ASE CHEAT SHEET

1/ Seven tips for clean code:

1. Use proper naming convention.
2. Write short functions that only do one thing.
3. Write good documentation.
4. Be Consistent.
5. Encapsulation + Modularization.
6. Follow Sandi Metz's Rules.
7. Follow DRY Principle.

2/ Sandi Metz's Rules:

1. Classes can be no longer than 100 lines of code.
2. Methods can be no longer than 5 lines of code.
3. Pass no more than 4 parameters into a method.
4. Controllers can instantiate only one object.

3/ Code smells:

1. **Rigidity:** The software is difficult to change
2. **Fragility:** The software breaks in many places due to a single change.
3. **Immobility:** You cannot reuse parts of the code in other projects because of involved risks and high effort.
4. **Needless Complexity**.
5. **Needless Repetition.**
6. **Opacity:** The code is hard to understand.

4/ DRY: Don't Repeat Yourself:

It states that every piece of knowledge must have a single, unambiguous, authori-tative representation within a system (codebase).

5/ Pass value through activity:

Send:

```
intent.putExtra("key", "value")
```

Receive:

```
var data =
intent.getStringExtra("key")
```

6/ Four phases of scrum:

```
1. Backlog
2. Sprint
3. Scrum meeting
4. Demos
```

7/ Data class:

```
// A data class is a structured data container
data class Student(val name: String, var year: Int)
// name is a read-only property, year is mutable
val newStudent = Student( name: "Rahat", year: 1)
// Data class with properties outside the constructor
data class Professor(val name: String) {
    var isTenured: Boolean = false
}
```

8/ Encryption-decryption library:

```
com.scottyab.aescrypt.AESCrypt
```

9/ Generate random string:

```
private fun getRandomString(length: Int) : String {
    val charset =
"ABCDEFGHIJKLMNOPQRSTUVWXTZabcdefghiklmnopqrstuvwxyz0123456789"
    return (1..length)
        .map { charset.random() }
        .joinToString("")
}
```

10/ Sealed Classes:

```
// can make multiple instances
sealed class Shape {
    class Circle(val radius: Int): Shape()
    class Square(val sideLength: Int): Shape()
}
val circle1 = Shape.Circle( radius: 3)
val circle2 = Shape.Circle( radius: 42)
val square = Shape.Square( sideLength: 5)
```