1. How many methods does the class Person have from Java's point of view?

   ```
   class Person(val name: String, var age: Int) _____
   ```

2. What will be printed? (The second declaration adds `data` modifier)

   ```
   class Evaluation1(val positions: Int, val letters: Int)

   >>> val e1 = Evaluation1(1, 3)
   >>> val e2 = Evaluation1(1, 3)
   >>> println(e1 == e2)        _____
   ```

   ————————————————————————

   ```
   data class Evaluation2(val positions: Int, val letters: Int)

   >>> val e1 = Evaluation2(1, 3)
   >>> val e2 = Evaluation2(1, 3)
   >>> println(e1 == e2)        _____
   ```

3. Find the correspondence between modifiers and their meaning:

Modifier		Meaning	
a	`final` (by default)	1	overrides a member in a superclass or interface
b	`open`	2	must be overridden (can't have an implementation)
c	`abstract`	3	cannot be overridden
d	`override` (mandatory)	4	can be overridden

4. Fill the table with the values:
   everywhere, in a module, in a file, in a class, in subclasses

Modifier	Class member	Top-level declaration
`public` (by default)	visible ?	visible ?
`internal`	visible ?	visible ?
`protected`	visible ?	———
`private`	visible ?	visible ?

5. Which class (nested or inner) stores a reference to an outer class?   _____

In Java	In Kotlin	Class declared within another class
`static class A`	`class A` (by default)	nested class
`class A` (by default)	`inner class A`	inner class

6. Top-level function declared in the file `Util.kt` is compiled to:

   a) a regular non-`final` method of the class UtilKt
   b) a `final` method of the class UtilKt
   c) a `static` method of the class UtilKt

7. How many times the phrase "Calculating the answer..." will be printed?

```kotlin
val foo1 = run {
    println("Calculating the answer...")
    42
}

val foo2: Int
    get() {
        println("Calculating the answer...")
        return 42
    }

fun main(args: Array<String>) {
    println("$foo1 $foo1 $foo2 $foo2")
}
```