Started on	Monday, 3 March 2025, 3:06 PM
State	Finished
Completed on	Monday, 3 March 2025, 3:22 PM
Time taken	16 mins 16 secs
Marks	9.67/15.00
Grade	**64.44** out of 100.00

Question 1

Complete

Mark 0.00 out of 1.00

What will be the output of the following code?

```
Integer a = 100;
Integer b = 100;
System.out.println(a == b);
```

- a. Runtime exception
- b. false
- c. true
- d. Compilation error

Question 2

Complete

Mark 1.00 out of 1.00

Which statements about wrapper class immutability are true?

- a. A new object is created when changing the value
- b. Wrapper objects cannot be modified once created
- c. Wrapper objects allow direct value modifications
- d. Wrapper objects use the same reference for all values

Question 3

Complete

Mark 0.67 out of 1.00

Which method retrieves the primitive value from a wrapper object?

- a. parseValue()
- b. intValue()
- c. booleanValue()
- d. doubleValue()

Question 4

Complete

Mark 0.50 out of 1.00

What happens when using == to compare two wrapper objects?

- a. It throws an exception
- b. It checks for reference equality
- c. It may return true for small cached values (-128 to 127)
- d. It always compares the values inside the wrapper

Question 5

Not answered

Marked out of 1.00

Which statements about the Boolean wrapper class are true?

- a. Boolean.valueOf("true") returns true
- b. It has a constructor that accepts a String
- c. It supports parseBoolean() returning a wrapper
- d. The Boolean class is mutable

Question 6

Not answered

Marked out of 1.00

Which statements about parseXxx() and valueOf() are true?

- a. valueOf() can return a primitive if needed
- b. parseXxx() and valueOf() always return the same type
- c. parseXxx() returns a primitive type
- d. valueOf() returns a wrapper object

Question 7

Complete

Mark 1.00 out of 1.00

Which of the following statements about the toString() method in wrapper classes are correct?

- a. It returns a string representation of the wrapped value
- b. It always returns a hexadecimal representation of the value
- c. It cannot be called on a wrapper object
- d. It is inherited from the Object class and overridden in wrapper classes

Question 8

Complete

Mark 1.00 out of 1.00

Which of the following are valid conversions using wrapper classes?

- a. Character c = Character.valueOf("c");
- b. Integer i = Integer.valueOf("42");
- c. Long l = Long.parseLong("1010", 2);
- d. Double d = Double.valueOf("3.14");

Question 9

Complete

Mark 1.00 out of 1.00

What are the primary purposes of wrapper classes?

- a. To enable direct file operations
- b. To enhance performance over primitive types
- c. To provide utility functions for primitive types
- d. To convert primitive types into objects

Question 10

Complete

Mark 1.00 out of 1.00

Which statements about autoboxing are correct?

- a. Java automatically converts a primitive type to its corresponding wrapper class
- b. Autoboxing is required for every primitive type conversion
- c. Java automatically converts a wrapper class object to its corresponding primitive type
- d. Autoboxing only works with integer types

Question 11

Complete

Mark 1.00 out of 1.00

Which methods convert a string into a wrapper class instance?

- a. Double.parseDouble("3.14")
- b. Boolean.valueOf("true")
- c. Character.parseChar("c")
- d. Integer.valueOf("123")

Question 12

Complete

Mark 1.00 out of 1.00

Which of the following are wrapper classes in Java?

- [x] a. Double
- [x] b. Integer
- [x] c. Boolean
- [x] d. String

Question 13

Complete

Mark 0.50 out of 1.00

Which are valid ways to create an Integer object?

- [x] a. Integer i = new Integer(10);
- [x] b. Integer i = Integer.toInteger(10);
- [x] c. Integer i = Integer.parseInteger(10);
- [ ] d. Integer i = Integer.valueOf(10);

Question 14

Not answered

Marked out of 1.00

Which of the following statements about boxing and unboxing are true?

- [ ] a. Unboxing converts a wrapper object to its corresponding primitive
- [ ] b. Boxing automatically wraps a primitive into its corresponding wrapper class
- [ ] c. Java does not allow mixing boxed and unboxed values in expressions
- [ ] d. Unboxing is always explicit

Question 15

Complete

Mark 1.00 out of 1.00

Which constructor calls are valid for wrapper classes?

- [x] a. Float f = new Float(3.14f);
- [ ] b. Boolean b = new Boolean(1);
- [ ] c. Character c = new Character("c");
- [x] d. Integer i = new Integer(42);