

## Lab 2 การเขียนโปรแกรมสร้าง Class และ Object ด้วยภาษา Kotlin

### เอกสารประกอบการปฏิบัติการรายวิชา SC362007 MOBILE DEVICE PROGRAMMING

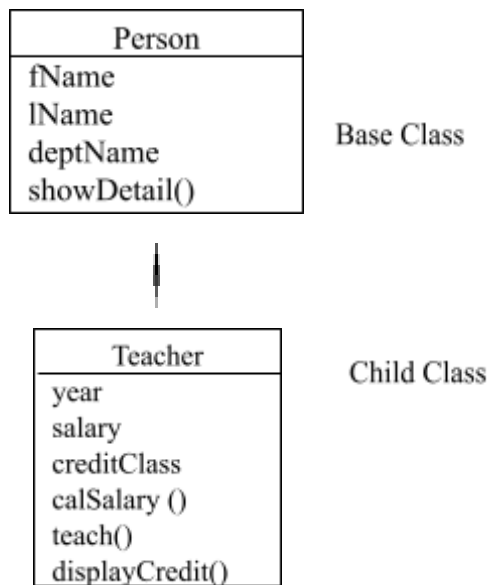
#### วัตถุประสงค์

1. เพื่อให้นักศึกษาสามารถใช้เขียนโปรแกรมด้วยภาษา Kotlin เบื้องต้นได้
2. เพื่อให้นักศึกษาสามารถใช้คำสั่งสร้าง Class และ Object ได้
3. เพื่อให้นักศึกษาสามารถใช้คำสั่งสร้างและเรียกใช้ Companion object และ Singleton object

**คำอธิบาย** ในใบปฏิบัติการนี้ให้นักศึกษาสร้าง Class และ Object โดยมีการสร้าง Class 3 คลาส ดังนี้

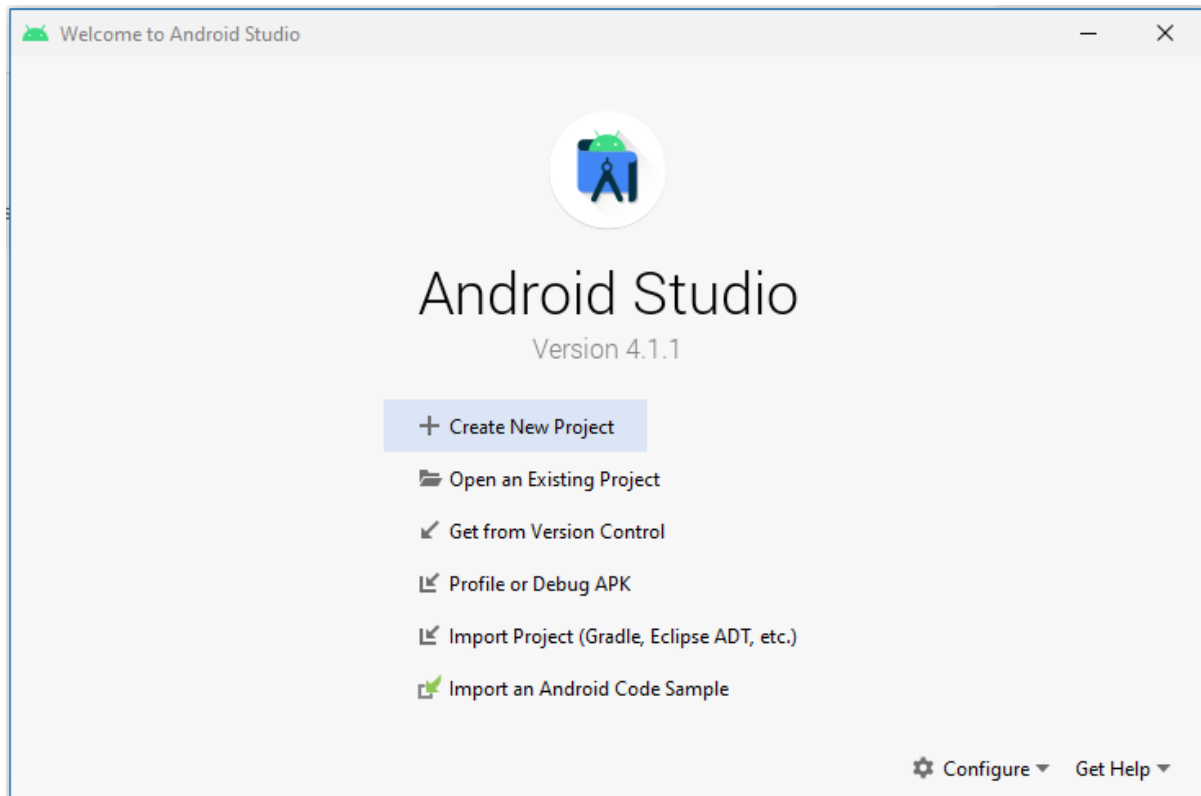
1. Class ชื่อ Subject เป็น Data Class ที่มี Constructor ที่รับค่า รหัสวิชา (id), ชื่อวิชา (name) และหน่วยกิต ของวิชา (credit)
2. Class ชื่อ Person มี Constructor ที่รับค่า ชื่อ(fName), นามสกุล(lName) และภาควิชา(deptName) โดยมีการ init ค่าของตัวแปร นอกจากนี้ยังมี ฟังก์ชัน showDetail ซึ่งสามารถสืบทอดไปยังคลาสลูกและสามารถแก้ไขฟังก์ชันนี้ได้ และสร้าง Companion object เพื่อให้สามารถใช้งานโดยไม่ต้องทำการประกาศ Object ได้
3. Class ชื่อ Teacher สืบทอดมาจาก Class ของ Person ที่มี Constructor ที่รับค่าเพิ่มมาจาก Class ของ Person คือ ปีที่ทำงาน(year) และมีตัวแปรใน Class เพิ่ม คือ เงินเดือน(salary) และ หน่วยกิตที่สอน(creditClass) นอกจากนี้ภายในคลาสมีฟังก์ชัน 4 ฟังก์ชัน คือ
  - ฟังก์ชัน showDetail ที่สืบทอดจากคลาส Person ซึ่งต้องแก้ไขข้อความ โดยเพิ่มค่าของ year แล้วปริ้นต์แสดงข้อมูล
  - ฟังก์ชัน calSalary เป็นฟังก์ชันคำนวณเงินเดือนจากอายุการทำงาน โดยกำหนดเงินเดือนดังนี้
    - ถ้าจำนวนปีที่ทำงาน < 5 ได้เงินเดือน = 25,000 + (2,000 \* จำนวนปีที่ทำงาน)
    - ถ้าจำนวนปีที่ทำงานน้อยกว่า 10 ได้เงินเดือน = 36,000 + (2,000 \* (จำนวนปีที่ทำงาน - 5))
    - ถ้าจำนวนปีที่ทำงานน้อยกว่า 15 ได้เงินเดือน = 47,000 + (2,000 \* (จำนวนปีที่ทำงาน - 10))
    - ถ้าจำนวนปีที่ทำงานน้อยกว่า 20 ได้เงินเดือน = 58,000 + (2,000 \* (จำนวนปีที่ทำงาน - 15))
    - ถ้าจำนวนปีที่ทำงานมากกว่าหรือเท่ากับ 20 ได้เงินเดือน = 60,000 + (2,000 \* (จำนวนปีที่ทำงาน - 20))
  - ฟังก์ชัน teach เป็นฟังก์ชันที่แสดงวิชาที่อาจารย์สอนโดยรับค่าจาก object ของ Class Subject ซึ่งเป็น data class แล้วแสดงรายละเอียดของวิชา
  - ฟังก์ชัน displayCredit เป็นฟังก์ชันที่แสดงจำนวนหน่วยกิตทั้งหมดที่อาจารย์สอน

นอกจากการสร้างคลาสด้านบนแล้ว ให้สร้าง Singleton object เพื่อทดสอบด้วย

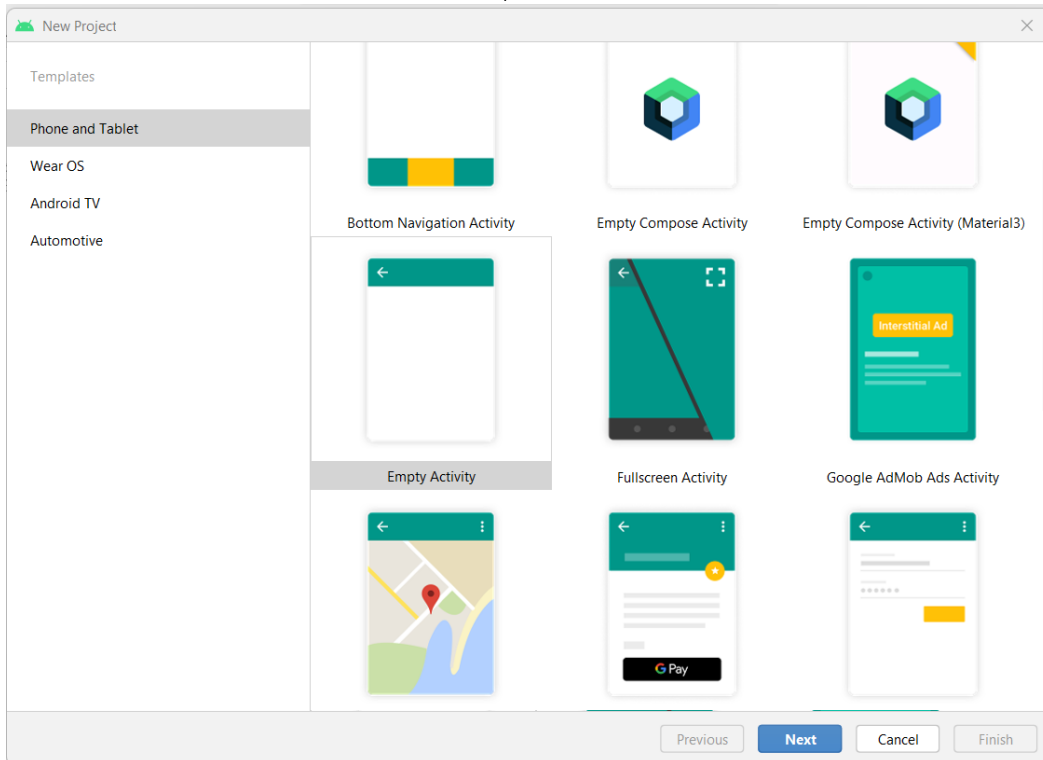


### ขั้นตอนการเขียนโปรแกรมสร้าง Class และ Object

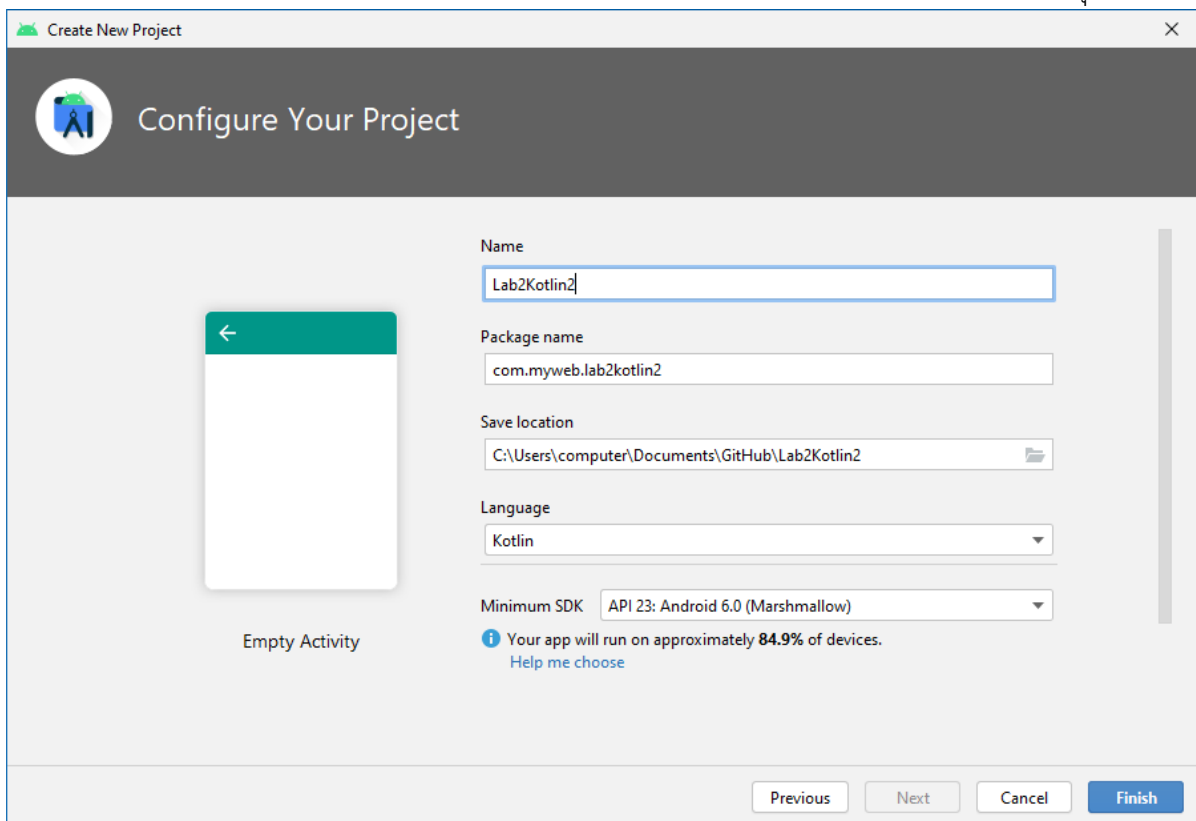
1. ให้เปิดโปรแกรม Android Studio จะแสดงหน้าจอดังนี้ ให้เลือก Create New Project



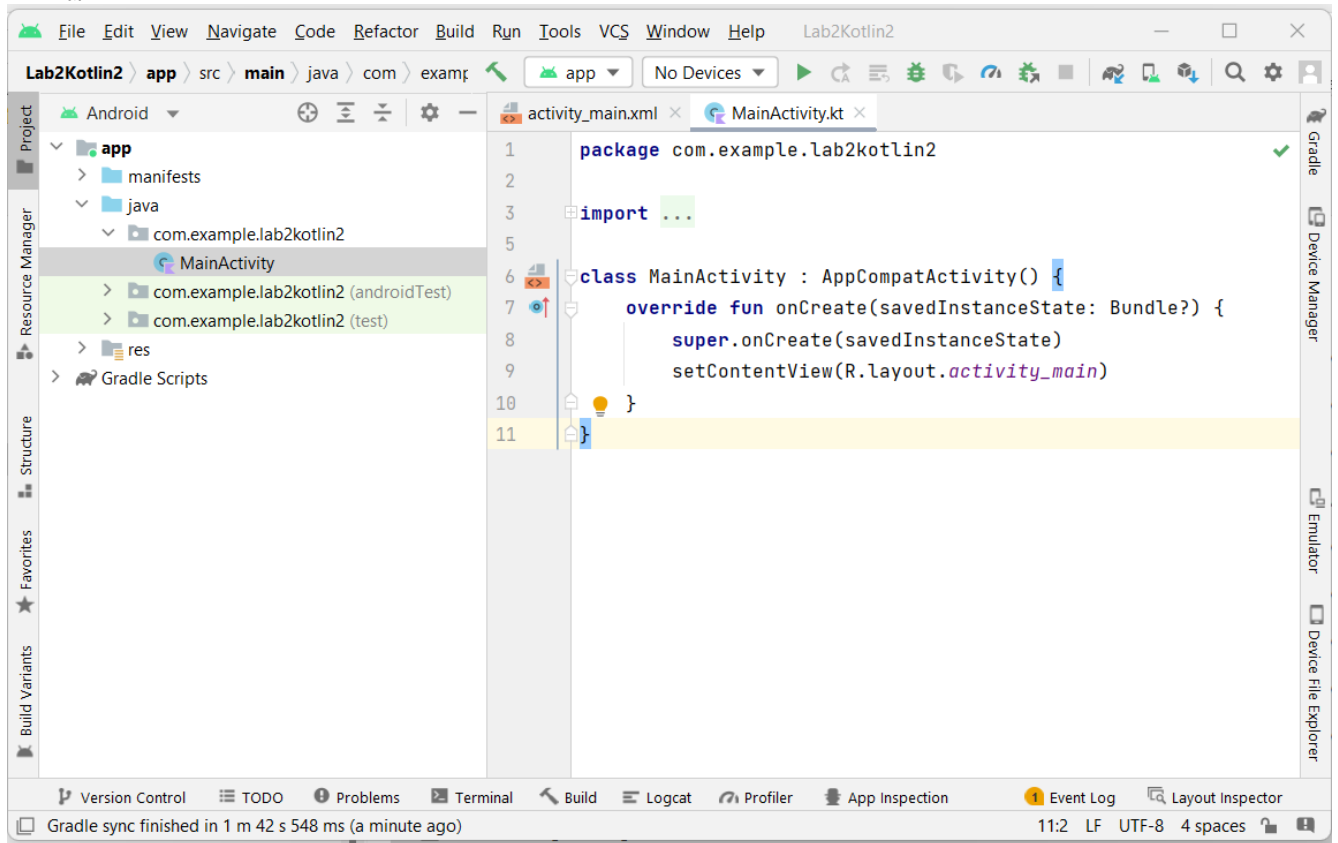
2. จากนั้นเลือก Empty Activity แล้วคลิกที่ปุ่ม Next



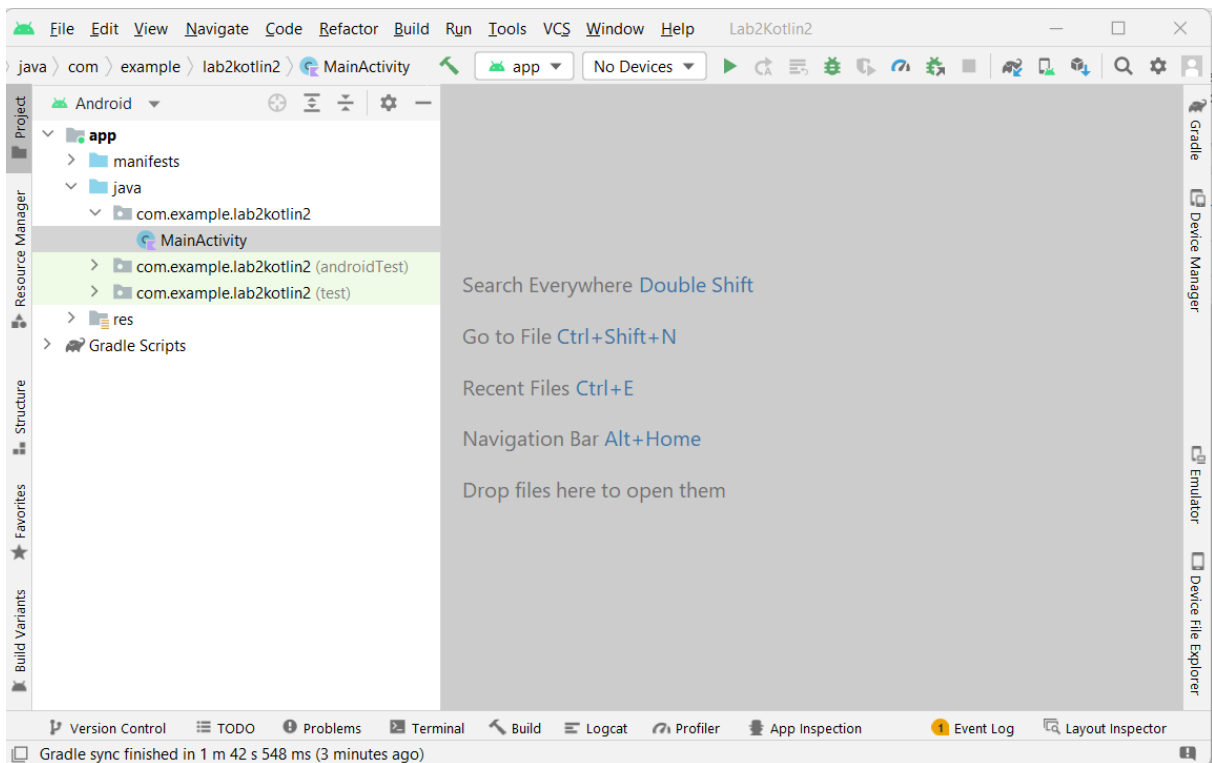
3. ต่อมาให้กำหนดชื่อโปรเจกต์ ให้กำหนดดังนี้ โดยเก็บไฟล์ไว้ที่ โฟลเดอร์ Lab2Kotlin2 แล้วคลิกที่ปุ่ม Finish



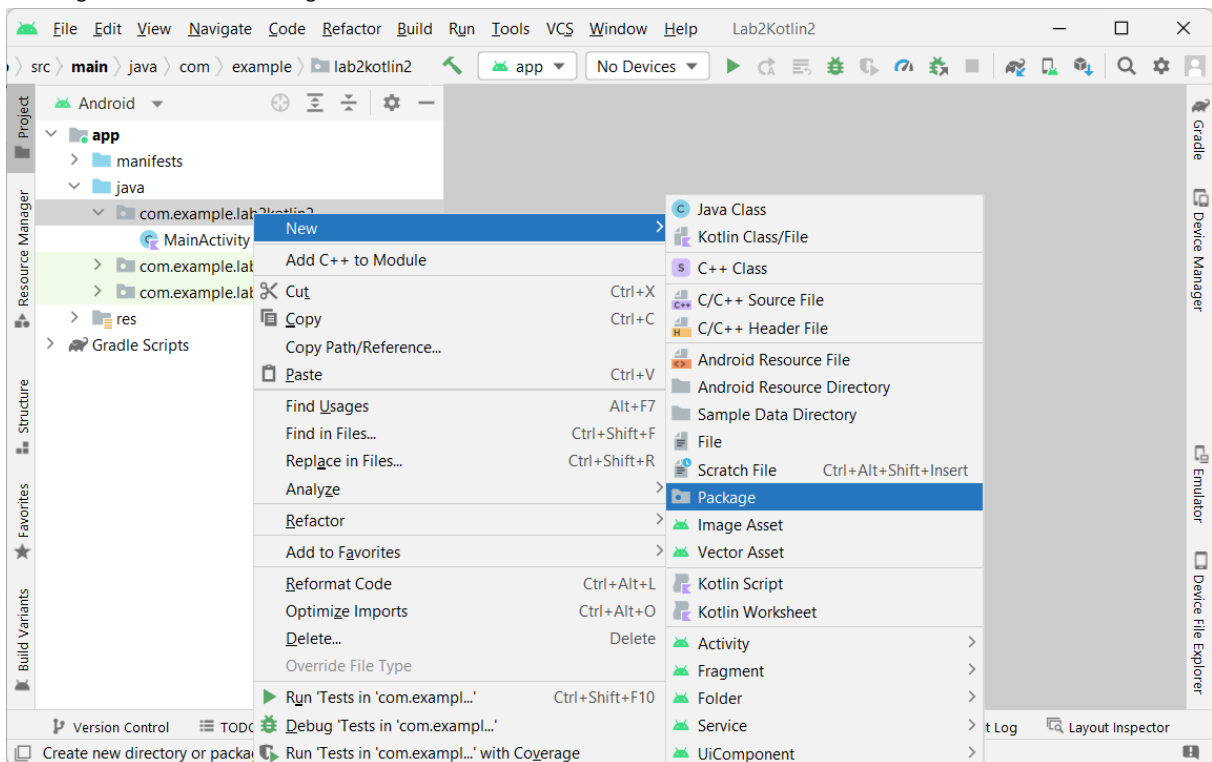
ปรากฏหน้าจอ ดังนี้

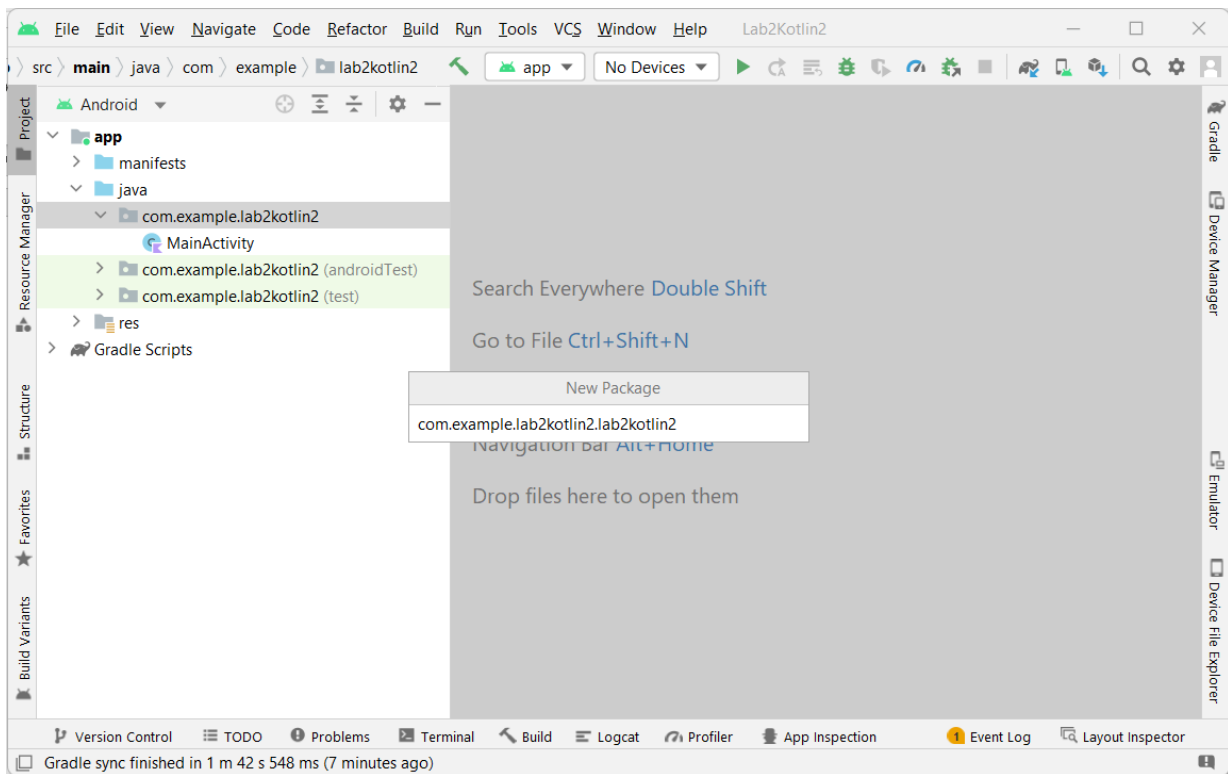


เนื่องจากในปฏิบัติการนี้จะเรียนเรื่องของการเขียนภาษา Kotlin ดังนั้นให้ปิดแท็บของ activity\_main.xml และ MainActivity.kt ไปก่อน เพื่อไม่ให้สับสน

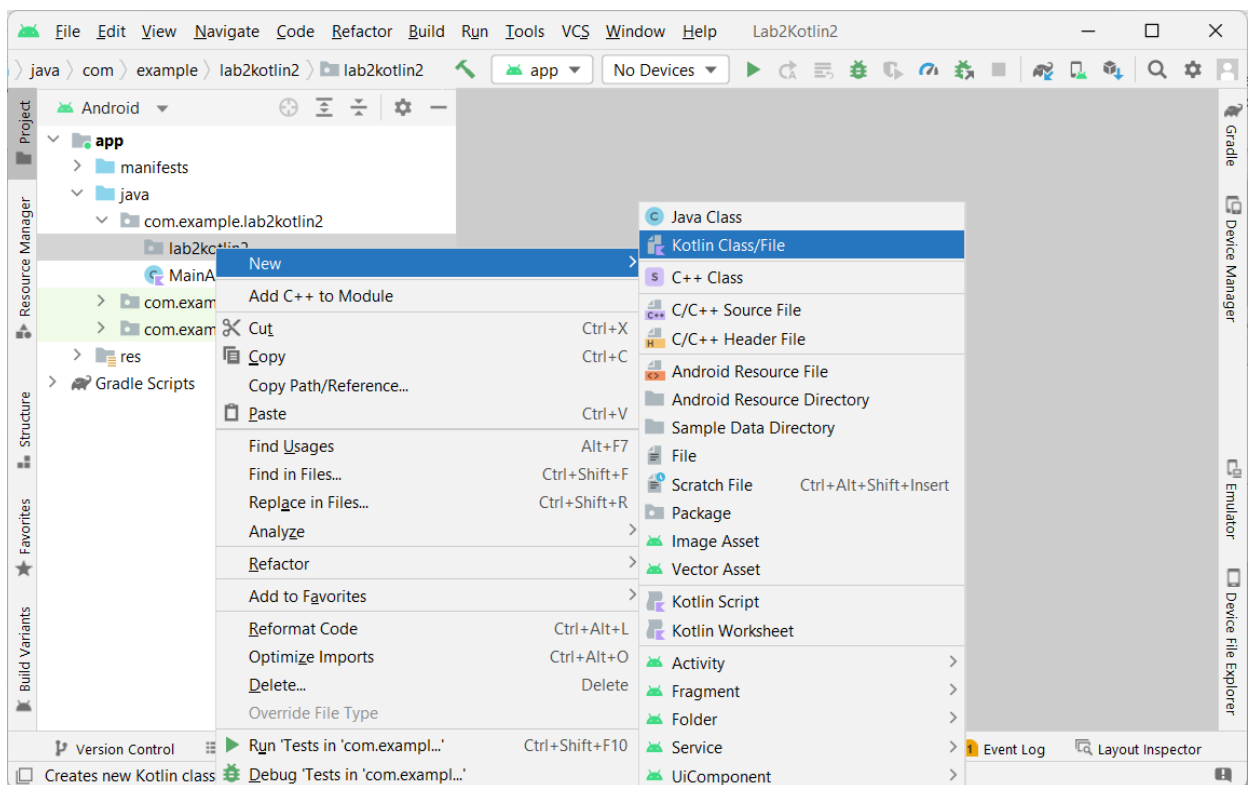


4. จากนั้นไปที่เมนูด้านซ้าย คลิกที่ app >> java >> com.example.lab2kotlin2 แล้วคลิกขวาแล้วเลือก New >> Package แล้วตั้งชื่อ Package ว่า lab2kotlin2

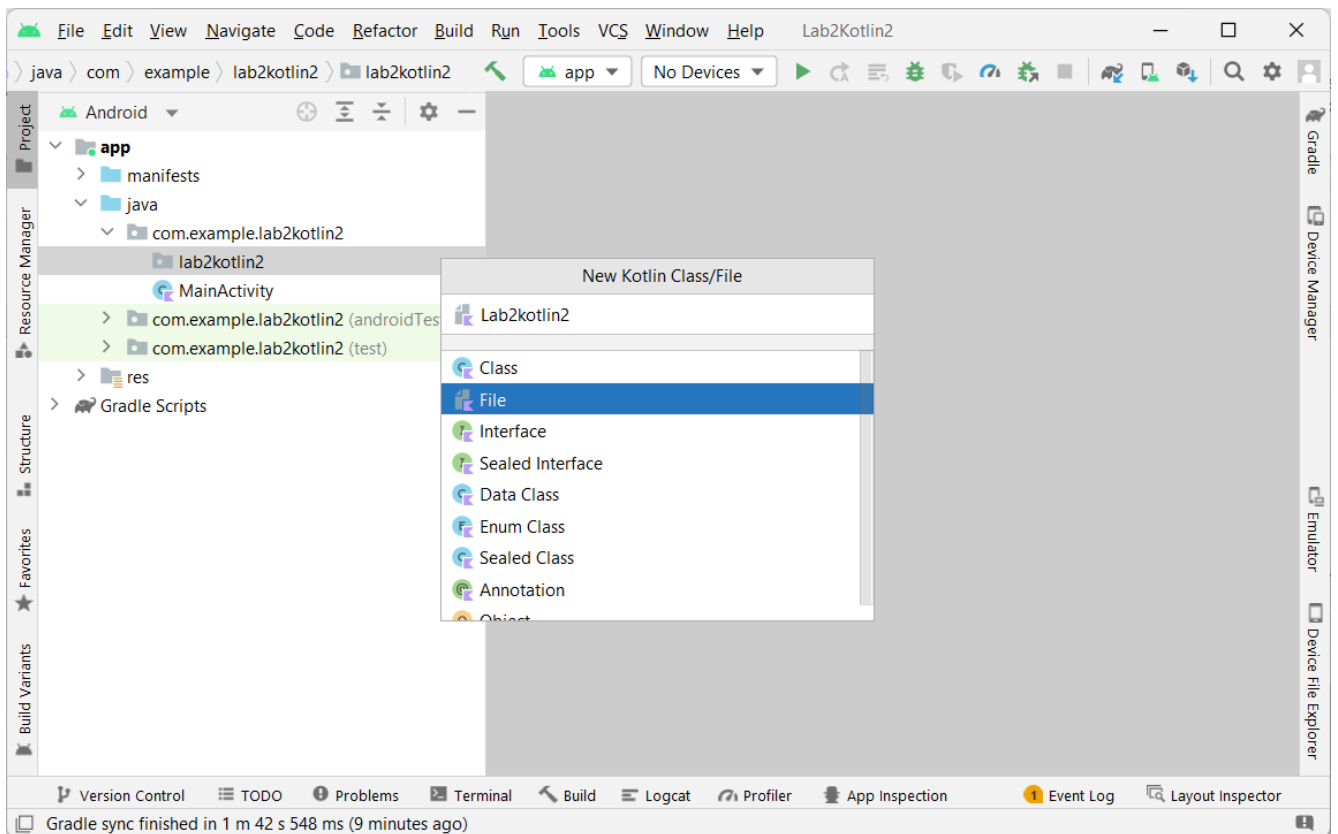




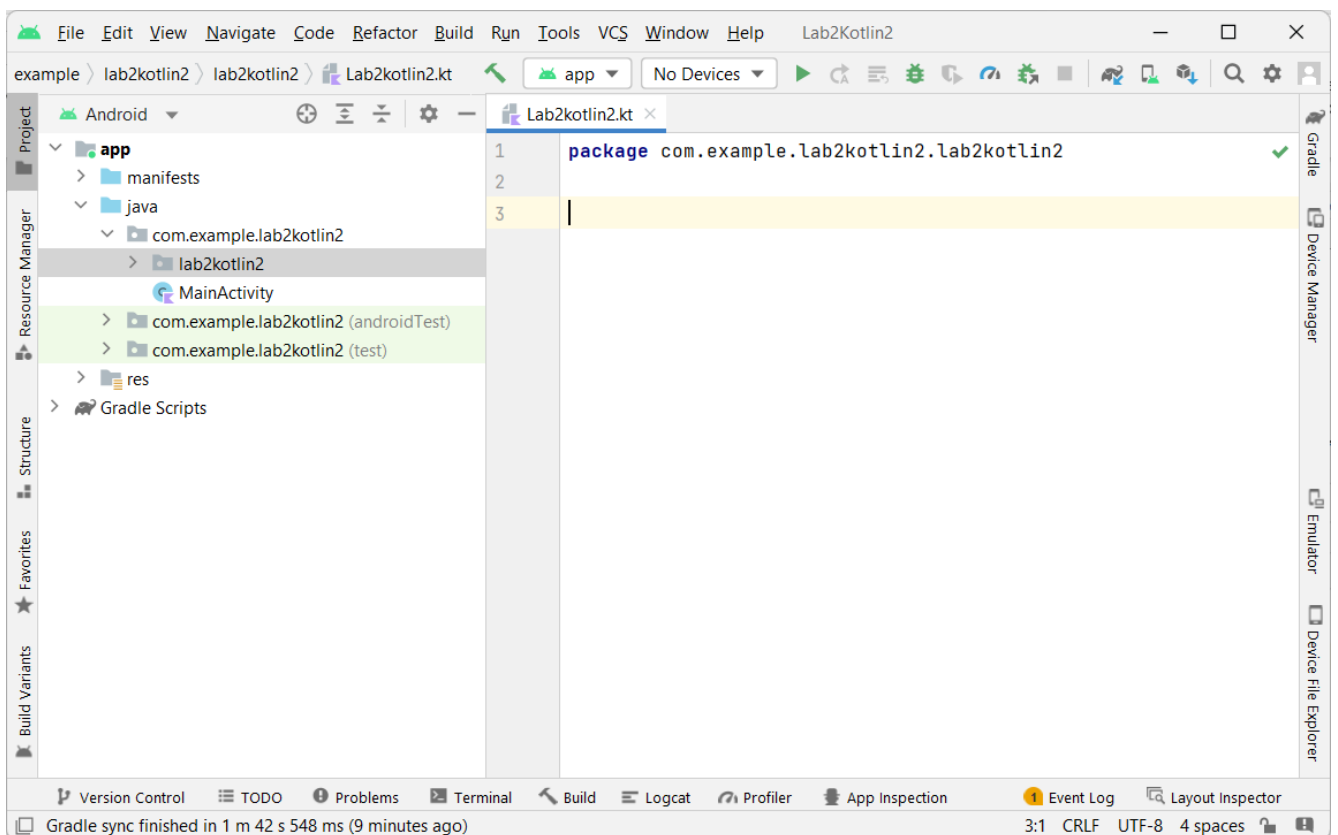
5. จากนั้นให้สร้างไฟล์ Kotlin ไปที่ lab2kotlin2 แล้วคลิกขวา New >> Kotlin File/Class



จากนั้นพิมพ์ชื่อ Lab2kotlin2 แล้วเลือกประเภทเป็น File



โปรแกรม Android Studio จะสร้างไฟล์ให้



6. จากนั้นให้พิมพ์คำสั่งเพื่อสร้าง Class โดยเริ่มสร้าง Subject ซึ่งเป็น Data Class ดังนี้

```
1 package com.example.lab2kotlin2.lab2kotlin2
2
3 //🔥 Data Class : Subject Class
4 data class Subject(val id: String, val name: String, val credit: Int)
```

7. จากนั้นให้พิมพ์คำสั่งต่อเพื่อสร้าง Class ชื่อ Person ซึ่งเป็น Parent Class (หนึ่งไฟล์สามารถสร้างคลาสได้หลายคลาส) ดังนี้

```
3 /// Data Class : Subject Class
4 data class Subject(val id: String, val name: String, val credit: Int)
5
6 //Parent Class : Person Class
7 open class Person(fName: String, lName: String, deptName: String ) {
8     val firstName: String
9     val lastName: String
10    protected val department: String
11
12    // initializer block
13    init {
14        firstName = fName.replaceFirstChar { it.uppercase() } // Uppercase on first Character
15        lastName = lName.replaceFirstChar { it.uppercase() } // Uppercase on first Character
16        department = "$deptName, College of Computing"
17    }
18
19    /// Member Function : showDetail
20    open fun showDetail() {
21        println("$firstName is at $department.")
22    }
23
24    /// Companion Object
25    companion object {
26        fun showCompanion(first_Name:String, last_Name:String ,age:Int){
27            println("Person is called from companion object : $first_Name $last_Name is $age years old.")
28        }
29    }
30 }
```



8. จากนั้นให้พิมพ์คำสั่งเพื่อสร้าง Class ชื่อ Teacher ที่สืบทอดมาจาก Class Person ดังนี้

```
32  /// Inheritance from Person Class
33  class Teacher(fName: String, lName: String, deptName: String, year: Int): Person(fName, lName, deptName)
34      private var salary : Int = 0
35      private val yearClass: Int = year
36      private var creditClass :Int = 0
37
38      /// Override showDetail() Method in Person Class
39      override fun showDetail() {
40          println("$firstName is a teacher for $yearClass years at $department.")
41      }
42      /// Calculate teacher's salary
43      fun calSalary () {
44          when {
45              yearClass< 5 -> salary = 25000 + (2000 * yearClass)
46              yearClass< 10 -> salary = 36000 + (2000 * (yearClass - 5 ))
47              yearClass< 15 -> salary = 47000 + (2000 * (yearClass - 10))
48              yearClass< 20 -> salary = 58000 + (2000 * (yearClass - 15))
49              else -> salary = 60000 + (2000 * (yearClass - 20))
50          }
51          println("$firstName's salary is $salary baht")
52      }
53
54      fun teach(subj : Subject){
55          println(subj.toString())
56
57          /// Summation all credits
58          creditClass += subj.credit
59      }
60
61      ///Show Summation of credits
62      fun displayCredit(){
63          println("$firstName teaches $creditClass credits.")
64      }
65  }
```

9. ต่อมาให้พิมพ์คำสั่งสร้าง Singleton object ชื่อ Singleton\_Person เพื่อให้สามารถเรียกใช้งานแบบไม่ต้องประกาศ Object

```
66  object Singleton_Person{
67      val first_Name = "David"
68      val last_Name = "Bowie"
69      var age = 23
70      fun showCompanion(){
71          println("Person is called from singleton object : $first_Name $last_Name is $age years old.")
72      }
73  }
```

10. ต่อมาให้พิมพ์คำสั่งใน ฟังก์ชัน main ดังนี้

```

75 ▶ fun main() {
76     /// Create object from Person Class
77     var person1 = Person( fName: "Alice", lName: "Wonderland", deptName: "Computer Science")
78     println("Member NO 1 : " + person1.firstName + " " + person1.lastName )
79     person1.showDetail()
80     println()
81
82     ////Call From companion object
83     println("Member NO 2 :")
84     Person.showCompanion( first_Name: "Bobby", last_Name: "Brown", age: 25)
85     println()
86
87     /// Create object from Subject Class
88     var subject1 = Subject( id: "SC362007", name: "Mobile Device Programming", credit: 3)
89     var subject2 = Subject( id: "SC362005", name: "Database Analysis and Design", credit: 3)
90     var subject3 = Subject( id: "SC361003", name: "Object Oriented Concepts and Programming", credit: 1)
91
92     /// Create object from Teacher Class
93     var person2 = Teacher( fName: "Chris", lName: "Evans", deptName: "Information Technology", year: 25)
94     println("Member NO 3 : " + person2.firstName + " " + person2.lastName)
95     person2.showDetail()
96     person2.calSalary()
97     println(person2.firstName + " teaches: ")
98     person2.teach(subject1)
99     person2.teach(subject2)
100    person2.teach(subject3)
101    person2.displayCredit()
102    println()
103
104    ////Call From Singleton Object
105    println("Member NO 4 :")
106    Singleton_Person.showCompanion()
107 }

```

เมื่อ Run โปรแกรม จะแสดงผลดังต่อไปนี้

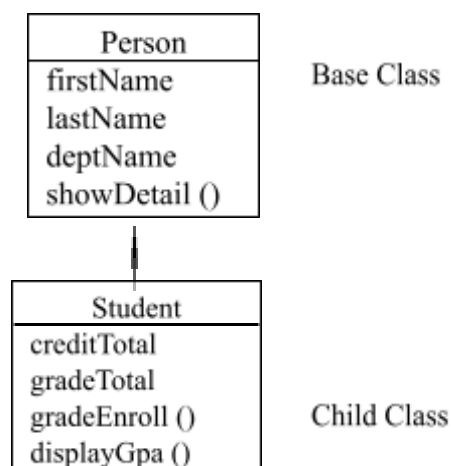
```
Run: Lab2kotlin2Kt x
Person is called from companion object : Bobby Brown is 25 years old.

Member NO 3 : Chris Evans
Chris is a teacher for 25 years at Information Technology, College of Computing.
Chris's salary is 70000 baht
Chris teaches:
Subject(id=SC362007, name=Mobile Device Programming, credit=3)
Subject(id=SC362005, name=Database Analysis and Design, credit=3)
Subject(id=SC361003, name=Object Oriented Concepts and Programming, credit=1)
Chris teaches 7 credits.

Member NO 4 :
Person is called from singleton object : David Bowie is 23 years old.

Process finished with exit code 0
```

## Assignment 2



กำหนดให้สร้าง Class ชื่อ **Student** ที่สืบทอดมาจาก Class ของ Person โดยให้สร้างตัวแปรเพิ่ม 2 ตัวแปร คือ creditTotal ใช้ในการเก็บผลรวมของหน่วยกิตที่นักศึกษาลงทะเบียน และ gradeTotal ใช้ในการเก็บผลรวมคะแนนที่ได้จากการคำนวณของเกรด และกำหนดให้สร้าง Function 3 อัน ดังนี้

1. ฟังก์ชัน showDetail ที่สืบทอดมาจาก Class ของ Person เพื่อแสดงข้อความ ดังตัวอย่าง

**Grace is a student at Information Technology, College of Computing.**

2. ฟังก์ชัน gradeEnroll() รับค่าจาก object ของ Class Subject และคะแนนที่ได้ โดยกำหนดค่าคะแนน เพื่อไปคำนวณเกรดของนักศึกษา ดังนี้

ถ้าได้คะแนน น้อยกว่า 50 ได้เกรด F

ถ้าได้คะแนน น้อยกว่า 55 ได้เกรด D

ถ้าได้คะแนน น้อยกว่า 60 ได้เกรด D+

ถ้าได้คะแนน น้อยกว่า 65 ได้เกรด C

ถ้าได้คะแนน น้อยกว่า 70 ได้เกรด C+

ถ้าได้คะแนน น้อยกว่า 75 ได้เกรด B

ถ้าได้คะแนน น้อยกว่า 80 ได้เกรด B+

ถ้าได้คะแนน มากกว่า 80 ได้เกรด A

ให้นำเกรดที่ได้ไปคำนวณเกรดเฉลี่ย(GPAX) โดยกำหนดให้เกรด A = 4, B+ = 3.5, B = 3, C+ = 2.5, C = 2, D+ = 1.5, D = 1, F = 0 เพื่อนำมาคำนวณคะแนนของเกรดทั้งหมด และต้องหาผลรวมของหน่วยกิตที่ลงทะเบียนทั้งหมด เพื่อนำค่าทั้ง 2 มาคำนวณหาเกรดเฉลี่ย โดยเกรดเฉลี่ยคำนวณได้จาก คะแนนของเกรดทั้งหมด / ผลรวมของหน่วยกิตที่ลงทะเบียนทั้งหมด

และเขียนคำสั่งใน **function main** ใช้คำสั่ง ดังนี้

\*\*\*โดยกำหนดให้ person3 คือ object จาก Class ของ Student

```
var subject = Subject("SC362007","Mobile Programming", 3)
```

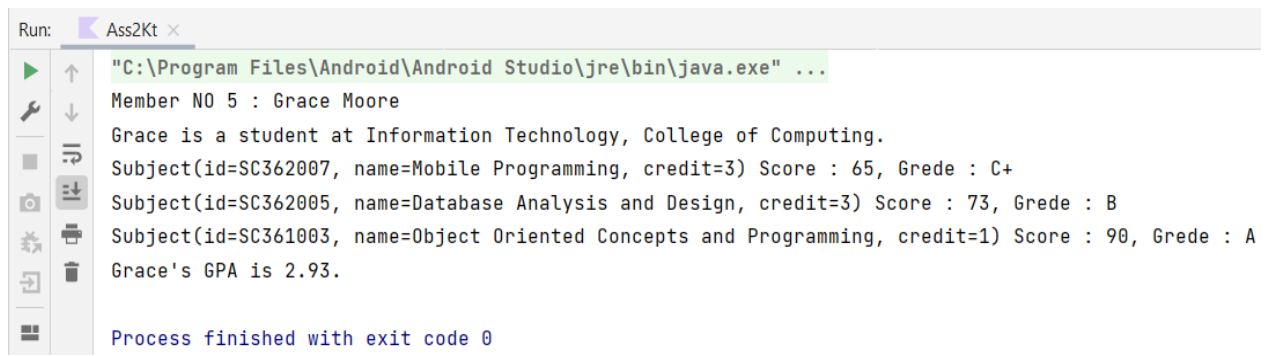
```
person3.gradeEnroll(subject, 65)
```

**Subject(id=SC362007, name=Mobile Programming, credit=3) Score : 65, Grede : C+**

3. ฟังก์ชัน displayGpa () เพื่อแสดงค่าของ GPA

**Grace's GPA is 2.93.**

โดยให้แสดงผลลัพธ์ทั้งหมด ดังนี้



```
Run: Ass2Kt x
"C:\Program Files\Android\Android Studio\jre\bin\java.exe" ...
Member NO 5 : Grace Moore
Grace is a student at Information Technology, College of Computing.
Subject(id=SC362007, name=Mobile Programming, credit=3) Score : 65, Grede : C+
Subject(id=SC362005, name=Database Analysis and Design, credit=3) Score : 73, Grede : B
Subject(id=SC361003, name=Object Oriented Concepts and Programming, credit=1) Score : 90, Grede : A
Grace's GPA is 2.93.
Process finished with exit code 0
```

**\*\*หมายเหตุ:** การแสดงรูปแบบของตัวเลขของ GPA ให้มีทศนิยม 2 ตำแหน่ง สามารถใช้คำสั่ง

`String.format("%.2f", ค่าของ GPA)`