Started on	Thursday, 5 June 2025, 11:32 AM
State	Finished
Completed on	Thursday, 5 June 2025, 12:08 PM
Time taken	36 mins 27 secs
Marks	39.00/50.00
Grade	**78.00** out of 100.00

Question 1

Complete

Mark 1.00 out of 1.00

What will be the output of the following code?

```
class A {
    int x = 10;
    A() {
        print();
    }
    void print() {
        System.out.println("A: " + x);
    }
}
class B extends A {
    int x = 20;
    void print() {
        System.out.println("B: " + x);
    }
}
public class Main {
    public static void main(String[] args) {
        A obj = new B();
    }
}
```

○ a. A: 10

○ b. Runtime Error

● c. B: 0

○ d. B: 20

Question 2

Complete

Mark 1.00 out of 1.00

What is the result of the following code?

```
interface I1 {
    default void display() {
        System.out.println("I1");
    }
}
interface I2 {
    default void display() {
        System.out.println("I2");
    }
}
class C implements I1, I2 {
    public void display() {
        I1.super.display();
    }
}
public class Test {
    public static void main(String[] args) {
        new C().display();
    }
}
```

● a. I1

○ b. I2

○ c. Compilation error

○ d. I1 I2

Question 3

Complete

Mark 0.00 out of 1.00

What will this code print?

```
class Super {
    static void method() {
        System.out.println("Super");
    }
}
class Sub extends Super {
    static void method() {
        System.out.println("Sub");
    }
}
public class Demo {
    public static void main(String[] args) {
        Super obj = new Sub();
        obj.method();
    }
}
```

○ a. Compilation error

● b. Sub

○ c. Runtime error

○ [✓ d. Super]

static methods are called using the reference type

Question 4

Complete

Mark 1.00 out of 1.00

Which one is not allowed in Java?

● a. Final class with abstract methods
○ b. Abstract class with constructor
○ c. A class with both abstract and non-abstract methods
○ d. Abstract class with static methods

Question 5

Complete

Mark 0.00 out of 1.00

What is the output?

```
class Test {
    private void display() {
        System.out.println("Private");
    }
}
class Sub extends Test {
    public void display() {
        System.out.println("Public");
    }
}
public class Demo {
    public static void main(String[] args) {
        Test t = new Sub();
        t.display();
    }
}
```

○ a. Runtime error
○ b. Private
✓ ○ c. Compilation error
● d. Public

private methods cannot be inherited

Question 6

Complete

Mark 1.00 out of 1.00

What is the output of the following code?

```
class Base {
    void show() {
        System.out.println("Base show()");
    }
}
class Derived extends Base {
    void show(int x) {
        System.out.println("Derived show(" + x + ")");
    }
}
public class Main {
    public static void main(String[] args) {
        Base b = new Derived();
        b.show();
    }
}
```

○ a. Runtime error

● b. Base show()

○ c. Derived show()

○ d. Compilation error

Question 7

Complete

Mark 1.00 out of 1.00

What is the output?

```
class A {
    int i = 10;
    A() {
        System.out.println(i);
        i = 20;
    }
}
public class Test {
    public static void main(String[] args) {
        A a = new A();
        System.out.println(a.i);
    }
}
```

○ a. 0 10

● b. 10 20

○ c. 20 20

○ d. 10 10

Question 8

Complete

Mark 1.00 out of 1.00

What is true about constructors in Java?

○ a. Constructors can be inherited

○ b. Constructors must be public

○ c. Constructor name can be different from class name

● d. A constructor can call another constructor in the same class using this()

Question 9

Complete

Mark 1.00 out of 1.00

What will be the output?

```
class Animal {
    void sound() { System.out.println("Generic sound"); }
}
class Dog extends Animal {
    void sound() { System.out.println("Bark"); }
}
public class Main {
    public static void main(String[] args) {
        Animal a = new Dog();
        a.sound();
    }
}
```

○ a. Compilation error

● b. Bark

○ c. Runtime error

○ d. Generic sound

Question 10

Complete

Mark 1.00 out of 1.00

Which of the following is not true for method overriding?

● a. Access modifier can be more restrictive

○ b. Method must be inherited

○ c. Return type must be same or subtype

○ d. Method name must be same

Question 11

Complete

Mark 1.00 out of 1.00

What will happen?

```
class A {
    final void show() {}
}
class B extends A {
    void show() {}
}
```

● a. Compilation error

○ b. Compiles and runs

○ c. Runtime error

○ d. Shows nothing

Question 12

Complete

Mark 1.00 out of 1.00

What is the output?

```
try {
    throw new IllegalArgumentException("Illegal");
} catch (IllegalArgumentException e) {
    System.out.println(e.getMessage());
}
```

○ a. Exception

○ b. null

○ c. IllegalArgumentException

● d. Illegal

Question 13

Complete

Mark 1.00 out of 1.00

What is true about finally block?

○ a. It executes only if no exception occurs

○ b. It must be used with catch

○ c. It is optional

● d. It always executes regardless of exceptions

Question 14

Complete

Mark 1.00 out of 1.00

What will be printed?

```
try {
    int[] arr = new int[5];
    arr[5] = 100;
} catch (ArrayIndexOutOfBoundsException e) {
    System.out.print("Caught ");
} finally {
    System.out.print("Finally");
}
```

● a. Caught Finally
○ b. Caught
○ c. Finally
○ d. Runtime Error

Question 15

Complete

Mark 1.00 out of 1.00

What happens if catch block is missing?

```
try {
    System.out.println(10 / 0);
} finally {
    System.out.println("Done");
}
```

● a. ArithmeticException is thrown after finally
○ b. Compilation error
○ c. Program compiles but doesn't run
○ d. Done

Question 16

Complete

Mark 1.00 out of 1.00

What is the output of the following code?

```
try {
    throw new Exception("Check");
} catch (RuntimeException e) {
    System.out.println("Runtime");
} catch (Exception e) {
    System.out.println("Exception");
}
```

○ a. Compilation error
● b. Exception
○ c. Runtime
○ d. Runtime error

Question 17

Complete

Mark 1.00 out of 1.00

Which of the following is a checked exception?

○ a. NullPointerException

○ b. ArithmeticException

○ c. IllegalArgumentException

● d. FileNotFoundException

Question 18

Complete

Mark 1.00 out of 1.00

What is the output?

```
try {
    int a = 5 / 0;
} catch (Exception e) {
    System.out.print("Catch ");
} finally {
    System.out.print("Finally");
}
```

● a. Catch Finally

○ b. Catch

○ c. Finally

○ d. Runtime Error

Question 19

Complete

Mark 0.00 out of 1.00

What happens if exception is thrown in finally block?

✓ ○ a. Original exception is suppressed

● b. Compile-time error

○ c. finally block never throws exception

○ d. Both exceptions are printed

Question 20

Complete

Mark 1.00 out of 1.00

What is the output?

```
public class Test {
    public static void main(String[] args) {
        try {
            System.out.println("Try");
            return;
        } finally {
            System.out.println("Finally");
        }
    }
}
```

○ a. Compilation error

○ b. Finally

○ c. Try

● d. Try Finally

Question 21

Complete

Mark 0.00 out of 1.00

What is the output of the following code?

```
LocalDate date = LocalDate.of(2024, Month.FEBRUARY, 29);
System.out.println(date.plusYears(1));
```

[✓ ○ a. 2025-02-28] tries to keep month and year as close as possible

● b. 2025-03-01

○ c. 2025-02-29

○ d. DateTimeException

Question 22

Complete

Mark 1.00 out of 1.00

What does this print?

```
LocalTime time = LocalTime.of(23, 59, 59);
System.out.println(time.plusSeconds(2));
```

● a. 00:00:01

○ b. 23:59:61

○ c. 00:01:00

○ d. 00:00

Question **23**

Complete

Mark 1.00 out of 1.00

Which class would you use to represent a date and time with time zone?

- ○ a. Instant
- ○ b. OffsetDateTime
- ● c. ZonedDateTime
- ○ d. LocalDateTime

Question **24**

Complete

Mark 1.00 out of 1.00

What will this code output?

```
DateTimeFormatter formatter = DateTimeFormatter.ofPattern("yyyy/MM/dd");
LocalDate date = LocalDate.parse("2023/08/15", formatter);
System.out.println(date);
```

- ○ a. 2023/08/15
- ○ b. Compilation Error
- ○ c. August 15, 2023
- ● d. 2023-08-15

Question **25**

Complete

Mark 0.00 out of 1.00

What is the output?

```
Instant instant = Instant.now();
ZonedDateTime zdt = instant.atZone(ZoneId.of("UTC"));
System.out.println(zdt.getOffset());
```

- ○ a. Throws Exception
- ● b. System timezone offset
- ○ [✓ c. +00:00]
- ○ d. null

Question **26**

Complete

Mark 1.00 out of 1.00

What does the JVM use the heap memory for?

- ● a. Object allocation
- ○ b. Thread-local storage
- ○ c. Stack frames
- ○ d. Method calls

Question 27

Complete

Mark 1.00 out of 1.00

What happens when an object becomes unreachable in Java?

- ● a. It becomes eligible for GC
- ○ b. JVM throws NullPointerException
- ○ c. StackOverFlowError occurs
- ○ d. It's immediately deleted

Question 28

Complete

Mark 0.00 out of 1.00

Which generation in GC typically contains short-lived objects?

- ● a. Old Generation
- ○ b. Tenured
- ○ [✓ c. Eden Space]
- ○ d. PermGen

Question 29

Complete

Mark 1.00 out of 1.00

What is true about finalize() in Java?

- ○ a. It guarantees memory cleanup
- ○ b. It is always called before GC
- ● c. It may or may not be called
- ○ d. It is used for performance

Question 30

Complete

Mark 1.00 out of 1.00

What is the role of System.gc()?

- ○ a. Immediately triggers GC
- ● b. Requests GC but does not force it
- ○ c. Prevents GC
- ○ d. Deletes static data

Question 31

Complete

Mark 1.00 out of 1.00

What is the output?

```
public class Test extends Thread {
    public void run() {
        System.out.println("Thread running");
    }
    public static void main(String[] args) {
        Test t = new Test();
        t.run();
        t.start();
    }
}
```

○ a. Only one "Thread running"

● b. Thread running Thread running

○ c. Compilation error

○ d. Runtime error

Question 32

Complete

Mark 1.00 out of 1.00

Which method causes the current thread to wait until another completes?

● a. join()

○ b. wait()

○ c. yield()

○ d. sleep()

Question 33

Complete

Mark 1.00 out of 1.00

What will this code output?

```
public class Main {
    public static void main(String[] args) {
        Thread t = new Thread(() -> {
            for (int i = 0; i < 2; i++) {
                System.out.print(Thread.currentThread().getName() + " ");
            }
        });
        t.setName("Worker");
        t.start();
    }
}
```

● a. Worker Worker

○ b. Main Main

○ c. Compiler Error

○ d. Thread-0 Thread-0

Question 34

Complete

Mark 1.00 out of 1.00

What does this print?

```
class MyThread extends Thread {
    public void run() {
        System.out.print("Hello ");
    }
    public static void main(String[] args) {
        MyThread t1 = new MyThread();
        t1.start();
        t1.start();
    }
}
```

● a. Runtime Exception

○ b. Compilation Error

○ c. Hello

○ d. Hello Hello

Question 35

Complete

Mark 0.00 out of 1.00

Which of the following is true about thread priorities?

● a. Priority affects the order of execution deterministically

○ b. Higher priority means more CPU time guaranteed

○ [✓ c. Thread priority is a hint to the thread scheduler]

○ d. Priorities are always respected by JVM

Question 36

Complete

Mark 0.00 out of 1.00

What does this code print?

```
import java.util.concurrent.*;
public class Main {
    public static void main(String[] args) {
        ExecutorService service = Executors.newFixedThreadPool(2);
        service.submit(() -> System.out.println("Task 1"));
        service.submit(() -> System.out.println("Task 2"));
        service.shutdown();
    }
}
```

○ a. Compilation error

● b. Task 1 Task 2

○ c. Task 2 Task 1

○ [✓ d. Any order of Task 1 and Task 2]

runs them sequentially or paralelly

Question 37

Complete

Mark 0.00 out of 1.00

What is the key feature of ReentrantLock over synchronized?

○ [✓ a. Fairness policy and tryLock() capability]

● b. No need to unlock manually

○ c. Simpler syntax

○ d. Can't be interrupted

Question 38

Complete

Mark 1.00 out of 1.00

```
What will happen here?
ConcurrentHashMap<String, Integer> map = new ConcurrentHashMap<>();
map.put("A", 1);
map.compute("A", (k, v) -> v + 1);
System.out.println(map.get("A"));
```

○ a. 1

○ b. NullPointerException

● c. 2

○ d. Compilation error

Question 39

Complete

Mark 1.00 out of 1.00

What is ForkJoinPool best suited for?

● a. Dividing tasks recursively and processing in parallel

○ b. Long I/O-bound tasks

○ c. UI event handling

○ d. Thread communication

Question 40

Complete

Mark 0.00 out of 1.00

What is the default parallelism level of ForkJoinPool?

○ [✓ a. Number of available processors]

○ b. Fixed at 4

○ c. 1

● d. Number of processors - 1

Question 41

Complete

Mark 1.00 out of 1.00

What will this code print?

```
List<String> list = Arrays.asList("apple", "banana", "cherry");
list.stream()
    .filter(s -> s.length() > 5)
    .map(String::toUpperCase)
    .forEach(System.out::print);
```

○ a. CHERRYBANANA
○ b. banana cherry
○ c. APPLEBANANACHERRY
● d. BANANACHERRY

Question 42

Complete

Mark 1.00 out of 1.00

What is the output of the following?

```
Stream.of(1, 2, 3, 4, 5)
      .filter(i -> i % 2 == 0)
      .map(i -> i * i)
      .findFirst()
      .ifPresent(System.out::print);
```

● a. 4
○ b. 16
○ c. 2
○ d. 1

Question 43

Complete

Mark 0.00 out of 1.00

Which operation is terminal in streams?

● a. peek()
✓ b. forEach()
○ c. filter()
○ d. map()

Question 44

Complete

Mark 1.00 out of 1.00

What is the result?

```
List<String> list = Arrays.asList("a", "bb", "ccc", "dd");
String result = list.stream()
    .filter(s -> s.length() == 2)
    .collect(Collectors.joining("-"));
System.out.println(result);
```

○ a. bb

○ b. a-bb-ccc-dd

● c. bb-dd

○ d. bbd

Question 45

Complete

Mark 1.00 out of 1.00

What will this output?

```
Stream<String> s = Stream.of("java", "lambda", "stream");
long count = s.map(String::length).filter(l -> l > 5).count();
System.out.println(count);
```

○ a. 0

○ b. 3

● c. 2

○ d. 1

Question 46

Complete

Mark 1.00 out of 1.00

What is the output?

```
List<String> list = new ArrayList<>();
list.add("a");
list.add("b");
list.add(1, "c");
System.out.println(list);
```

○ a. [c, a, b]

○ b. Compilation error

● c. [a, c, b]

○ d. [a, b, c]

Question 47

Complete

Mark 1.00 out of 1.00

Which collection is synchronized?

- ○ a. LinkedList
- ○ b. HashMap
- ● c. Vector
- ○ d. ArrayList

Question 48

Complete

Mark 1.00 out of 1.00

What does this print?

```
Map<String, String> map = new HashMap<>();
map.put("a", "apple");
map.put("b", "banana");
map.put("a", "avocado");
System.out.println(map.get("a"));
```

- ○ a. apple
- ○ b. banana
- ● c. avocado
- ○ d. null

Question 49

Complete

Mark 1.00 out of 1.00

What is true about HashSet?

- ● a. Uses hashCode and equals
- ○ b. Implements List
- ○ c. Allows duplicate elements
- ○ d. Maintains insertion order

Question 50

Complete

Mark 1.00 out of 1.00

Which Map maintains insertion order?

- ○ a. TreeMap
- ○ b. HashMap
- ○ c. Hashtable
- ● d. LinkedHashMap