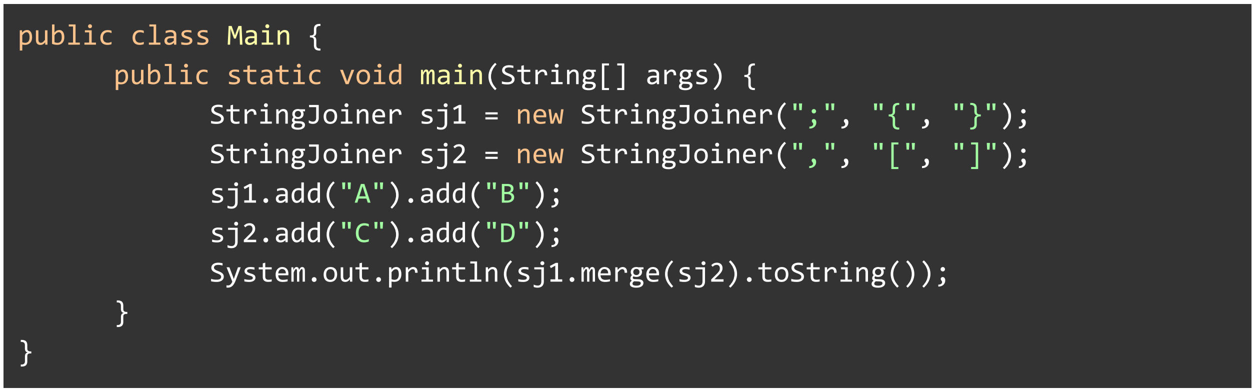
Core Java KBA Study Guide

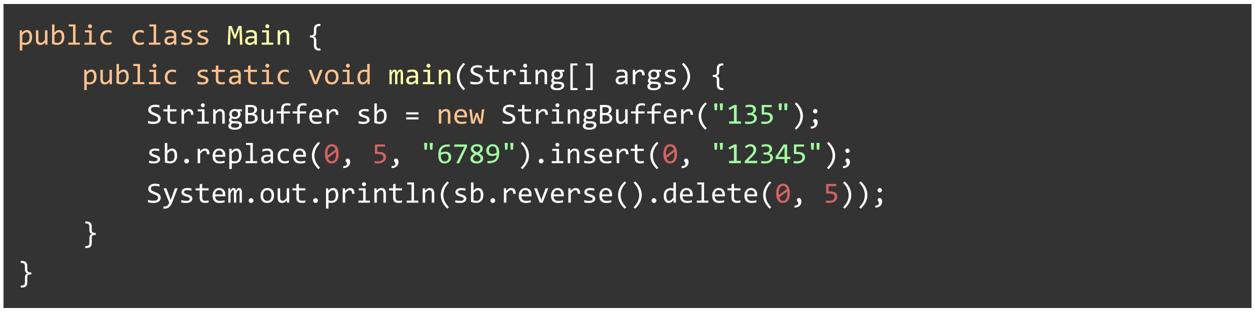
(All answer will be in bold)

1. Where do you define class variables? (In our course content, the term “class variables” is used to refer both static and instance variables/fields)
    - ○ **top of class definition**
    - ○ inside of a method
2. What is the data type you want to use for a customer name?
    - ○ char
    - ○ **String**
3. What is the difference between String class andStringBuffer class?
    - **String objects are immutable, thread-safe, and efficient for operations that don't require frequent modifications.**
    - **StringBuffer objects are mutable, not thread-safe, and efficient for operations that require frequent modifications.**
4. What would be the outcome after the following code is executed?
    - **{A;B;C,D}**



```java
public class Main {
      public static void main(String[] args) {
            StringJoiner sj1 = new StringJoiner(";", "{", "}");
            StringJoiner sj2 = new StringJoiner(",", "[", "]");
            sj1.add("A").add("B");
            sj2.add("C").add("D");
            System.out.println(sj1.merge(sj2).toString());
      }
}
```

5. What would be the outcome after the following code is executed?
    - **4 3 2 1**



```java
public class Main {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer("135");
        sb.replace(0, 5, "6789").insert(0, "12345");
        System.out.println(sb.reverse().delete(0, 5));
    }
}
```

6. A method with “protected” access modifier in a class A can be accessed by any other classes in the same package and any subclasses of the A even if the subclasses are in other packages? [True/False]
    - **TRUE**
7. What are the possible Java keywords that could be used in a class declaration?
    - **public, abstract, final, class, private, protected, static, extends, implements**
8. Inheritance is a HAS-A relationship while composition is IS-A relationship. [True/False]
    - **FALSE**

9. A method has to have at least one parameter. [True/False]
    - **FALSE**
10. A method declaration has to have parentheses () with or without a parameter list. [True/False]
    - **TRUE**
11. What is method overloading?
    - **Creating the same method with different number of parameters and behavior**
12. Java does support multiple inheritance for classes. [True/False]
    - **False, one class can only inherit from one other class**
13. A “static” class variable belong to the class not to the instances of the class. [True/False]
    - **True**
14. How many copies of static class variables and instance variables when 6 object instances are created of a class?
    - **1 copy of the static class variable and 6 copies of the instance variable**
15. What is “autoboxing”? What is “unboxing”?
    - **Autoboxing is converting primitive to wrapper class.**
    - **Unboxing is converting wrapper class to primitive data type.**
16. “inheritance” refers to an ability by which one class acquires the properties and behaviors of another class. [True/False]
    - **True**
17. The extension of Java source code files is “.class”. [True/False]
    - **False, it is ".java".**
    - **Java byteCode is ".class".**
18. A constructor can have a return type. [True/False]
    - **False**
19. A truncation for numeric value occurs when floating-point value is assigned to integer type. [True/False]
    - **True**
20. What should you do when a primitive “int” data type is required as a corresponding object? Choose one from below:
    - ○ Create a class containing that int as its only field
    - ○ **Cast it to its corresponding wrapper class**
21. What is the complete syntax of a “main” method?
    - **public static void main(String[] args){}**
22. What is a valid syntax? (Choose one)
    - ○ **Desk desk = new Desk();**
    - ○ desk = new Desk(); // There is no variable of “desk” defined until this code
23. What is the difference between “do-while” and “while” loop?
    - **"do-while" loops run the block of code inside the loop and then it checks the condition to see if it should run again**
    - **"while" loops check the condition to see if they should run and then if the condition is met, they run the block of code**
24. A class member (variable or method) with a “private” access modifier can be accessed directly by a class in the same package. [True/False]
    - **False**
25. Write code to declare and initialize an “int” array of size 10?
    - **int[] nums = new in[10];**
26. You use “new” keyword to create an array object instance. [True/False]
    - **True**

27. What would be the output of executing the following code?

- **ABC10DEF20PQR30**

```
class A {

    void ETL() {
        String E = "i";
        String T = "love";
        String L = "java";
        System.out.println(E + T + L);
    };

    void ETL(String ETL) {
        System.out.println(ETL.toUpperCase());
    }
    void ETL(String E, String T, String L) {
        System.out.println(E + T + L);
    }
}

public class Main {

    public static void main(String args[]) {
        String E = "abc10";
        String T = "def20";
        String L = "pqr30";

        A obj = new A();
        obj.ETL(E + T + L);
    }
}
```

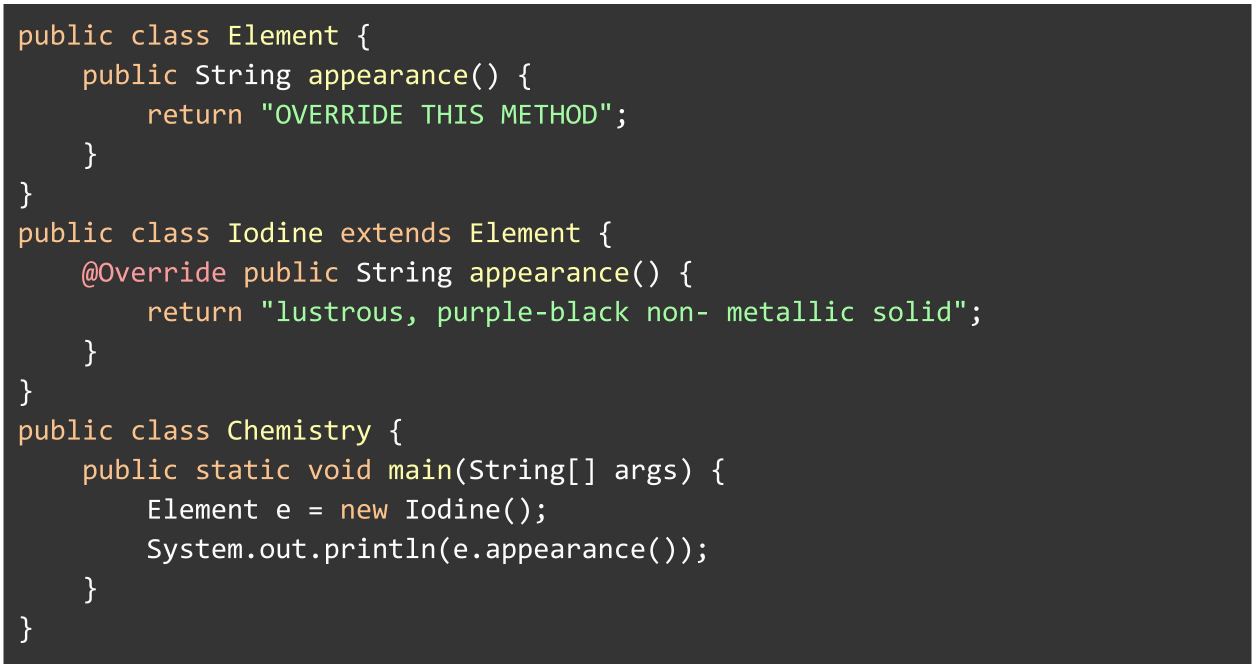
What would be the output of executing the following code?

- **Compile error, cannot override final declared method**

```
class A {
    String E = "extraction ";
    String T = "transformation ";
    String L = "loading ";

    final void ETL() {
        System.out.println(L + T + E);
    };
}

class B extends A {
    void ETL() {
        System.out.println(E + T + L);
    }
}

public class Main {
    public static void main(String args[]) {
        A obj = new B();
        obj.ETL();
    }
}
```

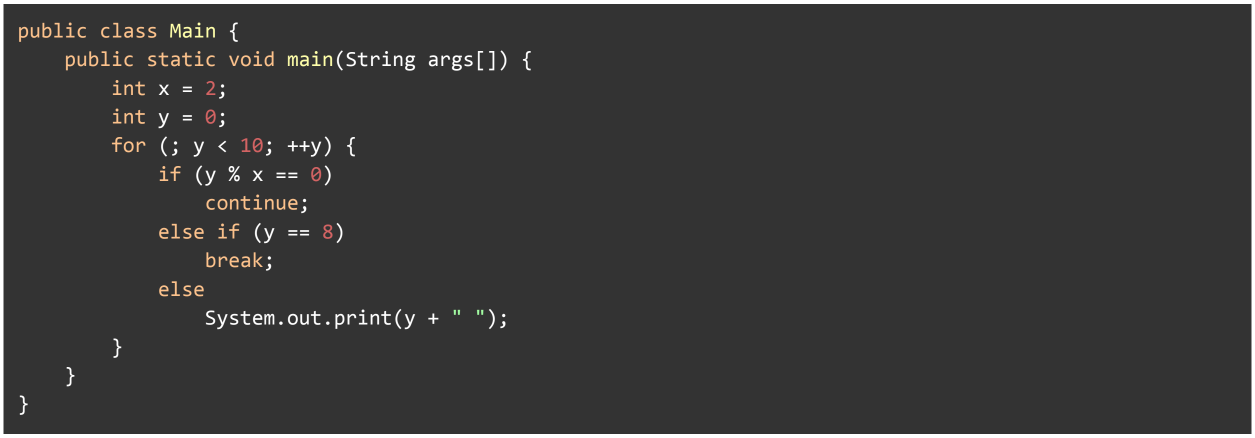
28. “polymorphism” in object-oriented programming refers to the ability of a variable, function or object to take multiple forms. [True/False]
    - **True**

29. What would be the output of the following code?
    - **LUSTROUS, PURPLE-BLACK NON-METALLIC SOLID**



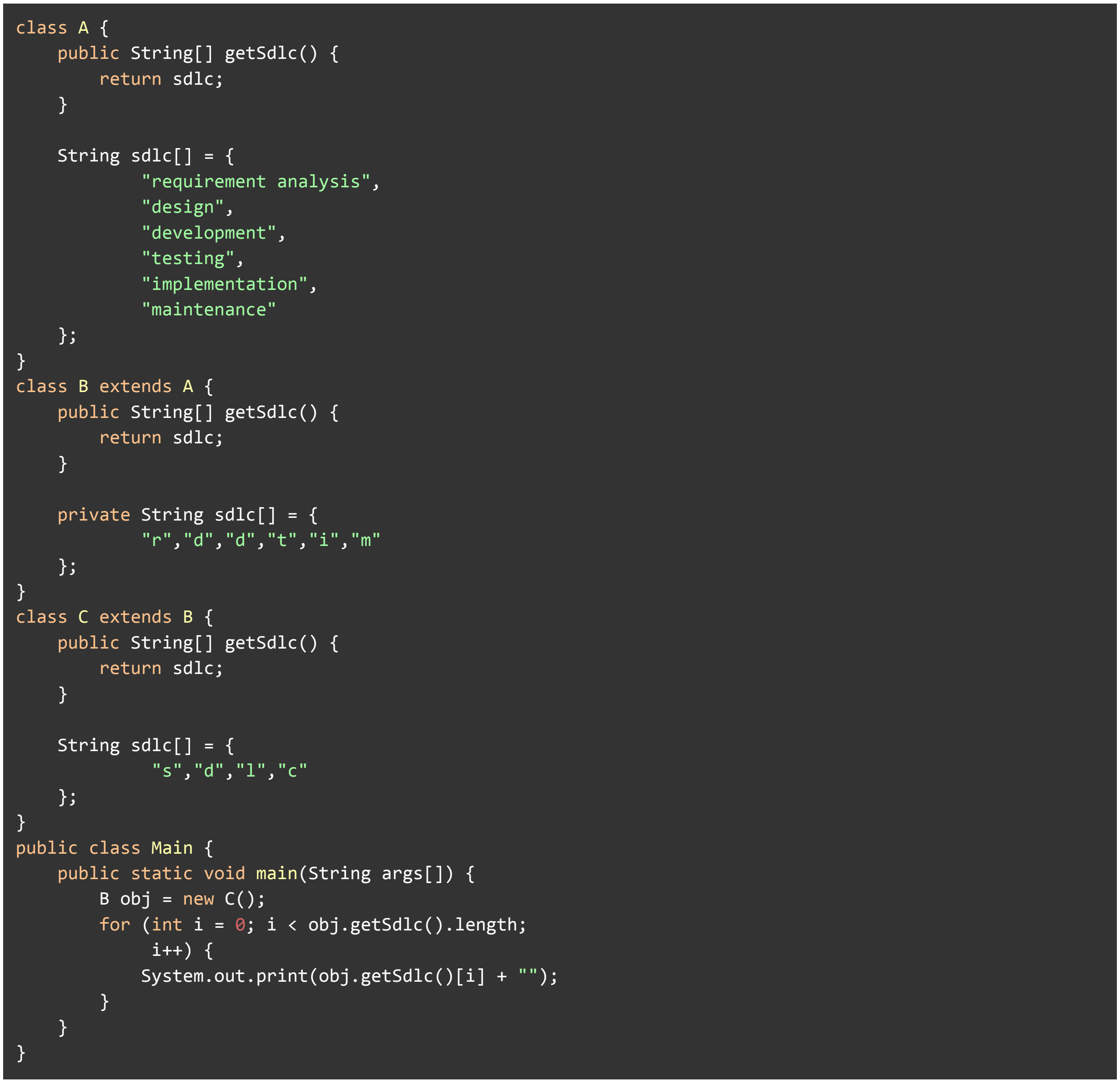
```
public class Element {
    public String appearance() {
        return "OVERRIDE THIS METHOD";
    }
}
public class Iodine extends Element {
    @Override public String appearance() {
        return "lustrous, purple-black non- metallic solid";
    }
}
public class Chemistry {
    public static void main(String[] args) {
        Element e = new Iodine();
        System.out.println(e.appearance());
    }
}
```

30. What would be the result of executing the following code?
    - **13579**



```
public class Main {
    public static void main(String args[]) {
        int x = 2;
        int y = 0;
        for (; y < 10; ++y) {
            if (y % x == 0)
                continue;
            else if (y == 8)
                break;
            else
                System.out.print(y + " ");
        }
    }
}
```

31. What would be the outcome of the following program?
    - **sdlc**



```
class A {
    public String[] getSdlc() {
        return sdlc;
    }

    String sdlc[] = {
            "requirement analysis",
            "design",
            "development",
            "testing",
            "implementation",
            "maintenance"
    };
}
class B extends A {
    public String[] getSdlc() {
        return sdlc;
    }

    private String sdlc[] = {
            "r","d","d","t","i","m"
    };
}
class C extends B {
    public String[] getSdlc() {
        return sdlc;
    }

    String sdlc[] = {
             "s","d","l","c"
    };
}
public class Main {
    public static void main(String args[]) {
        B obj = new C();
        for (int i = 0; i < obj.getSdlc().length;
             i++) {
            System.out.print(obj.getSdlc()[i] + "");
        }
    }
}
```

32. Abstract classes must have at least one abstract method. [True/False]
    - **TRUE**
33. Abstract classes can have constructors. [True/False]
    - **TRUE**
34. You can create an object instance from an abstract class. [True/false]
    - **FALSE**
35. Interfaces can have constructors. [True/False] //->
    - **FALSE**
36. You can create object instance from an Interface. [True/False]
    - **FALSE**
37. What are valid values that can be assigned to a boolean variable?
    - **TRUE OR FALSE**

38. In general, a typical interface only has methods but you can declare fields in an interface - the fields are implicitly declared static and final. [True/False]
    - **TRUE**