CRT

2. Explain the difference between a method declaration and a method body.

The method declaration specifies all the method's capabilities, including its name, return type, and parameters. The method body is where the actual execution happens, containing the set of instructions that define the method's behavior.

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

The public keyword is used as an access modifier.

4. What is another term used to describe the access level of a method?

This is commonly referred to as the method's visibility.

5. Explain the scope of each variable 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;
        }
```

```
    }
}
```

In the given Java code, each variable's scope is defined by the block in which it is declared.

6. Write a method declaration for each of the following descriptions:

a) A class method named getVowels that can be accessed by any other method, accepts a String parameter, and returns an integer value:

```
public static int getVowels(String input) {
   // method implementation goes here
}
```

b) A class method named extractDigit that can be accessed by any other method, takes an integer parameter, and returns an integer value:

```
public static int extractDigit(int number) {
   // method implementation goes here
}
```

c) A class method named insertString that can be accessed by any other method, takes a String parameter and an integer parameter, and returns a String:

```
public static String insertString(String str, int index) {
   // method implementation goes here
}
```

7.

a) How does the compiler distinguish one method from another?

The compiler differentiates methods based on their method signatures, which include the method name and the parameter list (types and number of parameters).

b) Can two methods in the same class have the same name? Explain.

Yes, two methods in the same class can have the same name, provided their method signatures are different. This is known as method overloading.

8.

a) What is the purpose of the return statement?

The return statement is used to exit a method and pass a value back to the calling code.

b) How many values can a return statement pass back to the calling code?

A return statement can pass back only one value to the calling code.

c) How does the declaration of a method that returns a value differ from one that does not?

The main difference is in the return type specified in the method declaration. A method that returns a value has a specific return type (like int, String, etc.), while a method that does not return a value uses the void keyword.

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;
    }
}
```

9.

The issue with this code is in the line where doSomething() is called without using its return value. Since doSomething() returns an integer, the return value should either be assigned to a variable or used in some way.