

2. Explain the difference between method declaration and method body

A method declaration defines the method's name, return type, and parameters. The method body contains the code that runs when the method is called.

3. What type of keyword is used to change the access level of a method?

The keyword used to change the access level of a method is an access modifier (e.g., public, private, protected).

4. What is another word used for describing the access level of a method?

Another word used to describe the access level of a method is visibility.

5. Explain the scope of each of the variables in the code below:

```
public class ScopeExample {  
    public static void main(String[] args) {  
        int var1;  
        for (int var2 = 0; var2 < 5; var2++) {  
            method1();  
        }  
    }  
    public static void method1() {  
        int var3;  
        for (int var4 = 0; var4 < 2; var4++) {  
            var3 += 1;  
        }  
    }  
}
```

- var1: Scope is the entire main method; it can be used anywhere in main.
- var2: Scope is limited to the for loop; it can't be used outside of the loop.
- var3: Scope is limited to method1; it can't be used outside of that method.
- var4: Scope is limited to the inner for loop in method1; it can't be used outside that loop.

6. Write a method declaration for each of the following descriptions: a) A class method named `getVowels` that can be called by any other method, requires a `String` parameter, and returns an integer value. b) A class method named `extractDigit` that can be called by any other method, requires an integer parameter, and returns an integer value. c) A class method named `insertString` that can be called by any other method, requires a `String` parameter and an integer parameter, and returns a `String` parameter.

a) `public static int getVowels(String str) {}`

b) `public static int extractDigit(int number) {}`

c) `public static String insertString(String str, int index) {}`

7. a) How does the compiler distinguish one method from another? b) Can two methods in the same class have the same name? Explain

a) The compiler distinguishes methods by their name, parameters (type and number), and return type (method signature).

b) Yes, two methods in the same class can have the same name if they have different parameters. This is called method overloading.

8. a) What is the return statement used for? b) How many values can a return statement send back to the calling statement? c) How is the declaration of a method returning a value different from the declaration of a method that does not return a value?

a) The return statement is used to send a value back to the code that called the method.

b) A return statement can send back only one value.

c) A method returning a value includes a return type in its declaration (e.g., `int`), while a method that does not return a value has a return type of `void`.

9. Find and explain the error in the code below: `public class MethodCallExample { public static void main(String[] args) { int num; doSomething(); num = dosomething();`

`}`

**`Public static int dosomething() {
return(5);
}`**

`]`

The error in the code is that the method `doSomething()` is called before it is defined, which will lead to a compilation error. Additionally, the method `dosomething()` is defined with a lowercase "d" but called with an uppercase "D," leading to a name mismatch. The correct declaration should be `public static int dosomething()`. Also, there is a syntax error with `Public` (should be lowercase `public`) and an unnecessary closing bracket `]` at the end.