [java switch statement rules](#)

1.4.12. Java switch statement syntax

Author: Yasser Ibrahim

Which of the following is correct after running this code ?

```
public class Tester {  
  
    public static void main(String[] args) {  
        char code = 'a';  
        for (int i = 1; i <= 10; i++) {  
            switch (i) {  
  
                case 5: {  
                    System.out.println("match");  
                }  
  
                default: {  
                    System.out.println("default");  
                }  
            }  
        }  
    }  
}
```

Please choose all the answers that apply:

- match will be printed 0 time
- match will be printed one time
- default will be printed 10 times
- default will be printed 11 times

Check the answer of this question online on JavaChamp.com: [Java switch statement syntax](#)

1.4.13. How to use continue and break in java loop?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class LoopTest {
    public static void main(String args[]) {
        int a = 5;
        outside: for (int i = 1; i <= 3; i++) {
            inside: for (int j = 1; j <= 3; j++) {
                System.out.print(j);
                if (a++ > 6)
                    break outside;
            }
        }
    }
}
```

Please choose only one answer:

- Compile error
- Runtime exception is thrown
- 1
- 123

Check the answer of this question online on JavaChamp.com: [How to use continue and break in java loop?](#)

1.4.14. Switch with char variable

Author: Pedro Kowalski

Knowing that character 'a' has a numeric value equal to 97, what is the output of the following code

```
public class CaseTest {  
    public static void main(String... s) {  
        char c = 'a';  
  
        switch (c) {  
            case 'a': {  
                System.out.println("&quot;a&quot;");  
                break;  
            }  
  
            case 97: {  
                System.out.println("&quot;(int)a&quot;");  
                break;  
            }  
        }  
    }  
}
```

Please choose only one answer:

- a
- (int)a
- a (int)a
- compilation fails

Check the answer of this question online on JavaChamp.com: [switch with char variable in java pedro kowalski](#)

1.4.15. How to use break and continue in java loop?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class LoopTest {  
    public static void main(String args[]) {  
        int a = 15;  
        outside: for (int i = 0; i < 3; i++) { // line 4  
            inside: System.out.print(i); // line 5  
            for (int j = 1; j < 3; j++) {  
                if (a > 5)  
                    continue inside; // line 8  
                break outside; // line 9  
            }  
        }  
    }  
}
```

Please choose only one answer:

- Compile Error at line 5
- Runtime exception is thrown at line 5
- Compile Error at line 8
- Runtime exception is thrown at line 8
- No output is produced
- Compile Error at line 4
- Compile Error at line 9
- Runtime exception is thrown at line 9

Check the answer of this question online on JavaChamp.com: [How to use break and continue in java loop?](#)

5. Declarations and Access Control

Exam Category Description and Objectives

1.5.1. How to use java static members?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    static int p = test(); //line 1  
  
    static public int test() {  
        System.out.print(p); //line 4  
        return 99;  
    }  
  
    public static void main(String[] args)  
    {  
        System.out.print(p); //line 11  
    }  
}
```

Please choose only one answer:

- 099
- Compilation error at line 1, p must be initialized by a value
- Compilation error at line 4, using uninitialized variable p
- Compilation error at line 11, p must be called using its class by writing Tester.p

Check the answer of this question online on JavaChamp.com: [How to use java static members?](#)

1.5.2. How to access java class from another package?

Author: Java Champ

Consider the following two classes declared and defined in two different packages, what can be added in class B to form what considered a correct access to class A from main() method of class B? (choose three)

```
package subPackage;

public class A { }

package anotherPackage;
// line 1
public class B {
    public static void main(String[] args) {
        // line 2
    }
}
```

Please choose all the answers that apply:

- at line 1 add nothing
at line 2 add : new A();
- at line 1 add: import package.*;
at line 2 add : new subPackage.A();
- at line 1 add: import subPackage.*;
at line 2 add : new A();
- at line 1 add: import subPackage.A;
at line 2 add : new A();
- at line 1 add nothing
at line 2 add : new subPackage.A();

Check the answer of this question online on JavaChamp.com: [How to access java class from another package?](#)

1.5.3. Can java static methods invoke this?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    static int x = 4;  
  
    public Tester() {  
        System.out.print(this.x); // line 4  
        Tester();  
    }  
  
    public static void Tester() { // line 8  
        System.out.print(this.x); // line 9  
    }  
  
    public static void main(String... args) { // line 12  
        new Tester();  
    }  
}
```

Please choose only one answer:

- Compile error at line 4 (static x must be only accessed inside static methods)
- Compile error at line 8 (constructors can't be static)
- Compile error at line 9 (static methods can't invoke this)
- Compile error at line 12 (invalid argument type for method main)
- 44

Check the answer of this question online on JavaChamp.com: [Can java static methods invoke this?](https://www.javachamp.com/question/can-java-static-methods-invoke-this/)

1.5.4. class modifiers in java

Author: [Java Champ](#)

A top level Java class can have the following modifiers:

Please choose all the answers that apply:

- strictfp and final
- abstract and final
- public and abstract
- protected
- private

Check the answer of this question online on JavaChamp.com: [class modifiers in java](#)

1.5.5. Declarations and Access Control

Author:

What is the output ?

```
class InitDemo{
    static int i=demo();
    static{System.out.println(i);}
    InitDemo(){
        System.out.println(""hello 1"");
    }
    public static void main(String... args){
        System.out.println(""Hello 2"");
    }
    static int demo(){
        System.out.println(""Inside Demo"");
        return 10;
    }
}
```

Please choose only one answer:

- Compilation error
- IllegalArgumentException is thrown at runtime.
- Inside Demo
10
Hello 2
- Hello 2
Inside Demo
10

Check the answer of this question online on JavaChamp.com: [declarations and access control Nikhil Kumar](#)

1.5.6. How to access a class from another package in java?

Author: Yasser Ibrahim

You have two packages, trunk1 and trunk2 where class Sheet declared in trunk1 and class Container declared in trunk2, the following code contains a compilation error, where could it be?

```
package trunk1;

public class Sheet {

    public static int pageNumber = 99; // line 1
    Sheet() {} // line 2
}


package trunk2;

import trunk1.Sheet;

public class Container {

    public static void main(String... args) { //line 1
        System.out.print(Sheet.pageNumber); //line 2
        Sheet sheet = new Sheet(); //line 3
    }
}
```

Please choose only one answer:

- In package trunk1 at line 2, constructor Sheet must be marked public like its class
- In package trunk2 at line 1, invalid string argument for method main()
- In package trunk2 at line 2, invalid access to member pageNumber
- In package trunk2 at line 3, invalid call to constructor Sheet()

Check the answer of this question online on JavaChamp.com: [How to access a class from another package in java?](#)

1.5.7. types of relationships between java classes

Author: Java Champ

Given the following declarations, what is considered a correct statement? (choose four)

```
interface Chewable {}  
interface Eatable extends Chewable{}  
class Vitamin{}  
class Food implements Eatable { Vitamin[] vitamins;}
```

Please choose all the answers that apply:

- Eatable is Chewable
- Vitamin is Eatable
- Food is Chewable
- Food has Vitamin
- Vitamin has Food
- Food is an Object

Check the answer of this question online on JavaChamp.com: [types of relationships between java classes](#)

1.5.8. Inner class modifiers in java

Author: [Java Champ](#)

An inner class in java can have the following modifiers (at the same time):

Please choose all the answers that apply:

- public and abstract
- abstract and final
- protected
- private

Check the answer of this question online on [JavaChamp.com](#): [inner class modifiers in java](#)

1.5.9. strictfp in java

Author: [Java Champ](#)

What does it mean to mark a class strictfp?

Please choose only one answer:

- it means this class can't be subclassed
- it means this class can never be instantiated and it is to be extended
- it means that any method code in the class will conform to the IEEE 754 standard rules for floating points

Check the answer of this question online on JavaChamp.com: [strictfp in java](#)

1.5.10. Can java constructor be private?

Author: Java Champ

What is the result of compiling and running the following code?

```
class Base {  
    private Base() {System.out.print("Base");}  
}  
  
public class Derived extends Base {  
    public Derived() {System.out.print("Derived");}  
  
    public static void main(String[] args) {  
        new Derived();  
    }  
}
```

Please choose only one answer:

- BaseDerived
- Derived
- Exception is thrown at runtime
- Compilation error

Check the answer of this question online on JavaChamp.com: [Can java constructor be private?](#)

1.5.11. what does final do in java?

Author: Yasser Ibrahim

Which of the following is true?

Please choose all the answers that apply:

- When a class marked final, it cannot be extended
- When a method marked final, it cannot be overridden
- When a method marked final, it cannot be overloaded
- Local variables of a method cannot be final
- A variable of a primitive type (int,float,..) which marked final, its value at initialization cannot be changed

Check the answer of this question online on JavaChamp.com: [what does final do in java?](https://www.javachamp.com/question/what-does-final-do-in-java/)

1.5.12. packages and access scope in java

Author: [Java Champ](#)

You have three packages, trunk1, trunk2 and trunk3, What is the expected output of compiling and running file Tester?

```
package trunk1;

public class Account {

    Account() { //line 1
        System.out.println(""Account"");
    }
}

package trunk2;

import trunk1.Account;

public class CurrentAccount extends Account{
    public CurrentAccount() { // line 2
        System.out.println(""CurrentAccount"");
    }
}

package trunk3;

import trunk1.Account;
import trunk2.CurrentAccount; // line 3

public class Tester {

    public static void main(String[] args) {
        Account c = new Account(); // line 4
        CurrentAccount ca = new CurrentAccount();
    }
}
```

Please choose all the answers that apply:

- Account
CurrentAccount
- Account
Account
CurrentAccount
- Compilation error in package trunk1 at line 1

- Compilation error, in package trunk2 at line 2
- Compilation error, in package trunk3 at line 3
- Compilation error, in package trunk3 at line 4

Check the answer of this question online on JavaChamp.com: [packages and access scope in java](#)

1.5.13. What can java static methods make references to?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static int x = 4;  
    int y = 9;  
  
    public Tester() {  
        System.out.print(this.x); // line 7  
        printVariables();  
    }  
  
    public static void printVariables() {  
        System.out.print(x); // line 12  
        System.out.print(y); // line 13  
    }  
  
    public static void main(String... args) { // line 16  
        new Tester();  
    }  
}
```

Please choose only one answer:

- Compile error at line 7 (static x must be only accessed inside static methods)
- Compile error at line 13 (static methods cant make reference to non-static variables)
- Compile error at line 16 (invalid argument type for method main)
- 49
- Compile error at line 12 (must access x by writing Tester.x)

Check the answer of this question online on JavaChamp.com: [What can java static methods make references to?](#)

1.5.14. strictfp modifier in java

Author: Yasser Ibrahim

What can be marked with strictfp modifier? (choose three)

Please choose all the answers that apply:

- an interface
- a class
- a member method
- a member field
- a local variable

Check the answer of this question online on JavaChamp.com: [strictfp modifier in java](#)

6. Interfaces and Abstract Classes

Exam Category Description and Objectives

1.6.1. How to declare java abstract class?

Author: Java Champ

Which of the following declaration will compile without errors? (choose two)

Please choose all the answers that apply:

- `public abstract class Digit { public abstract void print(); }`
- `public class Digit { public abstract void print(); }`
- `public abstract class Digit { public abstract void print(){} }`
- `public abstract class Digit { public void print();}`
- `public class Digit { public void print(){};}`

Check the answer of this question online on JavaChamp.com: [How to declare java abstract class?](#)

1.6.2. What are the rules to implement an interface?

Author: Yasser Ibrahim

Considering the following declaration for interface Convertable, which of the following code segments will compile?

```
public interface Convertable {  
  
    int convertToInt();  
    char convertToChar();  
  
}
```

Please choose all the answers that apply:

- class Digit implements Convertable {
 public char convertToChar() {
 return 0;
 }
 public int convertToInt() {
 return 0;
 }
}
- abstract class Digit implements Convertable {
 int convertToInt() ;
 char convertToChar();
}
- abstract class Digit implements Convertable {

 public int convertToInt() {
 return 0;
 }
}
- abstract class Digit implements Convertable {
 public int convertToInt() {
 return 0;
 }
 char convertToChar();
}
- class Digit implements Convertable {
 int convertToInt() {
 return 0;
 }

```
}
```

```
char convertToChar() {  
    return 0;  
}  
}
```

- interface Roundable extends Convertable {
 int roundUp();
}

Check the answer of this question online on JavaChamp.com: [What are the rules to implement an interface?](#)

1.6.3. How to declare a class inside an interface in java?

Author: Yasser Ibrahim

Is the following declaration for interface Bendable correct and free of compilation error?

```
abstract interface Bendable { // line 1  
  
    final int x = 2009; // line 3  
  
    void method1() ; // line 5  
    public static class Angle {} // line 6  
}
```

Please choose only one answer:

- Yes, this is a correct and free of error declaration
- No, compilation error at line 1 , abstract should be removed
- No, compilation error at line 3 , x should be declared public final
- No, compilation error at line 5 , method method1() should be declared public abstract
- No, compilation error at line 6 , can't declare a class inside an interface

Check the answer of this question online on JavaChamp.com: [How to declare a class inside an interface in java?](#)

1.6.4. Extending java abstract class

Author: Yasser Ibrahim

The following code contains a compilation error , what can be done to fix this error - independently?

```
abstract class AirPlane { // line 1
    abstract void fly(); // line 2

    void land() {
        System.out.print("Landing..");
    }
}

class AirJet extends AirPlane { // line 10

    AirJet() {
        super(); // line 13
    }

    void fly() {
        System.out.print("Flying..");
    }

    abstract void land() ; // line 20
}
```

Please choose all the answers that apply:

- Remove abstract from line 20 and add body to method land()
- Declare class AirJet as abstract to at line 10
- Remove super() call at line 13
- Remove abstract at line 1 and line 2

Check the answer of this question online on JavaChamp.com: [Extending java abstract class](#)

1.6.5. How to declare variables inside java interface?

Author: Java Champ

Which of the following variables is incorrectly declared?

```
public abstract interface Bouncable {  
  
    int a = 0;  
    public int b = 1;  
    public static int c = 2;  
    public static transient int d = 3;  
    public final int e = 3;  
    public static final int f = 3;  
  
}
```

Please choose only one answer:

- a
- b
- c
- d
- e
- f

Check the answer of this question online on JavaChamp.com: [How to declare variables inside java interface?](#)

1.6.6. How to declare java abstract method?

Author: Java Champ

Three of the methods are incorrectly declared, which are they?

```
public abstract class Tester {  
    public void test1();  
    public final void test2() {};  
    public static void test3() {};  
    public abstract static void test4();  
    public abstract final void test5();  
}
```

Please choose only one answer:

- test1 , test2 and test4
- test2 , test4 and test5
- test1 , test4 and test5

Check the answer of this question online on JavaChamp.com: [How to declare java abstract method?](#)

1.6.7. how to implement an interface in java?

Author: Yasser Ibrahim

To resolve the compilation error(s) in the following code, what can be done independently (choose two) :

```
interface Movable {  
    public abstract void m1(); // line 1  
    void m2(); // line 2  
    public void m3(); // line 3  
    abstract void m4(); // line 4  
}  
  
class Chair implements Movable { // line 5  
    public void m1() { } // line 6  
    void m2() { } // line 7  
}
```

Please choose all the answers that apply:

- mark class Chair "abstract";
- mark Chair "abstract"; and mark m2() "public";
- implement m3() and m4() in Chair (with public access modifier)
- implement the methods m3() and m4() in Chair (with public access modifier) and mark m2() in Chair "public";

Check the answer of this question online on JavaChamp.com: [how to implement an interface in java?](https://www.javachamp.com/question/how-to-implement-an-interface-in-java/)

1.6.8. What are java interface legal declaration rules?

Author: [Java Champ](#)

Is the following declaration for interface Bendable correct and free of compilation error?

```
abstract interface Bendable { // line 1
    final int x = 2009; // line 3
    void method1(){}; // line 5
}
```

Please choose only one answer:

- Yes, this is a correct and free of error declaration
- No, compilation error at line 1
- No, compilation error at line 3
- No, compilation error at line 5

Check the answer of this question online on JavaChamp.com: [What are java interface legal declaration rules?](#)

1.6.9. How to declare java interface?

Author: Yasser Ibrahim

Is the following declaration for interface Bendable correct and free of compilation error?

```
abstract interface Bendable { // line 1  
    final int x = 2009; // line 3  
    void method1(); // line 5  
}
```

Please choose only one answer:

- Yes, this is a correct and free of error declaration
- No, compilation error at line 1, Bendable should be declared public abstract
- No, compilation error at line 3 , x should be declared public final
- No, compilation error at line 5 , method method1() should be declared public abstract

Check the answer of this question online on JavaChamp.com: [How to declare java interface?](#)

1.6.10. How to extend java abstract class?

Author: Yasser Ibrahim

Will the following code compile correctly?

```
abstract class AirPlane {  
    abstract void fly();  
  
    void land() { // line 5  
        System.out.print("&quot;Landing..&quot;");  
    }  
}  
  
class AirJet extends AirPlane {  
    AirJet() {  
        super(); // line 14  
    }  
  
    void fly() {  
        System.out.print("&quot;Flying..&quot;");  
    }  
}
```

Please choose only one answer:

- Yes, it will compile with no errors
- No, because at line 5 method land() must be abstract since class AirPlane is abstract
- No, because class AirJet must override method land()
- No, because at line 14 AirJet constructor is calling the super() while AirPlane has no constructor defined

Check the answer of this question online on JavaChamp.com: [How to extend java abstract class?](#)

1.6.11. Rules of declaring java abstract Class

Author: Yasser Ibrahim

Which of the following is true?

Please choose all the answers that apply:

- If a class is declared abstract , it must contain at least one abstract method
- If a class is declared abstract , all its methods must be abstract
- A method can either be final or abstract
- If a method is abstract then its class must be declared abstract
- An Abstract method has no body and ends with a semicolon

Check the answer of this question online on JavaChamp.com: [Rules of declaring java abstract Class](#)

1.6.12. how to write an interface?

Author: Yasser Ibrahim

Which of the following is correct about an interface in java?

Please choose all the answers that apply:

- An Interface can declare data members but must be constants
- All methods in an interface are implicitly abstract
- Methods in an interface may be public, private, protected.
- An interface can implement an abstract class
- An interface can be final

Check the answer of this question online on JavaChamp.com: [how to write an interface?](#)

7. Exceptions

Exam Category Description and Objectives

1.7.1. how to handle exceptions in java?

Author: Java Champ

The following code contains compilation errors, what of the following possible solutions (made together) can be done to remove these errors, putting in mind that all necessary imports will be added automatically? (choose two)

```
import java.io.File;
import java.text.DateFormat;
import java.util.Date;

public class Tester {

    static void call() { // line 3
        File file = new File("&quot;javachamp.dat&quot;"); //line 4
        file.createNewFile(); // line 5
        throw new IllegalArgumentException(); // line 6
    }

    public static void main(String[] args) {

        call(); // line 11
        DateFormat df = DateFormat.getDateInstance(); // line 12
        Date parse = df.parse("&quot;12.11.2009&quot;"); // line 13
    }
}
```

Please choose all the answers that apply:

- Add throws IOException declaration to call() at line 3
- Add throws FileNotFoundException declaration to call() at line 3
- Add throws IllegalArgumentException declaration to call() at line 3
- Surround code from line 11 to 13 with try { } catch (IOException e) {} block
- Surround code from line 11 to 13 with try { } catch (Exception e) {} block

Check the answer of this question online on JavaChamp.com: [how to handle exceptions in java?](#)

1.7.2. try catch finally block in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
  
        System.out.print(""1"");  
        try {  
            return;  
        } catch (Exception e) {  
            System.out.print(""2"");  
        } finally {  
            System.out.print(""3"");  
        }  
        System.out.print(""4"");  
    }  
}
```

Please choose only one answer:

- 1234
- 13
- 1
- Compilation error

Check the answer of this question online on JavaChamp.com: [try catch finally block in java](#)

1.7.3. Can java constructors throw exceptions?

Author: Java Champ

What is the expected output of compiling and running the following code?

```
import java.io.IOException;

class AirPlane {
    public AirPlane() throws IOException, RuntimeException {
        System.out.println(""AirPlane"");
    }
}

class AirJet extends AirPlane { } // line 7

public class Tester {
    public static void main(String args[]) throws IOException { // line 10
        new AirPlane(); // line 11
    }
}
```

Please choose only one answer:

- AirPlane
- Compile error at line 7, AirJet must declare a constructor that throws IOException or any of its supertypes
- Compile error at line 10, main() must throw also RuntimeException
- Compile error at line 11 ,new AirPlane() must be within try/catch block

Check the answer of this question online on JavaChamp.com: [Can java constructors throw exceptions?](#)

1.7.4. How to handle java unchecked exceptions?

Author: Yasser Ibrahim

What is the expected output of compiling and running this code?

```
class Father {
    public Father() throws RuntimeException {
        System.out.print("&quot;Father&quot;");
        throw new RuntimeException();
    }
}

class Son extends Father {
    public Son() throws RuntimeException {
        System.out.print("&quot;Son&quot;");
    }
}

public class Tester {
    public static void main(String[] args) {
        new Son(); // line 17
    }
}
```

Please choose only one answer:

- Compile error, an import to java.lang.RuntimeException is required
- Father will be printed then a RuntimeException would be thrown
- Compile error at line 17, new Son() should be within a try/catch block
- Son
- SonFather
- FatherSon

Check the answer of this question online on JavaChamp.com: [How to handle java unchecked exceptions?](https://www.java-champ.com/question/unchecked-exceptions/how-to-handle-java-unchecked-exceptions/)

1.7.5. How to write nested try catch in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    static void method(){  
        throw new Exception();  
    }  
  
    public static void main(String[] args) {  
        try {  
            method();  
        } catch (Throwable e) {  
            try {  
                throw new Exception() ;  
            } catch (Exception ex) {  
                System.out.print("exception");  
            } finally {  
                System.out.print("finally");  
            }  
        }  
    }  
}
```

Please choose only one answer:

- exception
- finally
- exceptionfinally
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to write nested try catch in java?](https://www.javachamp.com/question/175/how-to-write-nested-try-catch-in-java/)

1.7.6. Catching java exceptions

Author: Java Champ

What is the expected output?

```
import java.io.IOException;

class Father {
    public Father() throws IOException {
        System.out.print("Father");
        throw new IOException();
    }
}

class Son extends Father {
    public Son() throws IOException {
        System.out.print("Son");
    }
}

public class Tester {
    public static void main(String[] args) {
        try {
            new Son();
        } catch (IOException e) {
            System.out.print("Inside catch");
        }
    }
}
```

Please choose only one answer:

- Compile error, constructors can't throw Exceptions
- FatherSon
- FatherInside catch
- Father, then an IOException is thrown
- Son
- SonInside catch

Check the answer of this question online on JavaChamp.com: [Catching java exceptions](#)

1.7.7. How to catch java exceptions?

Author: Yasser Ibrahim

What is the expected result of compiling and running the following code?

```
import java.io.IOException;

class AirPlane {
    public AirPlane() {
        System.out.print(""AirPlane"");
    }
}

class AirJet extends AirPlane {
    public AirJet() throws IOException {
        try {
            throw new IOException();
        } catch (IOException e) {
            System.out.print(""IOException is thrown in AirJet"");
        }
    }
}

public class Tester {
    public static void main(String args[]) {
        try {
            new AirJet();
        } catch (IOException e) {
            System.out.print(""IOException is thrown in Tester"");
        }
    }
}
```

Please choose only one answer:

- ""AirPlaneIOException is thrown in AirJetIOException is thrown in Tester" will be printed
- ""AirPlaneIOException is thrown in AirJet" will be printed
- ""AirPlaneIOException is thrown in Tester" will be printed
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to catch java exceptions?](https://www.javachamp.com/question/177/how-to-catch-java-exceptions/)

1.7.8. When java NoClassDefFoundError is thrown?

Author: [Java Champ](#)

In which case a NoClassDefFoundError will be thrown?

Please choose only one answer:

- When try to compile a non found java file
- When try to run a non found java .class file
- When try to compile a non found java file or try to run a non found java .class file

Check the answer of this question online on [JavaChamp.com](#): [When java NoClassDefFoundError is thrown?](#)

1.7.9. How to use a try catch in java?

Author: Yasser Ibrahim

What is the expected output?

```
interface Foldable {  
    public void fold() throws Exception ;  
}  
  
class Paper implements Foldable {  
    public void fold() { // line 6  
        System.out.print("Fold");  
    }  
}  
  
public class Tester {  
    public static void main(String args []) {  
        Foldable obj1 = new Paper();  
        obj1.fold(); // line 8  
        Paper obj2 = new Paper(); // line 10  
        obj2.fold();  
    }  
}
```

Please choose only one answer:

- FoldFold
- Compilation error at line 6, fold() must declare at header throws Exception
- Compilation error at line 8, unhandled exception
- RuntimeException at line 8
- Compilation error at line 10, unhandled exception
- RuntimeException at line 10

Check the answer of this question online on JavaChamp.com: [How to use a try catch in java?](#)

1.7.10. How to write try catch statement in java?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        String stmt = "javachamp 2009";  
        String[] arr = stmt.split(" ");  
        try {  
            int x = Integer.parseInt(arr[0]);  
            System.out.print(x);  
        } catch (Exception e) {  
            System.out.print("catch");  
        } finally {  
            System.out.print("finally");  
        }  
    }  
}
```

Please choose only one answer:

- 2009
- 2009finally
- catchfinally
- No output will be produced
- Compilation error
- 2009catchfinally

Check the answer of this question online on JavaChamp.com: [How to write try catch statement in java?](#)

1.7.11. How to declare try catch finally block in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
  
        System.out.print("&quot;1&quot;");  
        try {  
            System.out.print("&quot;2&quot;");  
            System.exit(0);  
        } finally {  
            System.out.print("&quot;3&quot;");  
        }  
    }  
}
```

Please choose only one answer:

- 123
- 12
- Compilation error , there should be at least one catch before finally

Check the answer of this question online on JavaChamp.com: [How to declare try catch finally block in java?](#)

1.7.12. How to write java nested try catch?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static void method() throws Exception {  
        throw new Exception();  
    }  
  
    public static void main(String[] args) {  
  
        try {  
            method();  
        } catch (Throwable e) {  
            try {  
                throw new Exception() ;  
            } catch (Exception ex) {  
                System.out.print("exception");  
            } finally {  
                System.out.print("finally");  
            }  
        }  
    }  
}
```

Please choose only one answer:

- "exception" is printed
- "finally" is printed
- "exceptionfinally" is printed
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to write java nested try catch?](https://www.javachamp.com/question/1712/how-to-write-java-nested-try-catch/)

1.7.13. How to handle exceptions using try/catch block?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        String stmt = "javachamp 2009";  
        String[] arr = stmt.split(" ");  
        try {  
            int x = Integer.parseInt(arr[1]);  
            System.out.print(x);  
        } finally {  
            System.out.print("finally");  
        }  
    }  
}
```

Please choose only one answer:

- 2009
- finally
- 2009finally
- No output will be produced
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to handle exceptions using try/catch block?](#)

1.7.14. How does division by zero with doubles work

Author: Pedro Kowalski

What is the output of the following code

```
public class Tester {  
    public static void main(String[] args) {  
        Double d = -4.0;  
  
        try {  
            d /= 0;  
        } catch (ArithmeticException e) {  
            System.out.println("&quot;EXCEPTION!&quot;");  
        } finally {  
            System.out.println(d);  
        }  
    }  
}
```

Please choose only one answer:

- NotANumber
- NaN
- EXCEPTION! -4.0
- -Infinity
- EXCEPTION! 0.0
- Compilation fails

Check the answer of this question online on JavaChamp.com: [how does division by zero with doubles work pedro kowalski](https://www.javachamp.com/question/how-does-division-by-zero-with-doubles-work-pedro-kowalski)

1.7.15. How to write java try catch statement?

Author: Java Champ

What is the result of compiling and running the following program?

```
public class Tester {  
    public static void main(String[] args) {  
        try {  
            throw new RuntimeException();  
        } catch (RuntimeException e) {  
            System.out.println("RuntimeException");  
        } catch (ArithmeticException e) {  
            System.out.println("ArithmeticException");  
        } catch (Exception e) {  
            System.out.println("Exception");  
        }  
    }  
}
```

Please choose only one answer:

- "RuntimeException" is printed
- "Exception" is printed
- "ArithmeticException" is printed
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to write java try catch statement?](#)

8. Inheritance

Exam Category Description and Objectives

1.8.1. How to implement or extend in java?

Author: Java Champ

Which of the following is true?

Please choose all the answers that apply:

- A class can extend more than one class
- A class can extend only one class but many interfaces
- An interface can extend many interfaces
- An interface can implement many interface
- A class can extend one class and implement many interfaces

Check the answer of this question online on JavaChamp.com: [How to implement or extend in java?](#)

1.8.2. Casting rules in java

Author: Yasser Ibrahim

What inserted, independently at // insert code here , will compile?

```
class Creature {
    void grow() {
    }
}

class Bird extends Creature {
    void fly() {
    }
}

class Falcon extends Bird {
    void hunt() {
    }
}

public class Tester {

    public static void main(String[] args) {
        Creature c1 = new Bird();
        Falcon c2 = new Falcon();

        // insert code here

    }
}
```

Please choose all the answers that apply:

- c1.grow();
- c1.fly();
- ((Bird)c1).fly();
- ((Falcon)c1).hunt();

Check the answer of this question online on JavaChamp.com: [Casting rules in java](#)

1.8.3. How to extend or implement in java?

Author: [Java Champ](#)

Are the following declarations correct, if you knew that they are in the same package?

```
interface Bouncable{ } // line 1
interface Colorable extends Bouncable{ } // line 2
class Super implements Colorable{ } // line 3
class Sub implements Colorable extends Super { } // line 4
```

Please choose only one answer:

- Yes, the declarations are correct
- No there is a compilation error at line 2, must implements not extends
- No there is a compilation error at line 4, over implementing for interface Colorable, Super is already implementing it
- No there is a compilation error at line 4, extends should come before implements

Check the answer of this question online on [JavaChamp.com](#): [How to extend or implement in java?](#)

1.8.4. When java classCastException is thrown?

Author: Yasser Ibrahim

Which statement(s), inserted independently at // insert code here, will compile?

```
class Creature {
    String getName() {
        return "Creature";
    }
}

class Bird extends Creature {
    String getName() {
        return "Bird";
    }
}

class Falcon extends Bird {
    String getName() {
        return "Falcon";
    }
}

public class Tester {

    public static Bird getIt(Creature c) {
        System.out.println(c.getName());
        return (Bird) c;
    }

    public static void main(String[] args) {
        // insert code here
    }
}
```

Please choose all the answers that apply:

- getIt(new Creature());
- getIt(new Bird());
- getIt(new Falcon());
- getIt(new Object());

Check the answer of this question online on JavaChamp.com: [When java classCastException is thrown?](#)

1.8.5. Inheritance in Java

Author:

What is the output ?

```
class A{
    A(){
        System.out.println(""Hello"");
    }
}
class InitDemo extends A{
    A ob=new A();
    InitDemo(){
        System.out.println(""hello 1"");
    }
    public static void main(String... args){
        System.out.println(""Hello 2"");
        new InitDemo();
    }
}
```

Please choose only one answer:

- Hello 2
hello 1
Hello
Hello
- Hello 2
Hello
Hello
hello 1
- Hello 2
Hello
hello 1
Hello
- Exception is thrown at runtime.

Check the answer of this question online on JavaChamp.com: [inheritance in java Nikhil Kumar](#)

1.8.6. How to java cast?

Author: Java Champ

What inserted, independently at // insert code here , will compile and run with no errors and exceptions?

```
class Creature {  
    void grow() {  
    }  
}  
  
class Bird extends Creature {  
    void fly() {  
    }  
}  
  
class Falcon extends Bird {  
    void hunt() {  
    }  
}  
  
public class Tester {  
  
    public static void main(String[] args) {  
        Creature c1 = new Bird();  
        Falcon c2 = new Falcon();  
  
        // insert code here  
    }  
}
```

Please choose all the answers that apply:

- c1.grow();
- c1.fly();
- ((Bird)c1).fly();
- ((Falcon)c1).hunt();

Check the answer of this question online on JavaChamp.com: [How to java cast?](#)

1.8.7. How to call this inside java constructor?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
class Base {  
    public Base() {System.out.print("Base");}  
}  
  
public class Derived extends Base {  
    public Derived() {this("JavaChamp"); System.out.print("Derived");}  
    public Derived(String s) { System.out.print(s);}  
  
    public static void main(String[] args) {  
        new Derived();  
    }  
}
```

Please choose only one answer:

- JavaChampDerived
- JavaChampBaseDerived
- BaseJavaChampDerived
- JavaChampDerivedBase
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to call this inside java constructor?](#)

1.8.8. using instanceof in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
interface Chewable {}
interface Eatable extends Chewable{}
class Food implements Chewable { }
class Meat extends Food {}
class Gum implements Chewable{}

public class Tester {

    public static void main(String[] args) {

        Food food = new Food();
        Meat meat = new Meat();
        Gum gum  = new Gum();

        System.out.print(food instanceof Eatable);
        System.out.print(meat instanceof Food);
        System.out.print(gum instanceof Food);
        System.out.print(gum instanceof Chewable);

    }
}
```

Please choose only one answer:

- true true true true
- true true false true
- false true false true
- Compilation error

Check the answer of this question online on JavaChamp.com: [using instanceof in java](#)

1.8.9. null and instanceof in java

Author: [Java Champ](#)

considering AnyClassOrInterface is a class or an interface in java
what would the following statement print?

```
System.out.print(null instanceof AnyClassOrInterface );
```

Please choose only one answer:

- true
- false
- compilation error
- NullPointerException is thrown at runtime

Check the answer of this question online on [JavaChamp.com](#): [null and instanceof in java](#)

1.8.10. how to implement interface or extend class in java?

Author: Java Champ

Which of the following statements are correct? (choose two)

Please choose all the answers that apply:

- A class can extend one class and many interfaces
- A class can implement one class and many interfaces
- A class can extend one class and implements many interfaces
- An interface can extend many classes and interfaces
- An interface can implement many classes and interfaces
- An interface can implement many interfaces
- An interface can extend many interfaces

Check the answer of this question online on JavaChamp.com: [how to implement interface or extend class in java?](#)

1.8.11. How to cast in java?

Author: Java Champ

The following program fails to compile, where could possibly be the compilation error(s)?

```
class Creature {}  
class Bird extends Creature {}  
class Falcon extends Bird {}  
public class Tester {  
    public static void main(String[] args) {  
        Creature c1 = new Creature();  
        Creature c2 = new Bird();  
  
        Bird b1 = (Bird) c1; // Line 1  
        Bird b2 = (Falcon) c2; // Line 2  
        Bird b3 = c2; // Line 3  
        Bird b4 = new Falcon(); // Line 4  
        Bird b5 = (Bird) new Creature(); // Line 5  
        Falcon f1 = b4; // Line 6  
    }  
}
```

Please choose all the answers that apply:

- Line 1
- Line 2
- Line 3
- Line 4
- Line 5
- Line 6

Check the answer of this question online on JavaChamp.com: [How to cast in java?](#)

1.8.12. How to cast java variables?

Author: Yasser Ibrahim

Will the following code compile and if yes , what is the output?

```
interface Colorable {}

class Vehicle {}

class Car extends Vehicle implements Colorable {}

public class Tester {

    public static void main(String[] args) {

        Vehicle a = new Car();
        Colorable i = (Colorable) a;

        Vehicle b = new Vehicle();
        Colorable j = (Colorable) b;
    }
}
```

Please choose only one answer:

- The code compiles fine and produces no output
- The code compiles fine but throws a RuntimeException because of the second casting
- There is a compilation error because of the second casting
- The code compiles fine but throws a RuntimeException because of the first casting

Check the answer of this question online on JavaChamp.com: [How to cast java variables?](https://www.javachamp.com/question/1.8.12-how-to-cast-java-variables/)

1.8.13. how to use instanceof java?

Author: Java Champ

what is the result of compiling and running the following code?

```
interface Colorable { }

interface Bouncable extends Colorable { }

class Super implements Bouncable { }

class Sub extends Super implements Bouncable { }

class Individual { }

public class Tester {

    public static void main(String[] args) {

        System.out.print(new Sub() instanceof Super);
        System.out.print(new Sub() instanceof Colorable);
        System.out.print(new Super() instanceof Sub);
        System.out.print(new Individual() instanceof Super);
    }
}
```

Please choose only one answer:

- true>true>false>false
- true>false>false>false
- Compilation error
- RuntimeException because of the print statemen

Check the answer of this question online on JavaChamp.com: [how to use instanceof java?](#)

1.8.14. How to write constructor for derived class in java?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
class Base {  
    public void Base() {System.out.print("Base");}  
}  
  
public class Derived extends Base {  
    public Derived() {System.out.print("Derived");}  
  
    public static void main(String[] args) {  
        new Derived();  
    }  
}
```

Please choose only one answer:

- BaseDerived
- DerivedBase
- Derived
- Base
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to write constructor for derived class in java?](#)

1.8.15. How to use instanceof in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
interface Chewable {}
interface Eatable extends Chewable{}
class Food implements Eatable { }
class Meat extends Food {}
class Gum implements Chewable{}

public class Tester {

    public static void main(String[] args) {

        Food food = new Food();
        Meat meat = new Meat();
        Gum gum  = new Gum();

        System.out.print(food instanceof Chewable);
        System.out.print(meat instanceof Eatable);
        System.out.print(gum instanceof Eatable);
        System.out.print(gum instanceof Chewable);

    }
}
```

Please choose only one answer:

- false true false true
- true true false true
- true true true true
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to use instanceof in java?](#)

1.8.16. Inheritance and constructors in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
class Base {  
    public Base(String n) {System.out.print("Base1");}  
    public void Base(String n) {System.out.print("Base2");}  
}  
  
public class Derived extends Base {  
    public Derived() {System.out.print("Derived");}  
    public static void main(String[] args) {  
        new Derived();  
    }  
}
```

Please choose only one answer:

- Base1Derived
- Base2Derived
- Derived
- Compilation error

Check the answer of this question online on JavaChamp.com: [Inheritance and constructors in java](#)

1.8.17. Order of constructor calls of java derived classes

Author: Yasser Ibrahim

What is the output of compiling and running the following program?

```
class Category {
    Category() {
        System.out.print("Category_");
    }
}

class SubCategory extends Category {
    SubCategory() {
        System.out.print("SubCategory_");
    }
}

class SubSubCategory extends SubCategory {
    SubSubCategory() {
        System.out.print("SubSubCategory_");
    }
}

public class Tester {

    public static void main(String[] args) {
        new SubSubCategory();
    }
}
```

Please choose only one answer:

- SubSubCategory_SubCategory_Category_
- SubSubCategory_
- Category_SubCategory_SubSubCategory_

Check the answer of this question online on JavaChamp.com: [Order of constructor calls of java derived classes](#)

1.8.18. How to use java instanceof?

Author: Yasser Ibrahim

Which lines will evaluate to true?

```
interface Colorable { }

interface Bouncable extends Colorable { }

class Super implements Bouncable { }

class Sub extends Super implements Bouncable { }

public class Tester {

    public static void main(String[] args) {

        System.out.println(new Sub() instanceof Super); //line 1
        System.out.println(new Sub() instanceof Bouncable); //line 2
        System.out.println(new Sub() instanceof Colorable); //line 3
        System.out.println(new Super() instanceof Sub); //line 4
        System.out.println(new Super() instanceof Colorable); //line 5

    }

}
```

Please choose only one answer:

- All lines will evaluate to true
- All lines will evaluate to true and only line 4 will evaluate to false
- Only line 1 and 2 will evaluate to true
- Lines 3 and 5 will evaluate to false

Check the answer of this question online on JavaChamp.com: [How to use java instanceof?](#)

1.8.19. casting objects in java

Author: Yasser Ibrahim

Would the following code compile and if yes , what is the output?

```
interface Colorable {}

class Vehicle {}

class Car extends Vehicle implements Colorable {}

public class Tester {

    public static void main(String[] args) {

        Vehicle a = new Car();
        Colorable i = (Colorable) a;

        Vehicle b = new Vehicle();
        Colorable j = (Colorable) b;
    }
}
```

Please choose only one answer:

- The code compiles fine and produces no output
- The code compiles fine but throws a RuntimeException because of the second casting
- There is a compilation error because of the second casting

Check the answer of this question online on JavaChamp.com: [casting objects in java](#)

9. Arrays

Exam Category Description and Objectives

1.9.1. how to declare java array?

Author: Java Champ

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int[] a = new int[0];  
    System.out.print(a.length);  
}
```

Please choose only one answer:

- 0
- Compilation error, arrays cannot be initialized to zero size
- Compilation error, it is a.length() not a.length

Check the answer of this question online on JavaChamp.com: [how to declare java array?](#)

1.9.2. When ArrayIndexOutOfBoundsException is thrown?

Author: [Java Champ](#)

What is true after running the following code?

```
public static void main(String[] args) {  
    String entries[] = {"entry1","entry2"};  
    int count=0;  
    while (entries [count++]!=null){  
        System.out.println(count);  
    }  
        System.out.println(count);  
}
```

Please choose all the answers that apply:

- An Exception will be thrown
- 0 will be printed as part of the output
- 2 will be printed as part of the output
- 3 will be printed as part of the output

Check the answer of this question online on [JavaChamp.com](#): [When ArrayIndexOutOfBoundsException is thrown?](#)

1.9.3. When java ArrayIndexOutOfBoundsException is thrown?

Author: Java Champ

What is the result of compiling and running the following program?

```
public class Tester {  
    final static int x[] = new int[5];  
  
    public static void main(String[] args) {  
        final int x = new Tester().x[5];  
  
        if (x <= 10)  
            System.out.println("javachamp");  
    }  
}
```

Please choose only one answer:

- Compilation error
- ArrayIndexOutOfBoundsException is thrown
- javachamp
- No output is produced

Check the answer of this question online on JavaChamp.com: [When java ArrayIndexOutOfBoundsException is thrown?](#)

1.9.4. Array declaration in java

Author: Yasser Ibrahim

Which of the following declarations of an array is incorrect?

```
public static void main(String[] args) {  
    int[] a[];  
    int b[3];  
    int []c[];  
    int[] d[];  
}
```

Please choose only one answer:

- a
- b
- c
- d

Check the answer of this question online on JavaChamp.com: [Array declaration in java](#)

1.9.5. array size in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static void test(int[] a) {  
        int[] b = new int[2];  
        a = b;  
        System.out.print(b.length);  
        System.out.print(a.length);  
    }  
  
    public static void main(String[] args) {  
  
        int[] a = new int[5];  
        test(a);  
        System.out.print(a.length);  
  
    }  
}
```

Please choose only one answer:

- 225
- 255
- 200

Check the answer of this question online on JavaChamp.com: [array size in java](#)

10. Operators

Exam Category Description and Objectives

1.10.1. Logical Operators ^ and ! in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    System.out.print(false ^ true);  
    System.out.print(!false);  
}
```

Please choose only one answer:

- falsefalse
- falsetrue
- truefalse
- truetrue

Check the answer of this question online on JavaChamp.com: [Logical Operators ^ and ! in java](#)

1.10.2. Which operands can be used in the comparison operators?

Author: Java Champ

What is the output of the running the following program?

```
class Test {
    public static void main(String arg[]) {
        A b = new B(1000);
        A c = new C(1000);
        System.out.println(b==c);
        System.out.println(b &lt;= c);
    }
}

class A {
    int value;
}

class B extends A {
    B(int val) {
        value = val;
    }
}

class C extends A {
    C(int val) {
        value = val;
    }
}
```

Please choose only one answer:

- "true" then "false" will be printed
- "false" then "false" will be printed
- "false" then "true" will be printed
- compilation error at the first print statement
- compilation error at the second print statement

Check the answer of this question online on JavaChamp.com: [which operands can be used in the comparison operators?](#)

1.10.3. | non-short-circuit OR in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int x = 0, y = 0;  
  
    if ((y == x++) | (x < ++y)) {  
        ++y;  
    }  
    System.out.println(x + " " + y);  
}
```

Please choose only one answer:

- 00
- 10
- 11
- 12
- 22

Check the answer of this question online on JavaChamp.com: [| non-short-circuit OR in java](#)

1.10.4. Java Boolean Logical Operators

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        boolean flag = true;  
        int x = 5;  
        int result = (8 * 2) % x;  
        System.out.print(!(result > 0) ^ true ^ (flag = false));  
        System.out.print(flag);  
    }  
}
```

Please choose only one answer:

- truefalse
- truetrue
- falsefalse
- falsetrue
- Compilation error

Check the answer of this question online on JavaChamp.com: [Java Boolean Logical Operators](#)

1.10.5. How to use java conditional operators?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static boolean isEven(int x) {  
        return (x % 2 == 0) ? true : false;  
    }  
  
    public static void main(String[] args) {  
        System.out.print(isEven(2));  
        System.out.print(isEven(3));  
        System.out.print(isEven(4));  
    }  
}
```

Please choose only one answer:

- truefalsetrue
- falsefalsetrue
- falsefalsefalse
- a compilation error
- an exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [How to use java conditional operators?](#)

1.10.6. short-circuit AND in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int x = 1, y = 0;  
    if ((y == x++) && (x < ++y)) { }  
    System.out.println(x + "" + y);  
}
```

Please choose only one answer:

- 20
- 21
- 10
- 11

Check the answer of this question online on JavaChamp.com: [short-circuit AND in java](#)

1.10.7. How to use java short circuit operators?

Author: Java Champ

Which of the following "if" statement(s) will throw a NullPointerException (assuming each statement will be executed independently)?

```
public class Tester {  
    public int code;  
    public static void main(String[] args) {  
        Tester val = null;  
        if((val!=null) & (val.code >0));//line 1  
        if((val!=null) && (val.code >0));//line 2  
        if((val==null) | (val.code==0));//line 3  
        if((val==null) || (val.code==0));//line 4  
    }  
}
```

Please choose all the answers that apply:

- Line 1
- Line 2
- Line 3
- Line 4

Check the answer of this question online on JavaChamp.com: [How to use java short circuit operators?](#)

1.10.8. short circuit operators in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        int x = 0;  
        int y = 0;  
        if ((true & false) | (true & false) & x++>0)  
            System.out.print("stmt1 ");  
        if (false || true | (++y> 0 | y++>0))  
            System.out.print("stmt2 ");  
        System.out.print(x+ " "+y);  
    }  
}
```

Please choose only one answer:

- stmt1 stmt2 1 2
- stmt1 stmt2 0 0
- stmt2 1 2
- stmt2 0 2
- 0 0

Check the answer of this question online on JavaChamp.com: [short circuit operators in java](#)

1.10.9. Concatenating operands in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        System.out.print("" + 7 + 2 + " ");  
        System.out.print(7 + 2 + " ");  
    }  
}
```

Please choose only one answer:

- 9 9
- 72 9
- 9 72
- 72 72

Check the answer of this question online on JavaChamp.com: [Concatenating operands in java](#)

1.10.10. Multiple assignment with generics

Author: Pedro Kowalski

What is the output of the following code?

```
import java.util.Arrays;
import java.util.List;

public class Tester {
    public static void main(String... xyz) {
        int i = 1;
        Integer[] arr = {0, 1, 2};
        int z = 0;

        arr[i] = z = ++i;

        List<Integer> li = Arrays.asList(arr);

        System.out.println(li + "<#>" + z);
    }
}
```

Please choose only one answer:

- [0, 2, 2]#2
- [0, 1, 2]#2
- [0, 1, 2]#1
- [0, 2, 2]#1
- compilation fails

Check the answer of this question online on JavaChamp.com: [arrays aslist](#)

1.10.11. short circuit logical operators in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        boolean flag = true;  
        int x = 5;  
        int result = (8 * 2) % x;  
        System.out.print(!(result > 0) ^ true ^ !(flag = false));  
        System.out.print(flag);  
    }  
}
```

Please choose only one answer:

- truefalse
- truetrue
- falsefalse
- falsetrue
- Compilation error

Check the answer of this question online on JavaChamp.com: [short circuit logical operators in java](#)

1.10.12. Which operands can be used in the equality comparison operators?

Author: Java Champ

What is the output of compiling and running the following code?

```
class Test {  
    public static void main(String arg[]) {  
        Integer x = 1000;  
        long y = 1000;  
        Integer z = 1000;  
        System.out.println(x==y);  
        System.out.println(x==z);  
    }  
}
```

Please choose only one answer:

- true then false will be printed
- true then true will be printed
- false then true will be printed
- false then false will be printed
- compilation error at the first print statement
- compilation error at the second print statement

Check the answer of this question online on JavaChamp.com: [which operands can be used in the equality comparison operators?](#)

1.10.13. short-circuit OR in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int x = 1, y = 0;  
    if ((y == x++) || (x < ++y)) { }  
    System.out.println(x + "" + y);  
}
```

Please choose only one answer:

- 10
- 11
- 20
- 21

Check the answer of this question online on JavaChamp.com: [short-circuit OR in java](#)

1.10.14. ++ and -- in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int x = 0, y = 0;  
    x = 5 + y++;  
    System.out.print(x + "" + y);  
  
    x = 0;  
    y = 0;  
    x = 5 + ++y;  
    System.out.print(x + "" + y);  
}
```

Please choose only one answer:

- 6161
- 671
- 5161

Check the answer of this question online on JavaChamp.com: [++ and -- in java](#)

1.10.15. if condition in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Test{  
    public static void main(String argv[]) {  
        int a = 6;  
        if (a = 6)  
            System.out.print("true");  
        }  
    }  
}
```

Please choose only one answer:

- no output is produced
- true
- a compilation error

Check the answer of this question online on JavaChamp.com: [if condition in java](#)

1.10.16. Incrementing java static variable

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    static int i ;  
  
    private static int increment(int i) {  
        return i++;  
    }  
  
    public Tester() {  
        i = 1;  
    }  
  
    public static void main(String[] args) {  
        System.out.print(i);  
        System.out.print(i = increment(i));  
    }  
}
```

Please choose only one answer:

- 01
- 00
- 12
- Compilation error

Check the answer of this question online on JavaChamp.com: [Incrementing java static variable](#)

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Test{  
    public static void main(String argv[]) {  
        Foo x = new Foo();  
        Foo y = new Foo();  
        Foo z = x;  
  
        System.out.print(x == y);  
        System.out.print(x == z);  
    }  
}  
  
class Foo {}
```

Please choose only one answer:

- false>true
- false>false
- true>true
- compilation error

Check the answer of this question online on JavaChamp.com: [using == in java](#)

1.10.18. & non-short-circuit AND in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int x = 0, y = 0;  
    if ((y == ++x) & (x < ++y)) {}  
    System.out.println(x + "" + y);  
}
```

Please choose only one answer:

- 10
- 11
- 20
- 21

Check the answer of this question online on JavaChamp.com: [& non-short-circuit AND in java](#)

1.10.19. Multiple assignment

Author: Pedro Kowalski

What is the output of the following code?

```
import java.util.Arrays;
import java.util.List;

public class Tester {
    public static void main(String[] arg) {
        int i = 1;
        int[] arr = {0, 1, 2};
        int z = 0;

        arr[i] = z = ++i;

        List<Integer> li = Arrays.asList(arr);

        System.out.println(li + " " + z);
    }
}
```

Please choose only one answer:

- [0, 2, 2]#2
- [0, 1, 2]#2
- [0, 1, 2]#1
- [0, 2, 2]#1
- [0, 1, 2]#2
- compilation fails

Check the answer of this question online on JavaChamp.com: [how arrays aslist method works](#)

1.10.20. java compound operator

Author: [pankaj gupta](#)

what will be the output when we compile and execute the code below:

```
class Compound{
public static void main(String args[])
{
int k=12;
k/=--k;
System.out.println(k);
k*=k++;
System.out.println(k);
k*=++k;
System.out.println(k);
}
}
```

Please choose only one answer:

- It will compile successfully and display 1 followed by 2 and then 3 as an output.
- It will compile successfully and display 1 followed by 1 and then 2 as an output.
- It will compile successfully and display 1 followed by 3 and then 4 as an output.
- It will generate compile time error.

Check the answer of this question online on JavaChamp.com: [java compound operator](#)

1.10.21. Short circuit vs non shortcircuit operators in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        int x = 0;  
        int i = 1;  
        for (; i > 0; i--)  
            if ((++x > 0) | (--i > 0))  
                x++;  
        System.out.println(x + " " + i);  
    }  
}
```

Please choose only one answer:

- Compilation error
- 21
- 20
- 2-1

Check the answer of this question online on JavaChamp.com: [Short circuit vs non shortcircuit operators in java](#)

1.10.22. String and + operator in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    System.out.print("4" + 5);  
    System.out.print(4 + 5);  
    System.out.print(4 + 5 + "9");  
}
```

Please choose only one answer:

- 4545459
- 459459
- 45999

Check the answer of this question online on JavaChamp.com: [String and + operator in java](#)

11. Variables and Methods

Exam Category Description and Objectives

1.11.1. How to write method main in java?

Author: Java Champ

Would the following code compile and run?

```
public class Tester {  
    public static void main(String args) {  
        System.out.println("JavaChamp");  
    }  
}
```

Please choose only one answer:

- The program will compile and run fine printing JavaChamp as output
- The program will compile fine but won't run correctly, a NoSuchMethodError exception would be thrown
- There is a compilation error at declaring the main() argument, should be an array of String instead

Check the answer of this question online on JavaChamp.com: [How to write method main in java?](#)

1.11.2. how to declare valid java variable?

Author: Java Champ

Which of the following is a valid variable declaration?

Please choose all the answers that apply:

- `int _javaChamp;`
- `int _&%javaChamp;`
- `int &javaChamp;`
- `int 2009JavaChamp;`
- `int $JavaChamp;`

Check the answer of this question online on JavaChamp.com: [how to declare valid java variable?](#)

1.11.3. java class data members

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static {  
        int x = 3;  
    }  
  
    static int x;  
  
    public static void main(String[] args) {  
        x--; // line 7  
        System.out.println(x);  
    }  
}
```

Please choose only one answer:

- 3
- 2
- -1
- Compilation error at line 7, x is not initialized

Check the answer of this question online on JavaChamp.com: [java class data members](#)

1.11.4. How to call java static method?

Author: Yasser Ibrahim

What is the result of compiling and running the following program?

```
public class Tester {  
    public static void print(Integer arg){  
        System.out.print("Integer");  
    }  
  
    public void print(int arg){  
        System.out.print("int");  
    }  
  
    public static void main(String[] args) {  
        print(new Integer(3));  
    }  
}
```

Please choose only one answer:

- int
- Integer
- RuntimeException is thrown at run time
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to call java static method?](#)

1.11.5. Using java variable before initialization

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public static void main(String[] args) {  
    int x = 5, y;  
    while (++x < 7) {  
        y = 2;  
    }  
    System.out.println(x + y);  
}
```

Please choose only one answer:

- 7
- 8
- 9
- a compilation error

Check the answer of this question online on JavaChamp.com: [Using java variable before initialization](#)

1.11.6. How to write a java bean?

Author: [Java Champ](#)

Which of the following methods follow the JavaBean standards? (choose two)

Please choose all the answers that apply:

- `public void addJCListener(JCListener listener)`
- `void setName(String name)`
- `public void removeJCListener(MCListener listener)`
- `public boolean isDeleted()`
- `public void fetchName(String name)`

Check the answer of this question online on [JavaChamp.com](#): [How to write a java bean?](#)

1.11.7. How to use java local variables?

Author: Java Champ

What is the result of compiling and running the following program?

```
public class Tester {  
    public static void main(String[] args) {  
        int x = 1;  
        int y;  
        while(++x<5) y++;  
        System.out.println(y);  
    }  
}
```

Please choose only one answer:

- 0
- 3
- 4
- Compilation error

Check the answer of this question online on JavaChamp.com: [How to use java local variables?](#)

1.11.8. What is java local variables scope?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        if(true){  
            int x = 5;  
            System.out.print(x);  
        } else {  
            ++x;  
            System.out.print(x);  
        }  
    }  
}
```

Please choose only one answer:

- 5
- 56
- 6
- Compilation error

Check the answer of this question online on JavaChamp.com: [What is java local variables scope?](#)

1.11.9. How variables are passed in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    int x = 12;  
  
    static Tester reset( Tester obj) {  
        obj = null;  
        return obj;  
    }  
  
    public static void main(String[] args) {  
        Tester o1 = new Tester();  
        o1 = reset(o1);  
        System.out.print(o1.x);  
    }  
}
```

Please choose only one answer:

- 0
- 12
- Compilation error
- NullPointerException is thrown
- Another type of exception is thrown

Check the answer of this question online on JavaChamp.com: [How variables are passed in java?](#)

1.11.10. How to pass variables in java?

Author: [Java Champ](#)

What is the result of compiling and running the following code?

```
public class Tester {  
    int x = 12;  
  
    static void reset(Tester obj) {  
        obj = null;  
    }  
  
    public static void main(String[] args) {  
        Tester o1 = new Tester();  
        reset(o1);  
        System.out.print(o1.x);  
    }  
}
```

Please choose only one answer:

- 0
- 12
- Compilation error
- NullPointerException is thrown
- Another type of exception is thrown

Check the answer of this question online on [JavaChamp.com](#): [How to pass variables in java?](#)

1.11.11. How to use java final variable?

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    final short s = 12; // Line 3  
  
    static void swap( Tester obj) {  
        obj.s = 11; // Line 6  
    }  
  
    public static void main(String[] args) {  
        final Tester obj1 = new Tester();  
        swap(obj1);  
        System.out.print(obj1.s); // Line 12  
    }  
}
```

Please choose only one answer:

- 11
- 12
- Compilation error at line 3
- Compilation error at line 6
- Compilation error at line 12

Check the answer of this question online on JavaChamp.com: [How to use java final variable?](#)

1.11.12. Java local variables initialization

Author: Yasser Ibrahim

What is the result of compiling and running the following program?

```
public class Tester {  
    public static void main(String[] args) {  
        int x = 1;  
        int y=0;  
        while(++x<5) y++;  
        System.out.println(y);  
    }  
}
```

Please choose only one answer:

- 0
- 3
- 4
- Compilation error

Check the answer of this question online on JavaChamp.com: [Java local variables initialization](#)

12. Wrappers and Autoboxing

Exam Category Description and Objectives

1.12.1. Boxing and Varargs issue

Author:

What is the output ?

```
class BoxVar{
    static void call(Integer... i){
        System.out.println(""hi""+i);
    }
    static void call(int... i ){
        System.out.println(""hello""+i);
    }
    public static void main(String... args){
        call(10);
    }
}
```

Please choose only one answer:

- compilation fails
- 10

Check the answer of this question online on JavaChamp.com: [boxing and varargs issue in java](#) Nikhil Kumar

1.12.2. How to compare between java wrappers?

Author: Yasser Ibrahim

Given the following code fragment, what is true about the following System.out statements?

```
public class Tester {  
  
    public static void main(String[] args) {  
        Integer[][] arr1 = { { 1, 2 }, { 3, 4 } };  
        Number[] arr2 = arr1[0];  
        int x = 1;  
  
        System.out.print(arr1[0][0] == 1);  
        System.out.print(arr2[0] == x);  
        System.out.print(x instanceof Integer);  
        System.out.print(arr1[1][0] > (Integer) arr2[0]);  
    }  
}
```

Please choose only one answer:

- All statements will compile
- Only one will compile and will print true
- Only two statements will compile and both will print true
- Only three statements will compile, only two of them will print true

Check the answer of this question online on JavaChamp.com: [How to compare between java wrappers?](#)

1.12.3. How does java autoboxing work?

Author: Yasser Ibrahim

Given that Long and Integer extend Number, what is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        Number x = 12; // Line 5  
        Number y = (Long) x; // Line 6  
        System.out.print(x+" "+y); // Line 7  
    }  
}
```

Please choose only one answer:

- 1212
- Compilation error at line 5
- Compilation error at line 6
- Compilation error at line 7
- An exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [How does java autoboxing work?](#)

1.12.4. Invoking equals() on java primitive type

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    public static void main(String[] args) {  
        Integer sum1 = 125; //line 1  
        int sum2 = 125; //line 2  
  
        System.out.print(sum1.equals(sum2)); //line 3  
        System.out.print(sum2.equals(sum1)); //line 4  
        System.out.print(sum1==sum2); //line 5  
    }  
}
```

Please choose only one answer:

- Compilation error at line 3
- Compilation error at line 4
- true>true>false
- true>true>true

Check the answer of this question online on JavaChamp.com: [Invoking equals\(\) on java primitive type](#)

1.12.5. java wrapper classes

Author: [pankaj gupta](#)

what will be the output when attempt to compile and execute this code:

```
class Test{
public static void main(String arg[])
{
Number n=10;
int i=10;
System.out.println(n==i);
}}
```

Please choose only one answer:

- true
- false
- compile time error
- run time exception

Check the answer of this question online on JavaChamp.com: [how to compare a primitive to a wrapper?](#)

1.12.6. var-args vs autoboxing in java

Author: [Java Champ](#)

Which, inserted independently at // insert code here , will compile? (choose four)

```
public class Tester {  
    // insert code here  
  
    public static void main(String[] args) {  
        calculate(12);  
    }  
}
```

Please choose all the answers that apply:

- static void calculate(Double i) {}
- static void calculate(Integer i) {}
- static void calculate(int i) {}
- static void calculate(int... i) {}
- static void calculate(Integer... i) {}

Check the answer of this question online on JavaChamp.com: [var-args vs autoboxing in java](#)

1.12.7. How does autoboxing work in java?

Author: Java Champ

What causes the compilation to fail?

```
public class Tester {  
    public static void main(String[] args) {  
        Long var = (long) 999; // Line 3  
        long x1 = var.longValue(); // Line 5  
        double x2 = var.longValue(); // Line 7  
        double x3 = (double) var.longValue(); // Line 9  
        Double x4 = Long.valueOf("999"); // Line 11  
        Number x5 = Integer.parseInt("999"); // Line 13  
        Long x6 = Integer.parseInt("999"); // Line 15  
    }  
}
```

Please choose all the answers that apply:

- Line 3
- Line 5
- Line 7
- Line 9
- Line 11
- Line 13
- Line 15

Check the answer of this question online on JavaChamp.com: [How does autoboxing work in java?](#)

13. Garbage Collection

Exam Category Description and Objectives

1.13.1. What makes java object eligible for garbage collection?

Author: Yasser Ibrahim

What can be inserted at // insert code here, to make object created at line 5 eligible for garbage collection?

```
public static void main(String[] args) {  
    Integer[] arr = new Integer[3];  
    arr[0] = new Integer("3"); // Line 5  
    Integer x = arr[0];  
    // insert code here  
}
```

Please choose only one answer:

- arr[0] = null;
- x = null;
- arr = null; x = new Integer(450);
- All of the above

Check the answer of this question online on JavaChamp.com: [What makes java object eligible for garbage collection?](#)

1.13.2. when does an object become subject to garbage collection in java?

Author: Java Champ

An object is subject to garbage collection in java :

Please choose only one answer:

- when the program is at a point of execution that is out the scope of the object
- when the object is set to null
- when the object becomes unreachable

Check the answer of this question online on JavaChamp.com: [when does an object become subject to garbage collection in java?](#)

1.13.3. When java object is eligible for garbage collection?

Author: Java Champ

How many objects are eligible for garbage collection after executing line 7?

```
public class Tester {  
    public static void main(String[] args) {  
        Integer x = new Integer(3000);  
        Integer y = new Integer(4000);  
        Integer z = new Integer(5000);  
  
        Object a = x;  
        x = y;  
        y = z;  
        z = null; //line 7  
    }  
}
```

Please choose only one answer:

- 0
- 1
- 2
- 3

Check the answer of this question online on JavaChamp.com: [When java object is eligible for garbage collection?](#)

14. Overriding and Overloading

Exam Category Description and Objectives

1.14.1. Can override java final method?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
class Base {  
    public final int getNext(int i) {  
        return ++i;  
    }  
}  
  
public class Derived extends Base {  
  
    public int getNext(int i) {  
        return i++;  
    }  
  
    public static void main(String[] args) {  
        int result = new Derived().getNext(3);  
        System.out.print(result);  
        result = new Base().getNext(3);  
        System.out.print(result);  
    }  
}
```

Please choose only one answer:

- 33
- 34
- 44
- 43
- a compilation error

Check the answer of this question online on JavaChamp.com: [Can override java final method?](#)

1.14.2. Legal overriding in java

Author: Abhishek Sawant

o/p of this program is 35. Which method executes, & whether it is method Overloading or method Overriding?

```
class BaseI{
    public int doSum(short a, short b){ // #1
        return(a+b);
    }
}

public class DerivedI extends BaseI{
    public int doSum(int a,short b){ // #2
        return(a+b);
    }

    public int doSum(short a, short b){ // #3
        return(a+b);
    }

    public static void main(String []args){
        DerivedI d = new DerivedI();
        short x =15;
        short y =20;
        System.out.println(d.doSum(x,y));
    }
}
```

Please choose only one answer:

- a) #1 & superclass method. No overloading or overriding
- b) #2 & overriding
- c) #2 & overloading
- d) #3 & overriding
- e) #3 & overloading
- f) None of the above

Check the answer of this question online on JavaChamp.com: [legal overriding in java](#)

1.14.3. Overriding rules in java

Author: Java Champ

Which statement(s), inserted independently at // insere code here, will compile? (choose three)

```
class Creature {  
    Creature getIt() {  
        return this;  
    }  
}  
  
class Bird extends Creature {  
    // insere code here  
}  
  
class Falcon extends Bird {  
}
```

Please choose all the answers that apply:

- Creature getIt() { return this;}
- private Falcon getIt() { return new Falcon();}
- Falcon getIt() {return this;}
- public Bird getIt() {return this;}
- Creature getIt(Creature c) {return this;}

Check the answer of this question online on JavaChamp.com: [Overriding rules in java](#)

1.14.4. var-args vs widening vs autoboxing in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static void call(Long x, Long y) {  
        System.out.print("Long x, Long y");  
    }  
  
    static void call(int... x) {  
        System.out.print("int... x");  
    }  
  
    public static void main(String[] args) {  
        int val = 3;  
        call(val, val);  
    }  
}
```

Please choose only one answer:

- Long x, Long y
- int... x
- Compilation error
- An exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [var-args vs widening vs autoboxing in java](#)

1.14.5. final parameters when overriding

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
class Base {
    public void method(final int x) {
        System.out.print("Base");
    }
}

public class Derived extends Base {

    public void method(int x) { // line 1
        System.out.print("Derived");
    }

    public static void main(String[] args) {
        Base b = new Derived();
        b.method(3);
    }
}
```

Please choose only one answer:

- will compile fine and wil print "Derived"
- will compile fine and wil print "Base"
- Won't compile because of line 1.Can be corrected by marking x as final,then the output will be "Derived"
- Won't compile because of line 1.Can be corrected by marking x as final,then the output will be "Base"

Check the answer of this question online on JavaChamp.com: [final parameters when overriding](#)

1.14.6. var-args vs autoboxing vs widening in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
  
    static void call(Long x, Long y) {  
        System.out.print("Long x, Long y");  
    }  
  
    static void call(int... x) {  
        System.out.print("int... x");  
    }  
  
    static void call(Number x, Number y) {  
        System.out.print("Number x, Number y");  
    }  
  
    public static void main(String[] args) {  
        int val = 3;  
        call(val, val);  
    }  
}
```

Please choose only one answer:

- Long x, Long y
- int... x
- Number x, Number y
- Compilation error
- An exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [var-args vs autoboxing vs widening in java](#)

1.14.7. java overriding (the flow of control)

Author: pankaj gupta

What will be the output :

```
class MySuper
{
    MySuper(){disp();}
    void disp(){System.out.println(&quot;superclass&quot;);}
}
class MySub extends MySuper
{
    double i=Math.ceil(8.4f);
    public static void main(String arg[])
    {
        MySuper obj= new MySub();
        obj.disp();
    }
    void disp(){System.out.println(i);}
}
```

Please choose only one answer:

- The program displays "superclass" followed by "9.0" as an output.
- The program displays "superclass" followed by "superclass" as an output.
- The program displays "9.0" followed by "9.0" as an output.
- The program displays "0.0" followed by "9.0" as an output.

Check the answer of this question online on JavaChamp.com: [Java overriding \(the flow of control\)](#)

1.14.8. Valid Return types in overriding

Author: Abhishek Sawant

What is the o/p of the given code?

```
class Ret{
    public long tryIt(){
        long num = 25;
        return num;    // $1
    }
}

class Return Type extends Ret{
    public long tryIt(){
        char ch = 'a';
        long num = 50;
        return ch;    // $2
    }

    public static void main(String []args){
        Ret r = new Return Type();    // $3
        System.out.println(r.tryIt());    // $4
    }
}
```

Please choose only one answer:

- a) 25
- b) 50
- c) Compilation Error at \$2
- d) Compilation Error at \$3
- e) Runtime Exception at \$4
- f) None of the above

Check the answer of this question online on JavaChamp.com: [valid return types in overriding in java](https://www.javachamp.com/question/valid-return-types-in-overriding-in-java)

1.14.9. Java overloading rules

Author: Yasser Ibrahim

What modification is necessary to produce the following output : SubType

```
class Type {
    public Object getTypeName(){
        return "Type";
    }
}

class SubType extends Type {
    public String getTypeName(){//line 8
        return "SubType";
    }
}

public class Tester {

    public static void main(String[] args) {
        Type first = new SubType(); //line 16
        System.out.println(first.getTypeName()); //line 17
    }
}
```

Please choose only one answer:

- No modification is needed
- At line 8 change the return type from String to Object to be a correct overloading of getTypeName()
- At line 16 change Type to SubType
- At line 17 cast first.getTypeName() to String

Check the answer of this question online on JavaChamp.com: [Java overloading rules](https://www.javachamp.com/question/1.14.9.9.java-overloading-rules)

1.14.10. How to override method with exceptions in java?

Author: Java Champ

Which, inserted independently at //insert overridden call() here, will represent an overridden call() and compile with no error?

```
class Base {  
    void call() throws IllegalArgumentException {  
    }  
}  
  
public class Derived extends Base {  
    //insert overridden call() here  
}
```

Please choose all the answers that apply:

- public void call() throws IllegalArgumentException { }
- void call() throws IllegalArgumentException,FileNotFoundException { }
- void call() throws RuntimeException { }
- void call() throws IllegalArgumentException,RuntimeException { }
- private void call() { }

Check the answer of this question online on JavaChamp.com: [How to override method with exceptions in java?](#)

15. Collections

Exam Category Description and Objectives

1.15.1. equals and hashCode contract in java

Author: Java Champ

Based on equals() and hashCode() contract in java, which of the following is true?

Please choose all the answers that apply:

- if o1.equals(o2) is true, then hashCode(o1)== hashCode(o2) must return true
- if o1.equals(o2) is true, then hashCode(o1)== hashCode(o2) might return true
- if o1.equals(o2) is false, then hashCode(o1)== hashCode(o2) must return false
- if o1.equals(o2) is false, then hashCode(o1)== hashCode(o2) might return true
- if hashCode(o1)== hashCode(o2) returns true then o1.equals(o2) must return true

Check the answer of this question online on JavaChamp.com: [equals and hashCode contract in java](#)

1.15.2. inserting into a TreeSet in java

Author: Yasser Ibrahim

The following code will throw a runtime exception, which line causes the exception?

```
import java.util.HashSet;
import java.util.Set;
import java.util.TreeSet;

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

        Set s = new HashSet();
        s.add("A"); // Line 1
        s.add(new Foo()); // Line 2

        Set t = new TreeSet();
        t.add("A"); // Line 3
        t.add(new Foo()); // Line 4
    }
}

class Foo {}
```

Please choose only one answer:

- Line 1
- Line 2
- Line 3
- Line 4

Check the answer of this question online on JavaChamp.com: [inserting into a TreeSet in java](#)

1.15.3. asList in java

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.util.*;

public class Test {

    public static void main(String[] args) {

        String[] arr = { "Java", "Champ", "." };
        List<String> list = (List<String>) Arrays.asList(arr); // line 1
        arr[2] = ".com"; // line 2
        for (String word : list) {
            System.out.print(word);
        }
    }
}
```

Please choose only one answer:

- JavaChamp.
- JavaChamp.com
- compilation error at line 1, the correct method is toList not asList
- compilation error at line 2, cannot modify array after converting to a List
- compilation error in other lines than lines 1 and 2

Check the answer of this question online on JavaChamp.com: [asList in java](#)

1.15.4. What does java headMap() do?

Author: Java Champ

What is the output of compiling and running the following program?

```
import java.util.NavigableMap;
import java.util.TreeMap;

public class Tester{

    public static void main(String[] args) {
        TreeMap tree = new TreeMap();
        tree.put("aa", 1);
        tree.put("cc", 2);
        tree.put("ee", 3);
        tree.put("gg", 4);

        NavigableMap nvMap = tree.headMap("ee", false);
        System.out.print(nvMap);

    }
}
```

Please choose only one answer:

- {aa=1, cc=2}
- {aa=1, cc=2, ee=3}
- {aa=1, cc=2, ee=3, gg=4}
- No output is produced
- Compilation error

Check the answer of this question online on JavaChamp.com: [What does java headMap\(\) do?](#)

1.15.5. How the Set collection allows no duplicates in java ?

Author: Yasser Ibrahim

In this program, the set humans allows for objects of type Human with the same age to be inserted. What can be done to make the set reject any Human object with value age equals to age value for an already inserted human object?

```
import java.util.HashSet;
import java.util.Set;

public class Test{

    public static void main(String[] args) {

        Set<Human> humans = new HashSet<Human>();
        humans.add(new Human(13));
        humans.add(new Human(33));
        humans.add(new Human(21));
        humans.add(new Human(21));
    }

}

class Human {
    Integer age;

    public Human(int age) {
        this.age = age;
    }

}
```

Please choose only one answer:

- only override method equals() in Human as follows:

```
public boolean equals(Object h) {
    return this.age==((Human)h).age;
}
```

- only override method hashCode() in Human as follows:

```
public int hashCode(){
    return 17;
}
```

- must override both methods hashCode() and equals() as illustrated in the above answers

Check the answer of this question online on JavaChamp.com: [How the Set collection allows no duplicates in java ?](#)

1.15.6. TreeMap methods in java

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.util.TreeMap;

public class Test {

    public static void main(String[] args) {
        TreeMap<Integer,String> map = new TreeMap<Integer,String>();
        map.put(1, "one");
        map.put(2, "two");
        map.put(3, "three");
        map.put(4, "four");
        System.out.print(map.higherKey(2));
        System.out.print(map.ceilingKey(2));
        System.out.print(map.floorKey(1));
        System.out.print(map.lowerKey(1));
    }
}
```

Please choose only one answer:

- 3211
- 321null
- 321-1
- 231null
- 2310

Check the answer of this question online on JavaChamp.com: [TreeMap methods in java](#)

1.15.7. How to use java headMap?

Author: Java Champ

What is the output of compiling and running the following program?

```
import java.util.NavigableMap;
import java.util.TreeMap;

public class Tester {

    public static void main(String[] args) {

        TreeMap tree = new TreeMap();
        tree.put("aa", 1);
        tree.put("cc", 2);
        tree.put("ee", 3);
        tree.put("gg", 4);

        NavigableMap nvMap = tree.headMap("ee", false);
        nvMap.put("nn", 5); // line 16
        System.out.println(nvMap);

    }
}
```

Please choose only one answer:

- {aa=1, cc=2, ee=3}
- {aa=1, cc=2, ee=3, nn=5}
- Compilation error at line 16
- Exception is thrown at line 16

Check the answer of this question online on JavaChamp.com: [How to use java headMap?](#)

1.15.8. how to override equals() in java?

Author: Yasser Ibrahim

What is the expected output after compiling and running the following code?

```
import java.util.HashSet;
import java.util.Set;

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

        Set<Foo> myFooSet = new HashSet<Foo>();
        myFooSet.add(new Foo(2));
        myFooSet.add(new Foo(1));
        myFooSet.add(new Foo(3));
        myFooSet.add(new Foo(3));
        myFooSet.add(new Foo(2));

        System.out.print(myFooSet.size());

    }
}

class Foo {
    Integer code;

    Foo(Integer c) {
        code = c;
    }

    public boolean equals(Foo f) {
        return false;
    }

    public boolean equals(Object f) {
        return true;
    }

    public int hashCode() {
        return 17;
    }
}
```

Please choose only one answer:

- 1
- 3
- 5
- compilation error

Check the answer of this question online on JavaChamp.com: [how to override equals\(\) in java?](#)

1.15.9. using binarySearch with an array in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.util.Arrays;

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

    String[] arr = { "java", "champ", "champion" };
    Arrays.sort(arr);
    System.out.print(Arrays.binarySearch(arr, "champion"));
    System.out.print(Arrays.binarySearch(arr, "You"));

}
}
```

Please choose only one answer:

- 10
- 1-1
- un expected result
- 20
- 2-1
- compilation error

Check the answer of this question online on JavaChamp.com: [using binarySearch with an array in java](#)

1.15.10. overriding equals() method in java

Author: Yasser Ibrahim

What is the output of compiling and running the following code?

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

        Foo foo1 = new Foo(1);
        Foo foo2 = new Foo(2);

        System.out.print(foo1.equals(foo2));

    }
}

class Foo {
    Integer code;

    Foo(Integer c) {
        code = c;
    }

    public boolean equals(Foo f) {
        return false;
    }

    public boolean equals(Object f) {
        return true;
    }
}
```

Please choose only one answer:

- true
- false

Check the answer of this question online on JavaChamp.com: [overriding equals\(\) method in java](#)

1.15.11. subSet in java

Author: [Java Champ](#)

What is the result of compiling and running the following code?

```
import java.util.SortedSet;
import java.util.TreeSet;

public class Test {

    public static void main(String[] args) {
        TreeSet<Integer> map = new TreeSet<Integer>();
        map.add(1);
        map.add(2);
        map.add(4);
        map.add(7);
        SortedSet<Integer> smap = map.subSet(2,7);
        map.add(5);
        map.add(9);
        System.out.println(smap);
    }
}
```

Please choose only one answer:

- [2, 4, 5, 7]
- [2, 4, 7]
- [2, 4, 5, 7, 9]
- [2, 4, 5]
- [2, 4]

Check the answer of this question online on [JavaChamp.com](#): [subSet in java](#)

1.15.12. tailMap, headMap and subMap in java

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.util.SortedMap;
import java.util.TreeMap;

public class Test {

    public static void main(String[] args) {
        TreeMap<Integer,String> map = new TreeMap<Integer,String>();
        map.put(1, "one");
        map.put(2, "two");
        map.put(3, "three");
        map.put(4, "four");
        SortedMap<Integer, String> smap1 = map.tailMap(2);
        SortedMap<Integer, String> smap2 = smap1.headMap(4);
        SortedMap<Integer, String> smap3 = smap2.subMap(2, 3);
        System.out.println(smap3);
    }
}
```

Please choose only one answer:

- {2=two, 3=three, 4=four}
- {2=two, 3=three}
- {2=two}
- no output is printed

Check the answer of this question online on JavaChamp.com: [tailMap, headMap and subMap in java](#)

1.15.13. binarySearch in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.util.Arrays;

public class Tester {
    public static void main(String[] args) {
        Integer[] arr = { 1, 2, 3 };
        System.out.print(Arrays.binarySearch(arr, 1)); // line 1
        System.out.print(Arrays.binarySearch(arr, &quot;javachamp&quot;)); // line 2
    }
}
```

Please choose all the answers that apply:

- line 1 will print 0
- line 1 will print 1
- line 1 causes compilation error
- line 2 will print 0
- line 2 will causes compilation error
- line 2 will causes RunTimeException

Check the answer of this question online on JavaChamp.com: [binarySearch in java](#)

1.15.14. PriorityQueue methods in java

Author: Yasser Ibrahim

What can be inserted, independently, at line // insert code here to cause the program to print out : [3, 4, 7, 0]
(choose three)

```
import java.util.*;

public class Test {

    public static void main(String[] args) {
        Queue<Integer> queue = new LinkedList<Integer>();
        queue.add(1);
        queue.add(3);
        queue.add(4);
        queue.add(7);

        // insert code here

        System.out.println(queue);
    }
}
```

Please choose all the answers that apply:

- queue.poll();queue.offer(0);queue.peek();
- queue.peek();queue.add(0);queue.peek();queue.poll();
- queue.add(0);queue.remove();queue.peek();
- queue.add(0);queue.poll();queue.remove();

Check the answer of this question online on JavaChamp.com: [PriorityQueue methods in java](https://www.javachamp.com/question/1.15.14-priorityqueue-methods-in-java)

1.15.15. how to sort a collection in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;

public class Tester{

    public static void main(String[] args) {

        List<Human> humans = new ArrayList<Human>();
        humans.add(new Human(13));
        humans.add(new Human(33));
        humans.add(new Human(21));
        humans.add(new Human(21)); // line 1
        HumanComparator c = new HumanComparator(); // line 2
        Collections.sort(humans, c); // line 3
        System.out.print(humans.get(0).age);
        Collections.sort(humans); // line 4
        System.out.print(humans.get(0).age);

    }

}

class Human implements Comparable<Human> {
    Integer age;

    public Human(int age) {
        this.age = age;
    }

    public int compareTo(Human h) {
        return h.age.compareTo(this.age);
    }

}

class HumanComparator implements Comparator<Human> {

    public int compare(Human h1, Human h2) {
        return h1.age.compareTo(h2.age);
    }

}
```

Please choose only one answer:

- line 1 causes a compilation error (no duplicates are allowed)
- line 2 causes a compilation error, the correct instantiation for HumanComparator is as so :
new HumanComparator<Human>();
- line 4 causes a compilation error, must supply a Comparator to method sort.

- The program compiles and runs fine printing 1333
- The program compiles and runs fine printing 3313

Check the answer of this question online on JavaChamp.com: [how to sort a collection in java?](#)

1.15.16. inserting into a HashSet in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.util.HashSet;
import java.util.Set;

public class Test{

    public static void main(String[] args) {

        Set<Human> humans = new HashSet<Human>();
        humans.add(new Human(13));
        humans.add(new Human(33));
        humans.add(new Human(21));
        humans.add(new Human(21));
        System.out.print(humans.size()+" ");
        System.out.print(humans);

    }

}

class Human implements Comparable<Human> {
    Integer age;

    public Human(int age) {
        this.age = age;
    }

    public int compareTo(Human h) {
        return h.age.compareTo(this.age);
    }

    public String toString() {
        return ""+this.age;
    }

}
```

Please choose only one answer:

- 4 [33, 21, 13, 21]
- 3 [21, 33, 13]
- compilation error
- exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [inserting into a HashSet in java](#)

1.15.17. ceilingKey in TreeMap in java

Author: Yasser Ibrahim

What does the method ceilingKey in TreeMap do?

Please choose only one answer:

- Returns the least key greater than or equal to the given key, or null if there is no such key.
- Returns the least key strictly greater than the given key, or null if there is no such key.
- Returns the least key in the tree

Check the answer of this question online on JavaChamp.com: [ceilingKey in TreeMap in java](#)

1.15.18. How to implement Comparable in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

public class Test{

    public static void main(String[] args) {

        List<Human> humans = new ArrayList<Human>();
        humans.add(new Human(13));
        humans.add(new Human(33));
        humans.add(new Human(21));
        humans.add(new Human(21));
        Collections.sort(humans);
        System.out.print(humans.get(0).age);
        System.out.print(humans.size());

    }

}

class Human implements Comparable<Human> {
    int age;

    public Human(int age) {
        this.age = age;
    }

    public int compareTo(Human h) {
        return h.age.compareTo(this.age);
    }

}
```

Please choose only one answer:

- 333
- 334
- 133
- 134
- Compilation fails

Check the answer of this question online on JavaChamp.com: [How to implement Comparable in java](#)

1.15.19. searching an array using binarySearch in java

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.util.Arrays;

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

        String[] arr = { "java", "champ", "you" };
        System.out.print(Arrays.binarySearch(arr, "java"));
        System.out.print(Arrays.binarySearch(arr, "You"));

    }
}
```

Please choose only one answer:

- 02
- 0-1
- 13
- the result is unpredictable
- compilation error

Check the answer of this question online on JavaChamp.com: [searching an array using binarySearch in java](#)

1.15.20. inserting into a non generic TreeSet in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.util.*;

public class Test {

    public static void main(String[] args) {
        TreeSet s = new TreeSet();
        s.add(1);
        s.add(99.9);
        s.add(99.9);
        s.add(96.9);

        for (int i = 0; i < s.size(); i++) {
            System.out.print(s.pollFirst()+" ");
        }
    }
}
```

Please choose only one answer:

- 1 96.9 99.9
- 1 96.9 99.9 99.9
- 1
- compilation error
- an exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [inserting into a non generic TreeSet in java](#)

1.15.21. how to sort java TreeSet?

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.util.Arrays;
import java.util.Set;
import java.util.TreeSet;

public class Tester extends Thread {

    public static void main(String[] args) {
        Integer[] arr = {7,5,7,3};
        Set<Integer> set = new TreeSet<Integer>(Arrays.asList(arr));
        set.add(4);
        for(Integer value: set){
            System.out.print(value);
        }
    }
}
```

Please choose only one answer:

- Compilation error
- 75734
- 3457
- 34577

Check the answer of this question online on JavaChamp.com: [how to sort java TreeSet?](#)

16. Generics

Exam Category Description and Objectives

1.16.1. Proper instantiation of java generic collection

Author: Java Champ

The following program contains two compilation errors, at what lines?

```
import java.util.ArrayList;
import java.util.List;

interface Chewable {}
class Meat implements Chewable {}

public class Tester {

    public static List<? extends Chewable> printSize(List<? extends Chewable> list) { // Line 10
        System.out.println(list.size());
        return list;
    }
    public static void main(String[] args) {

        List<? extends Chewable> list1 = new ArrayList<Meat>(); // Line 16
        List<Chewable> list2 = new ArrayList<Chewable>(); // Line 17
        List<Meat> list3 = new ArrayList<Meat>(); // Line 18

        list1 = printSize(list1); // Line 20
        list2 = printSize(list2); // Line 21
        list3 = printSize(list3); // Line 22

    }
}
```

Please choose all the answers that apply:

- Line 10
- Line 16
- Line 17
- Line 18
- Line 20
- Line 21
- Line 22

Check the answer of this question online on JavaChamp.com: [Proper instantiation of java generic collection](#)

1.16.2. java generic collection rules?

Author: [Java Champ](#)

Which of the following is a correct declaration and instantiation of list? (choose four)

Please choose all the answers that apply:

- `List<String> list = new ArrayList<String>();`
- `List<?> list = new ArrayList<String>();`
- `List<? extends Object> list = new ArrayList<String>();`
- `List<Object> list = new ArrayList<String>();`
- `List list = new ArrayList<String>();`
- `List list = new ArrayList<?>();`

Check the answer of this question online on [JavaChamp.com: java generic collection rules?](#)

1.16.3. How to declare java generic collection?

Author: Yasser Ibrahim

What is the result of compiling and running the following program using jdk 1.5 or higher?

```
import java.util.ArrayList;
import java.util.List;

interface chewable {}

class Gum implements chewable {}

class Meat implements chewable {}

public class Tester {

    public static void main(String[] args) {
        List list1 = new ArrayList<chewable>();// line 1
        list1.add(new Gum());// line 2
        list1.add(new Meat());// line 3
        list1.add(new Integer(9));// line 4
        System.out.println(list1.size());// line 5
    }
}
```

Please choose only one answer:

- Compilation error at line 1
- Compilation error at line 4
- Runtime exception thrown at line 1
- Runtime exception thrown at line 4
- 3

Check the answer of this question online on JavaChamp.com: [How to declare java generic collection?](https://www.javachamp.com/question/1.16.3-how-to-declare-java-generic-collection/)

1.16.4. When were Generics first introduced in Java?

Author: [Java Champ](#)

When were generics first introduced in Java?

Please choose only one answer:

- in JDK 1.4
- in JDK 1.5
- in JDK 1.6

Check the answer of this question online on [JavaChamp.com](#): [When were Generics first introduced in Java?](#)

1.16.5. generic collection as return type in java

Author: Java Champ

Considering that all needed imports are there, what, inserted independently at line 1, will compile? (choose four)

```
class A {}
class B extends A{}
class C extends B{}

public class Test {

    public List<? extends B> ring(){
        // line 1
    }
}
```

Please choose all the answers that apply:

- return new ArrayList();
- return new ArrayList<C>();
- return new ArrayList<A>();
- return new ArrayList<? extends B>();
- return new LinkedList();
- return new PriorityQueue();
- return new ArrayList();

Check the answer of this question online on JavaChamp.com: [generic collection as return type in java](#)

1.16.6. When cannot add to a generic collection?

Author: Yasser Ibrahim

The following code contains a compilation error, at what line?

```
import java.util.ArrayList;
import java.util.List;

interface Chewable {}
class Meat implements Chewable {}

public class Tester {

    public static void main(String[] args) {

        List<? extends Chewable> list1 = new ArrayList<Meat>(); // Line 11

        List<Chewable> list2      = new ArrayList<Chewable>(); // Line 13

        Meat meat = new Meat();

        list1.add(meat); // Line 17

        list2.add(meat); // Line 19
    }
}
```

Please choose only one answer:

- Line 11
- Line 13
- Line 17
- Line 19

Check the answer of this question online on JavaChamp.com: [When cannot add to a generic collection?](#)

1.16.7. inserting into a non-generic collection in java

Author: Yasser Ibrahim

What is the expected output of compiling and running the following code?

```
import java.util.ArrayList;
import java.util.List;

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

        List list = new ArrayList();

        list.add("Hello");
        Foo f = new Foo();
        list.add(f);

        f = list.get(1);
        System.out.print(list.get(0) + "-" + f);

    }
}

class Foo {
    public String toString() {
        return "Foo";
    }
}
```

Please choose only one answer:

- Hello-Foo
- compilation error
- exception at run time

Check the answer of this question online on JavaChamp.com: [inserting into a non-generic collection in java](#)

1.16.8. How to declare instantiate java generic collection?

Author: Yasser Ibrahim

What is the result of compiling and running the following program?

```
import java.util.ArrayList;
import java.util.List;

public class Tester {

    public static void main(String[] args) {

        List<String> list1 = new ArrayList<String>();//line 1
        List<Object> list2 = list1;//line 2
        list2.add(new Integer(12));//line 3
        System.out.println(list2.size());//line 4

    }
}
```

Please choose only one answer:

- Compilation error at line 1, must initialize the size
- Compilation error at line 2, Type mismatch, cannot convert from List<String> to List<Object>
- Compilation error at line 3, can add objects of type String only
- 1

Check the answer of this question online on JavaChamp.com: [How to declare instantiate java generic collection?](https://www.javachamp.com/question/1168/how-to-declare-instantiate-java-generic-collection/)

1.16.9. how to sort a generic collection?

Author: Yasser Ibrahim

What is the expected output of compiling and running the following code?

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

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

        List<Foo> myFooList = new ArrayList<Foo>();
        myFooList.add(new Foo("C"));
        myFooList.add(new Foo("A"));
        myFooList.add(new Foo("D"));

        Collections.sort(myFooList);
        System.out.print(myFooList.get(0).code);

    }
}

class Foo implements Comparable<Foo> {
    String code;

    Foo(String c) {
        code = c;
    }

    int compareTo(Foo f) {
        return this.code.compareTo(f.code);
    }
}
```

Please choose only one answer:

- A
- C
- D
- no output is printed
- compilation error
- exception thrown at run time

Check the answer of this question online on JavaChamp.com: [how to sort a generic collection?](#)

1.16.10. generic methods in java

Author: Yasser Ibrahim

Will the following code compile?

```
import java.util.*;

class B {
}

class C extends B {
}

public class Test {

    public static <E extends B> Queue<E> ring(List<E> list) {
        return null;
    }

    public static void main(String[] args) {
        List<C> list1 = new ArrayList<C>();
        ArrayList<C> list2 = new ArrayList<C>();

        Queue<C> q1;
        PriorityQueue<C> q2;

        q1 = ring(list1); // line1
        q1 = ring(list2); // line2
        q2 = ring(list1); // line3
        q2 = ring(list2); // line4
    }
}
```

Please choose all the answers that apply:

- Yes, the program is free of compilation errors
- No, there is a compilation error at // Line 1
- No, there is a compilation error at // Line 2
- No, there is a compilation error at // Line 3
- No, there is a compilation error at // Line 4

Check the answer of this question online on JavaChamp.com: [generic methods in java](#)

1.16.11. how to declare a generic collection in java?

Author: Yasser Ibrahim

How can this program be modified to make use of appropriate generic types? choose 3 answers (one modification for each line)

```
import java.util.*;

public class Test {

    public static void main(String[] args) {
        List ids = new ArrayList(); // Line 1
        ids.add(123);
        ids.add(999);

        Map students = new HashMap(); // Line 2
        students.put("Jess",ids.get(0));
        students.put("Jimmy",ids.get(1));

        int x = ((Long)students.get("Jimmy")).intValue(); // Line 3
    }
}
```

Please choose all the answers that apply:

- replace line 1 with `List<Integer> ids = new ArrayList<Integer>();`
- replace line 1 with `List<Long> ids = new ArrayList<Long>();`
- replace line 2 with `Map<Integer,String> students = new HashMap<Integer,String>();`
- replace line 2 with `Map<String,Integer> students = new HashMap<String,Integer>();`
- replace line 3 with `int x = students.get("Jimmy");`
- replace line 3 with `Long x = students.get("Jimmy");`

Check the answer of this question online on JavaChamp.com: [how to declare a generic collection in java?](#)

1.16.12. how to declare a generic Map in java?

Author: Java Champ

Given the following incomplete program, based on the given put statements in map, what is considered a correct declaration and instantiation for map ?

```
import java.util.*;

public class Test {

    public static void main(String[] args) {

        // insert code here

        map.put(new ArrayList<Integer>(), 1);
        map.put(new ArrayList<Integer>(), 12);
        map.put(new LinkedList<Integer>(), new Integer(1));
        map.put(new LinkedList<Integer>(), new Long(1));

    }
}
```

Please choose only one answer:

- 1) Map<List<Integer>, ? extends Number>; map = new HashMap<List<Integer>, Number>();
- 2) Map<List<Integer>, Number>; map = new HashMap<List<Integer>, Integer>();
- 3) Map<List<Integer>, Number>; map = new HashMap<List<Integer>, Number>();
- 4) Map<List<Integer>, Integer>; map = new HashMap<List<Integer>, Integer>();
- 5) Map<ArrayList<Integer>, Integer>; map = new HashMap<ArrayList<Integer>, Integer>();
- 6) Map<List<Integer>, Long>; map = new HashMap<List<Integer>, Long>();

Check the answer of this question online on JavaChamp.com: [how to declare a generic Map in java?](https://www.javachamp.com/question/1.16.12-how-to-declare-a-generic-map-in-java/)

1.16.13. How to add to java generic collection?

Author: Java Champ

What is the result of compiling and running the following program using JDK 1.5 or higher?

```
import java.util.ArrayList;
import java.util.List;

interface chewable {}

class Gum implements chewable {}

public class Tester {

    public static void main(String[] args) {
        List<Gum> list1 = new ArrayList<Gum>();
        list1.add(new Gum());
        List list2 = list1;
        list2.add(new Integer(9));
        System.out.println(list2.size());
    }
}
```

Please choose only one answer:

- Compilation error
- 2 will be printed but with warnings
- 2 will be printed without warnings
- An exception will be thrown at runtime

Check the answer of this question online on JavaChamp.com: [How to add to java generic collection?](#)

17. Formatting

Exam Category Description and Objectives

1.17.1. java classes instantiated using factory method

Author: Yasser Ibrahim

Which of the following is instantiated using a factory method ?

Please choose all the answers that apply:

- Date
- Calendar
- Locale
- DateFormat
- NumberFormat

Check the answer of this question online on JavaChamp.com: [java classes instantiated using factory method](#)

1.17.2. writing java printf or format statement

Author: Yasser Ibrahim

Which of the following statements produces the output 3 + 8 ?

```
public class Tester {  
  
    public static void main(String[] args) {  
        int x = 8;  
        int y = 3;  
  
        System.out.printf("%d + %d \n", y, x); // stmt1  
        System.out.printf("%f + %f \n", (float) y, (float) x); // stmt2  
        System.out.printf("%d + %d \n", x, y); // stmt3  
        System.out.format("%2$d + %1$d", x, y); // stmt4  
    }  
}
```

Please choose all the answers that apply:

- Statement 1
- Statement 2
- Statement 3
- Statement 4

Check the answer of this question online on JavaChamp.com: [writing java printf or format statement](#)

18. I/O

Exam Category Description and Objectives

1.18.1. How to create file in java?

Author: Yasser Ibrahim

At what line in the following program the file "data.txt" will be created in the file system?

```
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

public class Tester {

    public static void main(String[] args) {
        try {
            File file = new File(&quot;data.txt&quot;); // line 5
            file.createNewFile(); // line 6
            FileWriter fr = new FileWriter(file); // line 7
            BufferedWriter br = new BufferedWriter(fr); // line 8
            br.append(&quot;javachamp&quot;);
            br.flush();
            br.close(); // line 11
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

Please choose only one answer:

- Line 5
- Line 6
- Line 7
- Line 8
- Line 11

Check the answer of this question online on JavaChamp.com: [How to create file in java?](#)

1.18.2. How to instantiate java PrintWriter?

Author: Yasser Ibrahim

Which of the following statements is a correct instantiation of PrintWriter?

Please choose all the answers that apply:

- `PrintWriter writer = new PrintWriter("file.dat");`
- `PrintWriter writer = new PrintWriter(new BufferedWriter("file.dat"));`
- `PrintWriter writer = new PrintWriter(new FileWriter("file.dat"));`
- `PrintWriter writer = new PrintWriter(new File("file.dat"));`

Check the answer of this question online on JavaChamp.com: [How to instantiate java PrintWriter?](#)

1.18.3. What java exceptions thrown by File?

Author: Yasser Ibrahim

Given the following code segment enclosed within a try/catch block, what valid Exception type can be caught causing no compilation error? (choose three)

```
try {  
    File file = new File("&quot;file.dat&quot;");  
    file.createNewFile();  
  
} catch (  
    // INSERT EXCEPTION TYPE  
    e) {  
    e.printStackTrace();  
}
```

Please choose all the answers that apply:

- Exception
- RuntimeException
- Throwable
- FileNotFoundException
- IOException

Check the answer of this question online on JavaChamp.com: [What java exceptions thrown by File?](#)

1.18.4. How to instantiate java BufferedWriter?

Author: [Java Champ](#)

Which of the following is a correct instantiation for a BufferedWriter?

Please choose all the answers that apply:

- `BufferedWriter bw = new BufferedWriter(new File("data.txt"));`
- `BufferedWriter bw = new BufferedWriter(new FileWriter("data.txt"));`
- `BufferedWriter bw = new BufferedWriter(new PrintWriter("data.txt"));`
- `BufferedWriter bw = new BufferedWriter("data.txt");`

Check the answer of this question online on [JavaChamp.com](#): [How to instantiate java BufferedWriter?](#)

19. Threads

Exam Category Description and Objectives

1.19.1. synchronized static method and threads in java

Author: Yasser Ibrahim

What is the possible output of running this program once as it is and once by marking swim() synchronized?

```
class Swimmer implements Runnable{
    String name ;
    Swimmer(String name){
        this.name = name;
    }
    public void run() {
        Test.swim(name);
    }
}

public class Test {

    public static void swim(String name) {
        System.out.print(name);
        System.out.print(name);
    }

    public static void main(String[] args) {
        new Thread(new Swimmer("Tom")).start();
        new Thread(new Swimmer("Hanks")).start();
    }
}
```

Please choose all the answers that apply:

- as it is, a possible output is TomHanksTomHanks
- as it is, a possible output is TomHanksHanksTom
- by synchronizing swim(), a possible output is TomHanksHanksTom
- by synchronizing swim(), a possible output is HanksHanksTomTom
- by synchronizing swim(), a possible output is TomHanksTomHanks

Check the answer of this question online on JavaChamp.com: [synchronized static method and threads in java](#)

1.19.2. using wait and sleep in threads in java

Author: Java Champ

What are the possible results of compiling and running the following code? (choose four)

```
public class Test {  
    public static void main(String[] args) throws InterruptedException {  
        Runnable t1 = new Runnable() {  
            public void run() {  
                try {  
                    System.out.print(""t1before"");  
                    Thread.sleep(100);  
                    System.out.print(""t1after"");  
                } catch (InterruptedException e) {  
                }  
            }  
        };  
  
        final Thread t2 = new Thread() {  
            public void run() {  
                try {  
                    System.out.print(""t2before"");  
                    wait();  
                    System.out.print(""t2after"");  
                } catch (InterruptedException e) {  
                }  
            }  
        };  
  
        t2.start();  
        new Thread(t1).start();  
    }  
}
```

Please choose all the answers that apply:

- t1before may be part of th output
- t1after may be part of th output
- t2before may be part of th output
- t2after may be part of th output
- an InterruptedException will be thrown at run time
- a NumberFormatException will be thrown

Check the answer of this question online on JavaChamp.com: [using wait and sleep in threads in java](#)

1.19.3. synchronized method in java

Author: Java Champ

What are the possible outputs of running this program once as it is, and second with marking swimIn() synchronized?

```
class Swimmer implements Runnable {
    String name;
    Pool pool;

    Swimmer(String name, Pool pool) {
        this.name = name;
        this.pool = pool;
    }

    public void run() {
        pool.swimIn(name);
    }
}

public class Pool {

    public void swimIn(String name) {
        System.out.print(name);
        System.out.print(name);
    }

    public static void main(String[] args) {
        Pool pool = new Pool();
        new Thread(new Swimmer("Tom", pool)).start();
        new Thread(new Swimmer("Hanks", pool)).start();
    }
}
```

Please choose all the answers that apply:

- as it is, the output is always two consecutive "Tom" followed by two consecutive "Hanks" or viceversa
- as it is, the output could be TomHanksTomHanks
- by synchronizing swimIn(), the output is always two consecutive "Tom" followed by two consecutive "Hanks" or viceversa
- by synchronizing swimIn(), the output could be TomHanksTomHanks
- by synchronizing swimIn(), the output could be TomHanksHanksTom

Check the answer of this question online on JavaChamp.com: [synchronized method in java](#)

1.19.4. instantiate a thread by implementing Runnable in java

Author: Java Champ

What is the output of compiling and running the following code?

```
class Writer extends Thread {  
    public void run() {  
        System.out.println("Writer run");  
    }  
  
    public static void main(String[] args) {  
        Runnable c = new Writer();  
        Thread t = new Thread(c);  
        t.run();  
    }  
}
```

Please choose only one answer:

- Compilation error
- compiles fine, but no output produced
- compiles fine and prints "Writer run"
- compiles fine but throws an exception

Check the answer of this question online on JavaChamp.com: [instantiate a thread by implementing Runnable in java](#)

1.19.5. What is the output of the program?

Author:

Synchronization issue

```
class SyncDemo{
    public static void main(String... args){
        class A implements Runnable{
            public synchronized void run(){
                display();
            }
            synchronized void display(){
                for(int i=0;i<5;i++){
                    System.out.print("&quot;Hello&quot;");
                    try{
                        Thread.sleep(2000);
                    } catch (InterruptedException e){}
                    System.out.println(Thread.currentThread().getName());
                }
            }
        }

        A ob1=new A();
        A ob2=new A();
        Thread ob3=new Thread(ob1,"Saurabh");
        Thread ob4=new Thread(ob2,"Nikhil");
        ob3.start();
        ob4.start();
    }
}
```

Please choose only one answer:

- Compilation Fails
- Hello Saurabh
Hello Nikhil
.....
Printed in Synchronized way

Check the answer of this question online on JavaChamp.com: [synchronization in java Nikhil Kumar](#)

1.19.6. join thread in java

Author: Java Champ

What is true?

```
public class Test implements Runnable {  
    public static void main(String[] args) {  
        Test test = new Test();  
        Thread thread = new Thread(test);  
        thread.start();  
        thread.join();  
        System.out.print("main");  
    }  
  
    public void run() {  
        System.out.print("run");  
    }  
}
```

Please choose only one answer:

- the program could prints runmain
- the program could prints mainrun
- the compilation fails
- an exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [join thread in java](#)

1.19.7. thread join java

Author: Java Champ

What is ture?

```
public class Test implements Runnable {  
    public static void main(String[] args) throws InterruptedException {  
        Test test = new Test();  
        Thread t= new Thread(test);  
        t.start();  
        t.join();  
        System.out.print("main");  
    }  
  
    public void run() {  
        System.out.print("run");  
    }  
}
```

Please choose only one answer:

- the output could be "mainrun"
- the output could be "runmain"
- the output could be "run" then an exception is thrown at run time
- compilation error

Check the answer of this question online on JavaChamp.com: [thread join java](#)

1.19.8. which thread methods are static?

Author: [Java Champ](#)

Which of the following methods defined in Thread class are static?

Please choose all the answers that apply:

- `sleep()`
- `start()`
- `yield()`
- `join()`
- `run()`

Check the answer of this question online on [JavaChamp.com](#): [which thread methods are static?](#)

1.19.9. which methods are defined in calss object?

Author: Yasser Ibrahim

Which methods are defined in calss Object?

Please choose all the answers that apply:

- wait()
- sleep()
- toString()
- finalize()
- notify()

Check the answer of this question online on JavaChamp.com: [which methods are defined in calss object?](#)

1.19.10. starting many threads together in java

Author: Java Champ

What is the possible output of compiling and running the following code?

```
public class Test {  
    public static void main(String[] args) {  
        Thread request1 = new Thread(new InternetRequest (),&quot;request#1 &quot;);  
        Thread request2 = new Thread(new InternetRequest (),&quot;request#2 &quot;);  
  
        request1.start();  
        request2.start();  
    }  
}  
  
class InternetRequest implements Runnable {  
  
    public void run() {  
        System.out.print(Thread.currentThread().getName());  
    }  
}
```

Please choose all the answers that apply:

- request#2 request#1
- request#1 request#2
- an InterruptedException is thrown
- request#1 request#1

Check the answer of this question online on JavaChamp.com: [starting many threads together in java](#)

1.19.11. instantiate a thread by implementing Runnable interface

Author: Yasser Ibrahim

How many times the statement "we are painting" would be printed in this program?

```
public class Test{  
    public static void main(String[] args) {  
        Painter painter1 = new Painter();  
        painter1.start();  
  
        Painter painter2 = new Painter();  
        painter2.start();  
    }  
}  
  
class Painter implements Runnable {  
    public void run() {  
        System.out.println("we are painting");  
    }  
}
```

Please choose only one answer:

- two times
- zero times
- the program will not compile

Check the answer of this question online on JavaChamp.com: [instantiate a thread by implementing Runnable interface](#)

Author: Java Champ

What is the possible result of compiling and running the following code?

```
public class Test implements Runnable {
    Integer id;

    public static void main(String[] args) {
        new Thread(new Test()).start();
        new Thread(new Test()).start();
    }

    public void run() {
        press(id);
    }

    synchronized void press(Integer id) {
        System.out.print(id.intValue());
        System.out.print(++id.intValue());
    }
}
```

Please choose only one answer:

- 0101
- 0011
- -10-10
- -1-100
- compilation error
- an exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [thread and synchronized method in java](#)

1.19.13. Using wait with no notify on Thread

Author:

What is the output of running the following program?

```
class Tester extends Thread {
    int total;

    public static void main(String[] args) throws Exception {
        Tester t = new Tester();
        t.start();
        System.out.println("hi how are you:");
        synchronized (t) {
            System.out.println("waiting for t to complete");
            t.wait();
            System.out.println("total" + t.total);
        }
    }

    synchronized public void run() {
        for (int i = 0; i < 3; i++) {
            total = total + i;
        }
    }
}
```

Please choose only one answer:

- main thread will wait indefinitely.
- total 0 is part of the output.
- total 3 is part of the output .
- it depends upon which thread has got the cpu first .

Check the answer of this question online on JavaChamp.com: [java thread wait notify usage](#)

1.19.14. When java IllegalMonitorStateException is thrown?

Author: Java Champ

What is the result of compiling and running the following program?

```
public class Tester {  
    public void validate() {  
        int i = 0;  
        while (++i < 3) {  
            try {  
                wait();  
            } catch (InterruptedException e) {  
                e.printStackTrace();  
            }  
            System.out.print(i);  
        }  
    }  
  
    public static void main(String[] args) {  
        new Tester().validate();  
    }  
}
```

Please choose only one answer:

- Compilation error because of calling wait() outside a synchronized block
- Compilation error because IllegalMonitorStateException is not handled
- At runtime, it throws an IllegalMonitorStateException when trying to wait
- 12

Check the answer of this question online on JavaChamp.com: [When java IllegalMonitorStateException is thrown?](#)

1.19.15. synchronized method and threads in java

Author: Yasser Ibrahim

What are the possible outputs of running this program once as it is, and second with marking swim() synchronized?

```
class Swimmer implements Runnable{
    String name ;
    Swimmer(String name){
        this.name = name;
    }
    public void run() {
        new Test().swim(name);
    }
}

public class Test {

    public void swim(String name) {
        System.out.print(name);
        System.out.print(name);
    }

    public static void main(String[] args) {
        new Thread(new Swimmer("Tom")).start();
        new Thread(new Swimmer("Hanks")).start();
    }
}
```

Please choose all the answers that apply:

- as it is, the output is always consecutive two Tom then two Hanks or consecutive two Hanks then two Tom
- as it is, the output is undetermined. It is in any order of two Tom and two Hanks
- by synchronizing swim(), the output is always consecutive two Tom then two Hanks or consecutive two Hanks then two Tom
- by synchronizing swim(), the output is undetermined. It is in any order of two Tom and two Hanks

Check the answer of this question online on JavaChamp.com: [synchronized method and threads in java](#)

1.19.16. Can java thread invoke start more than once?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester extends Thread {  
    public void run() {  
        System.out.print("run");  
    }  
  
    public static void main(String[] args) {  
        Tester thread = new Tester();  
        new Thread(thread).start();  
        new Thread(thread).start();  
    }  
}
```

Please choose only one answer:

- Compilation error, can't invoke start() twice
- runrun
- InterruptedException will be thrown because of the second invoke to start()
- run

Check the answer of this question online on JavaChamp.com: [Can java thread invoke start more than once?](https://www.javachamp.com/question/can-java-thread-invoke-start-more-than-once/)

1.19.17. synchronizing and multithreading in java

Author: [Yasser Ibrahim](#)

The idea of this program is to allow two workers to build a wall (which consists of bricks and cement) , given the following code, what necessarily modifications are needed to build a wall by alternating between bricks and cement (brickcementbrickcement...) and to avoid as possible, one worker monopolizing the work alone ? (choose all what apply)

```

class Worker extends Thread {
    Contract contract;

    Worker(Contract contract) {
        this.contract = contract;
    }

    public void run() {
        contract.work();
    }
}

public class Contract {

    StringBuilder wall = new StringBuilder(""brick"");
    boolean isCementLastAdded = false;

    public void putBrick() {
        if (isCementLastAdded && !isWallDone()) {
            wall.append(""brick"");
            isCementLastAdded = false;
        }
    }

    public void putCementLayer() {
        if (!isCementLastAdded && !isWallDone()) {
            wall.append(""cement"");
            isCementLastAdded = true;
        }
    }

    public boolean isWallDone() {
        return wall.length() >= 100;
    }

    public void work() {
        while (!isWallDone()) {
            putCementLayer();
            putBrick();
        }
    }

    public static void main(String[] args) {
        Contract contract = new Contract();
        new Worker(contract).start();
        new Worker(contract).start();
    }
}

```

Please choose all the answers that apply:

- synchronize putBrick()
- synchronize putCementLayer()
- synchronize work()
- add Thread.sleep() after putBrick() in work() method
- synchronize isWallDone()

1.19.18. sleep in thread in java

Author: [Java Champ](#)

What does sleep(long millis) in Thread class do?

Please choose only one answer:

- causes the thread, which sleep is invoked on, to sleep (temporarily cease execution) for the specified number of milliseconds
- causes the currently executing thread to sleep (temporarily cease execution) for the specified number of milliseconds
- causes the main() thread to sleep for the specified number of milliseconds
- causes the currently executing thread to wait(temporarily cease execution) for the specified number of milliseconds then brings it back to run.

Check the answer of this question online on [JavaChamp.com](#): [sleep in thread in java](#)

1.19.19. how to synchronize a method in java

Author: Yasser Ibrahim

What is the possible result of compiling and running the following code?

```
class Swimmer implements Runnable {
    String name;
    Pool pool;

    Swimmer(String name, Pool pool) {
        this.name = name;
        this.pool = pool;
    }

    public void run() {
        pool.swimIn(name);
    }
}

public class Pool {

    public void swimIn(String name) {
        synchronized {
            System.out.print(name);
            System.out.print(name);
        }
    }

    public static void main(String[] args) {
        Pool pool = new Pool();
        new Thread(new Swimmer("Tom", pool)).start();
        new Thread(new Swimmer("Hanks", pool)).start();
    }
}
```

Please choose only one answer:

- TomTomHanksHanks
- HanksHanksTomTom
- HanksTomHanksTom
- undetermined order
- compilation error

Check the answer of this question online on JavaChamp.com: [how to synchronize a method in java](#)

1.19.20. running a thread in java

Author: Yasser Ibrahim

What is the could be the output of compiling and running the following code? (choose two)

```
public class Test extends Thread {
    static int count = 0;

    public static void main(String argv[]) throws InterruptedException {
        Test t = new Test ();
        t.increment(count);
        t.start();
        Thread.sleep(1000);
        System.out.println(count);
    }

    public void increment(int count) {
        ++count;
    }

    public void run() {
        count = count + 5;
    }
}
```

Please choose all the answers that apply:

- 5
- 6
- Compilation error
- An InterruptedException may be thrown

Check the answer of this question online on JavaChamp.com: [running a thread in java](#)

1.19.21. When java IllegalStateException is thrown?

Author: Java Champ

What could be a part of the output of compiling and running the following code?

```
public class Tester extends Thread {  
    public void run() {  
        System.out.println("run");  
    }  
  
    public static void main(String[] args) {  
        Thread thread = new Tester();  
        thread.run();  
        thread.start();  
        thread.start();  
    }  
}
```

Please choose all the answers that apply:

- Compilation error
- Prints "run"; twice, not necessarily consecutively
- InterruptedException will be thrown
- IllegalStateException will be thrown
- Prints "run"; three times

Check the answer of this question online on JavaChamp.com: [When java IllegalStateException is thrown?](#)

1.19.22. when a thread can call wait()?

Author: [Java Champ](#)

A thread that invokes the wait() method of an object, must own the lock of the object.

Please choose only one answer:

- True
- False

Check the answer of this question online on [JavaChamp.com](#): [when a thread can call wait\(\)?](#)

1.19.23. how can a thread own the lock of an object?

Author: Yasser Ibrahim

How can a thread own the lock of an object?

Please choose all the answers that apply:

- When the thread executes a synchronized instance method of that object
- When the thread executes a synchronized statement block in the object
- When the thread calls wait() on this object

Check the answer of this question online on JavaChamp.com: [how can a thread own the lock of an object?](#)

1.19.24. synchronized non static method in java

Author: Yasser Ibrahim

What is the possible result of compiling and running the following code?

```
public class Test implements Runnable {
    Integer id = 0;

    public static void main(String[] args) {
        new Thread(new Test()).start();
        new Thread(new Test()).start();
    }

    public void run() {
        press(id);
    }

    synchronized void press(Integer id) {
        System.out.print(id.intValue());
        System.out.print(++id.intValue());
    }
}
```

Please choose only one answer:

- 1010
- 0101
- 0123
- compilation error
- an exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [synchronized non static method in java](#)

1.19.25. when to override thread method run()?

Author: Yasser Ibrahim

When a class implements interface Runnable, it must provide implementation for method start():

Please choose only one answer:

- False
- True

Check the answer of this question online on JavaChamp.com: [when to override thread method run\(\)?](#)

1.19.26. using sleep and wait in thread in java

Author: Java Champ

What is the possible results of compiling and running the following code? (choose three)

```
public class Test {  
    static Runnable t1 = new Runnable() {  
        public void run() {  
            try {  
                System.out.print("&quot;t1before&quot;");  
                Thread.sleep(100);  
                System.out.print("&quot;t1after&quot;");  
            } catch (InterruptedException e) {  
            }  
        }  
    };  
  
    static Thread t2 = new Thread() {  
        public void run() {  
            try {  
                System.out.print("&quot;t2before&quot;");  
                synchronized (this) {  
                    wait();  
                }  
                System.out.print("&quot;t2after&quot;");  
            } catch (InterruptedException e) {  
            }  
        }  
    };  
  
    public static void main(String[] args) throws InterruptedException {  
        new Thread(t1).start();  
        t2.start();  
    }  
}
```

Please choose all the answers that apply:

- t1before may be part of the output
- t1after may be part of the output
- t2before may be part of the output
- t2after may be part of the output
- compilation fails
- IllegalMonitorStateException is thrown at run time

Check the answer of this question online on JavaChamp.com: [using sleep and wait in thread in java](#)

1.19.27. how to instantiate a thread object?

Author: Yasser Ibrahim

Which of the following Thread instantiations are correct?

```
public static void main(String[] args) {  
    new Thread(); // line 1  
    new Thread("myThread"); // line 2  
    new Thread(new Long(14)); // line 3  
    new Thread(new Runnable(){public void run(){} }); // line 4  
    Thread.getInstance(); // line 5  
}
```

Please choose all the answers that apply:

- line 1
- line 2
- line 3
- line 4
- line 5

Check the answer of this question online on JavaChamp.com: [how to instantiate a thread object?](#)

1.19.28. How java thread start running?

Author: Java Champ

What is considered an impossible output of running the following program?

```
public class Tester extends Thread {  
    int code = 9;  
  
    public void run() {  
        this.code = 7;  
    }  
  
    public static void main(String[] args) {  
  
        Tester thread = new Tester();  
        thread.start();  
  
        for (int i = 0; i < 5; i++) {  
            System.out.print(thread.code);  
        }  
    }  
}
```

Please choose only one answer:

- 99777
- 97777
- 77777
- 79999
- 99999

Check the answer of this question online on JavaChamp.com: [How java thread start running?](#)

1.19.29. Create a thread by implementing Runnable

Author: Java Champ

When creating a thread by implementing Runnable interface :

Please choose only one answer:

- you must implement the method run()
- you can override run(), but in case you don't you'll be restricted to the provided run() method, which contains no code and does nothing

Check the answer of this question online on JavaChamp.com: [Create a thread by implementing Runnable](#)

1.19.30. run vs start in threads in java

Author: Yasser Ibrahim

What are the possible results of compiling and running the following code?

```
public class Test implements Runnable {  
    int id;  
  
    Test(int id) {  
        this.id = id;  
    }  
  
    public static void main(String[] args) throws InterruptedException {  
        Thread thread1 = new Thread(new Test(1));  
        Thread thread2 = new Thread(new Test(2));  
        thread1.run();  
        thread2.start();  
        System.out.print("main");  
    }  
  
    public void run() {  
        System.out.print(id);  
    }  
}
```

Please choose all the answers that apply:

- 12main
- 21main
- 2main1
- 1main2
- compilation error, cannot invoke run() directly on thread1
- compilation error, cannot invoke start() directly on thread2

Check the answer of this question online on JavaChamp.com: [run vs start in threads in java](#)

1.19.31. How to create a Thread object?

Author: Yasser Ibrahim

Thread objects are created by:

Please choose all the answers that apply:

- direct instantiation from java.lang.Thread class
- instantiation from a class which implements the java.lang.Thread class
- instantiation a Runnable object (from a class which implements the Runnable interface)
- instantiation a Runnable object (from a class which implements the Runnable interface) and passing this Runnable object to a Thread constructor.

Check the answer of this question online on JavaChamp.com: [How to create a Thread object?](#)

1.19.32. handling a checked exception

Author: Java Champ

What is the output of compiling and running the following code?

```
public class Test extends Thread {
    static int count = 0;

    public static void main(String argv[]) {
        Test t = new Test ();
        t.increment(count);
        t.start();
        Thread.sleep(1000);
        System.out.println(count);
    }

    public void increment(int count) {
        ++count;
    }

    public void run() {
        count = count + 5;
    }
}
```

Please choose only one answer:

- 5
- 6
- Compilation error

Check the answer of this question online on JavaChamp.com: [handling a checked exception](#)

Author: Java Champ

What are the possible results of compiling and running the following code? (choose two)

```
public class Test extends Thread {  
    int id;  
  
    Test(int id) {  
        this.id = id;  
        start();  
    }  
  
    public static void main(String[] args) {  
        Thread t = new Thread(new Test(2));  
        t.start();  
        System.out.print("&quot;main&quot;");  
    }  
  
    public void run() {  
        System.out.print(id);  
    }  
}
```

Please choose all the answers that apply:

- main22
- 0main2
- main02
- 2main2
- compilation error for calling start twice
- InterruptedException is thrown for calling start twice

Check the answer of this question online on JavaChamp.com: [thread in java](#)

20. Enums

Exam Category Description and Objectives

1.20.1. comparing enums in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Test{  
    enum BookStatus {  
        PUBLISHED, DRAFT  
    }  
  
    public static void main(String[] args) {  
        BookStatus s1 = BookStatus.PUBLISHED;  
        BookStatus s2 = BookStatus.PUBLISHED;  
        System.out.print(s1 == BookStatus.PUBLISHED);  
        System.out.print(s1.equals(s2));  
    }  
}
```

Please choose only one answer:

- false>true
- true>true
- false>false
- compilation error, s1 and s2 are incorrectly instantiated

Check the answer of this question online on JavaChamp.com: [comparing enums in java](#)

1.20.2. how to declare java enum?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
enum Status {  
    Published(1), Rejected(2); // Line 2  
    int index; // Line 3  
    Status(int i) { index = i; }  
        int getIndex(){return index;} // Line 5  
}  
  
public class Tester {  
    public static void main(String [] args) {  
        System.out.println(Status.Published.index + Status.Rejected.index); // Line 10  
    }  
}
```

Please choose only one answer:

- Compilation error at line 2. Must be terminated by a colon ','
- Compilation error at line 3. index must be marked static
- Compilation error at line 5. must be marked public
- Compilation error at line 10, can't access index directly, must use getIndex() instead
- 3

Check the answer of this question online on JavaChamp.com: [how to declare java enum?](#)

1.20.3. Where java enum can be defined?

Author: Yasser Ibrahim

enum can be declared and defined inside :

Please choose all the answers that apply:

- Another enum
- A class
- An interface
- A method

Check the answer of this question online on JavaChamp.com: [Where java enum can be defined?](#)

21. Data Types

Exam Category Description and Objectives

1.21.1. Primitive Data types in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    static void test(float x) {  
        System.out.print("float");  
    }  
  
    static void test(double x) {  
        System.out.print("double");  
    }  
  
    public static void main(String[] args) {  
        test(99.9);  
    }  
}
```

Please choose only one answer:

- float
- double
- Compilation error
- Exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [Primitive Data types in java](#)

1.21.2. compound operator in typecasting

Author: pankaj gupta

which of the followings are correct when tried to compile and execute the below code ?

```
class s
{
public static void main(String agr[])
{
    short s1=4;    //LINE 1
    short s2 = s1+=s1; //LINE 2
    short s3= s1+s2; //LINE 3
    byte b1=(byte)s1 +(byte)s2; //LINE 4
    byte b2=(byte)((byte)s1 +(byte)(byte)s2); //LINE 5
}
}
```

Please choose all the answers that apply:

- compile time error at LINE 1
- compile time error at LINE 2
- compile time error at LINE 3
- compile time error at LINE 4
- compile time error at LINE 5
- compiles successfully.

Check the answer of this question online on JavaChamp.com: [java type casting](#)

1.21.3. float data type in java

Author: Java Champ

What is the result of compiling and running the following code?

```
public class Tester {  
    static void test(float x) {  
        System.out.print("float");  
    }  
  
    static void test(double x) {  
        System.out.print("double");  
    }  
  
    public static void main(String[] args) {  
        test((float) 99.9);  
    }  
}
```

Please choose only one answer:

- float
- double
- Compilation error
- Exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [float data type in java](#)

1.21.4. Data types in java

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
public class Tester {  
    static void test(float x) {  
        System.out.print("float");  
    }  
  
    public static void main(String[] args) {  
        test(99.9);  
    }  
}
```

Please choose only one answer:

- float
- Compilation fails
- An Exception is thrown at run time

Check the answer of this question online on JavaChamp.com: [Data types in java](#)

22. Static and init blocks

Exam Category Description and Objectives

1.22.1. In what order static and init blocks get called in java?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
class Bird {
    static {
        System.out.print("static1 ");
    }

    {
        System.out.print("init3 ");
    }

    public Bird() {
        System.out.print("Bird ");
    }

    static {
        System.out.print("static2 ");
    }
}

public class Falcon extends Bird {

    Falcon(){
        System.out.print("Falcon ");
    }
    public static void main(String[] args) {
        System.out.print("main ");
        new Falcon();
    }
}
```

Please choose only one answer:

- main static1 static2 init3 Bird Falcon
- main static1 static2 Falcon init3 Bird
- main static1 static2 Falcon Bird init3
- static1 static2 main init3 Bird Falcon
- static1 static2 main Bird init3 Falcon

Check the answer of this question online on JavaChamp.com: [In what order static and init blocks get called in java?](#)

23. Serialization

Exam Category Description and Objectives

1.23.1. How to combine inheritance and serialization in java?

Author: Java Champ

What is the result of compiling and running the following code?

```
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.io.Serializable;

class Base {
    int code = 99;

    Base() {
        code = 77;
    }
}

public class Derived extends Base implements Serializable {

    public static void main(String[] args) {
        Derived derived1 = new Derived();
        derived1.code = 88;

        try {
            FileOutputStream fos = new FileOutputStream("data.txt");
            ObjectOutputStream os = new ObjectOutputStream(fos);
            os.writeObject(derived1);
            os.close();
            FileInputStream fis = new FileInputStream("data.txt");
            ObjectInputStream is = new ObjectInputStream(fis);
            Derived derived2 = (Derived) is.readObject();
            System.out.print(derived2.code);
            is.close();
        } catch (Exception x) {

        }

    }
}
```

Please choose only one answer:

- 0
- 77
- 88
- 99

- Compilation error

Check the answer of this question online on JavaChamp.com: [How to combine inheritance and serialization in java?](#)

1.23.2. How to serialize java class?

Author: Yasser Ibrahim

Given the following two classes (Address and Account), what may happen if you attempt to serialize an instance of Account?

```
class Address{}  
class Account implements Serializable {Address address;}
```

Please choose only one answer:

- Compilation error
- Exception is thrown at run time
- Serialization will succeed correctly with no problems

Check the answer of this question online on JavaChamp.com: [How to serialize java class?](#)

1.23.3. What cannot be serialized in java?

Author: Yasser Ibrahim

What is the result of compiling and running the following code?

```
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.io.Serializable;

public class Tester implements Serializable{

    transient int x = 5;

    public static void main(String[] args) {
        Tester tester1 = new Tester();
        try {
            ObjectOutputStream os = new ObjectOutputStream(
                new FileOutputStream("data.txt"));
            os.writeObject(tester1);
            os.close();
            System.out.print(++tester1.x + " ");
            ObjectInputStream is = new ObjectInputStream(new FileInputStream(
                "data.txt"));
            Tester tester2 = (Tester) is.readObject();
            is.close();
            System.out.println(tester2.x);
        } catch (Exception x) {
            System.out.println("Exception thrown");
        }
    }
}
```

Please choose only one answer:

- 5 5
- 6 5
- 6 0
- An exception is thrown

Check the answer of this question online on JavaChamp.com: [What cannot be serialized in java?](#)

1.23.4. What variables cannot be serialized in java?

Author: [Java Champ](#)

Which of the following data members of a class cannot be serialized?

Please choose all the answers that apply:

- private data members
- transient data members
- static data members
- final data members

Check the answer of this question online on [JavaChamp.com](#): [What variables cannot be serialized in java?](#)

