



Vault of Codes

Assignment 1: Answers

Task1:

Review the following codes, find and fix errors also explain the errors

Assignment 1: Java Code Review and Error Correction

1. Code Snippet 1

```
public class Main {  
  
    public static void main(String[] args) {  
  
        System.out.println("Hello, World!");  
  
    }  
  
}
```

Errors and Fixes

1. **Missing semicolon:** Added ; after `System.out.println("Hello, World!");`.
2. **Improper formatting:** Fixed indentation for better readability.

2. Code Snippet 2

```
public class Main {  
  
    public void greet() {  
  
        System.out.println("Hello");  
  
    }  
  
    public static void main(String[] args) {  
  
        Main main = new Main();  
  
        main.greet();  
  
    }  
  
}
```

Errors and Fixes

1. `greet()` **called outside a method**: Moved it into the `main` method.
 2. **No `main` method**: Added a `main` method as the program's entry point.
 3. **Non-static method call**: Created an instance of `Main` to call the non-static `greet()` method.
-

3. Code Snippet 3

```
public class Main {  
  
    public static void main(String[] args) {  
  
        int number = 10;  
  
        System.out.println("The number is: " + number);  
  
    }  
  
}
```

Errors and Fixes

1. **Invalid assignment** (`int number = "10";`)

4. Code Snippet 4

```
public class Main {  
  
    public static void main(String[] args) {  
  
        int[] numbers = {1, 2, 3, 4};  
  
        System.out.println("The fourth element is: " + numbers[3]);  
  
    }  
  
}
```

Errors and Fixes

1. **Array index out of bounds (`numbers[4]`):**
 1. Arrays in Java are zero-indexed, so `numbers[4]` tries to access the fifth element, which doesn't exist (valid indices are 0 to 3).
 2. **Fix:** Changed `numbers[4]` to `numbers[3]` to access the last element in the array.

5. Code Snippet 5

```
public class Main {  
  
    public static void main(String[] args) {  
  
        Main main = new Main();  
  
        int result = main.addNumbers(5, 10);  
  
        System.out.println("Result: " + result);  
  
    }  
  
    public int addNumbers(int a, int b) {  
  
        return a + b;  
  
    }  
  
}
```

Errors and Fixes

1. Calling non-static method from `main`:

1. `addNumbers` is a non-static method, but `main` is static, so it can't call `addNumbers` directly.
2. **Fix:** Created an instance of `Main` to call the `addNumbers` method.

6. Code Snippet 6

```
public class Main {  
  
    public static void main(String[] args) {  
  
        int age = 18; // Initialize age with a value  
  
        if (age >= 18) {  
  
            System.out.println("You are eligible to vote.");  
  
        }  
  
    }  
  
}
```

Errors and Fixes

1. Uninitialized variable (`int age;`):

1. The variable `age` was declared but not initialized, causing a compilation error.
2. **Fix:** Initialized `age` with a value (`int age = 18;`).

7. Code Snippet 7

```
public class Main {  
  
    public static void main(String[] args) {  
  
        int i; // Declare 'i' outside the loop for access after the loop  
  
        for (i = 0; i < 5; i++) {  
  
            System.out.println("Number: " + i);  
  
        }  
  
        System.out.println("Outside loop: " + i); } }
```

Errors and Fixes

Scope issue with `i`:

1. `i` declared inside the `for` loop is not accessible outside it.
2. **Fix:** Declared `i` outside the loop to make it accessible after the loop.

Unnecessary `i++` outside the loop:

1. The loop already increments `i`. Incrementing again is redundant.
2. **Fix:** Removed the extra `i++` statement.

8. Code Snippet 8

```
public class Main {  
    public static void main(String[] args) {  
        int count = 0; // Initialize 'count'  
        while (count < 10) {  
            System.out.println("Count: " + count);  
            count++;  
        }  
    }  
}
```

Errors and Fixes

Missing variable declaration (`count`):

1. `count` was used without being initialized.
2. **Fix:** Declared and initialized `count` with `int count = 0;`.
- 2.

Syntax error in `while` condition:

1. The `while` loop condition was missing parentheses.
2. **Fix:** Added parentheses around the condition: `while (count < 10`