Critical Thinking

1. Explain the difference between method declaration and method body.

- Method body: The statements that implement a method.
- Method declaration: The first line of the method, which contains the method names, access name, return type, and parameter, if any.

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

- The 'public' is an access modifier, meaning that it is used to set the access level for classes, attributes, methods and constructors.

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

- Visibility.

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

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

- The scope of var3 is inside the method1 and can only used within the mothed1
- The scope of var4 is inside the method1 and specifically in the for loop, which is strictly and only used in the for loop.
- The scope of var1 is inside the main method and can only be used within the main method.
- The scope of var2 is inside the main method and specifically in the for loop, which is strictly and only used in the for loop.

5. 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.

```
public static int getVowels(String input) {
    int count = 0;
    input = input.toLowerCase();
    for (int i = 0; i < input.length(); i++) {
        char ch = input.charAt(i);
        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {
            count++;
        }
    }
    return count;
}
```

b) A class method named extractDigit that can be called by any other method, requires an integer parameter, and returns a String parameter.

c) A class method name insertString that can be called by any other method, requires a String parameter and an integer parameter, and returns a String parameter.

```
public class Only_for_testing {
    public static String insertString(int index, String beginString, String newString) {
        if (index<0||index>beginString.length()) {
            System.out.print("Invalid. Run the program again.");
        }
        else {
            newString = beginString.substring(0,index+1)+newString+beginString.substring(index);
        }
        return newString;
    }
    public static void main(String[] arg) {
        Scanner userinput = new Scanner (System.in);

        System.out.print("Enter s string: ");
        String beginString = userinput.nextLine();
        System.out.print("Enter the index you want to insert the next string: ");
        int index = userinput.nextInt();
        userinput.nextLine();
        System.out.print("Enter a new string: ");
        String newString = userinput.nextLine();
        System.out.println(insertString(index, beginString, newString));
    }
    }
```

6.

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

- It distinguishes them by different types parameters, numbers of parameters or order of parameters.

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

- Two methods in the same class can have the same name but need to have different parameter(s) for the compiler to identify.

7.

a) What is the return statement used for?

- Statement used to send a value from a method back to the calling statement.

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

- The return type void indicates that the method will not return a value at all. Return types can be primitive types, such as int and double, or abstract data types, such as String; and only return only value.

c) How is the declaration of a method returning a value different from the declaration of a method that does not return a value?

- Any method declared void doesn't return a value and cannot contain a return statement. Any method that is not declared void must contain a return statement.

8. Find and explain the error in the code below:

```
public class MethodCallExample {
        public static int doSomething() {
```

```
                return(5);
        }
}

  public static void main(String[] arg ) {
        int num;
        doSomething();
        num=doSomething();
  }
```

In order to call the method, the doSomething() and main method have to be in the same class. But in this situation, the doSomething() and main method should be all covered inside the MethodCallExample class in order for the main method to call the doSomething() method.