

## COMP2232 - Lab Practical #3

Concepts: *Inheritance, Overriding, keyword final; Introduction to GUI using Swing Components.*

### A – Inheriting from a Class

---

Inheritance allows a class to gain all of the properties (attributes and behaviours) of another class.

1. You will need to use the **Vehicle** class you created in a previous lab.
2. Create a class called **FamilyCar** which inherits from the **Vehicle** class. Your **FamilyCar** class should have an additional method, void carPool(). This method simply outputs a message “Mom’s Taxi ready to roll!”
3. Create a class called **SportCar** which inherits from the **Vehicle** class. Your **SportsCar** class should have an additional method, void race(). This method simply outputs a message “Let’s race! VROOM-VROOM!”
4. Create a class called **MyDriving** which contains a **main** method. This method will create an instance (object) of type **FamilyCar** and on one of type **SportsCar**.
  - a. Ask the user if they want to drive a family car or sportscar and at what speed;
  - b. Use the correct object to accelerate and then brake;
  - c. Try to call the **race** method on the **FamilyCar** object. What happens?
  - d. You would receive an error – FamilyCar does not contain a race method.



### B – Overriding Inherited Methods

---

Overriding allows a subclass to provide a new definition for an inherited method.

1. Your **SportsCar** class has inherited the *accelerate* method from Vehicle.
2. Using your **SportsCar** class, override the accelerate method so that the car will accelerate at a rate of 10km.
3. Use your **MyDriving** class to test this new method by calling the accelerate method on the SportCar object. Does it behave differently to the original version in Vehicle?



### C – The Keyword final

---

The keyword final can be used to achieve any one of the following depending on how it is used: prevent inheritance, prevent overriding, create a constant.

1. Experiment - Inheritance:
  - a. Add final to the Vehicle class as follows: `final public class Vehicle`
  - b. Try to compile your files. What happens?

- c. You should receive an error (FamilyCar & SportsCar). This usage of the keyword final prevents inheritance of the Vehicle class.
2. Experiment - Overriding:
  - a. Add final to the Vehicle class as follows: `final public void accelerate()`
  - b. Try to compile your files. What happens? Did you get an error?
  - c. You should receive an error (SportsCar). This usage of the keyword final prevents overriding (redefinition) of the accelerate method by the subclass.
3. Experiment - Constants:
  - a. Add a data member to your FamilyCar class as follows:  
`final private MAX_SEATS = 5;`
  - b. Add the method below to the class:

```
public void changeSeats(int newMax)
{
    MAX_SEATS = newMax;
}
```
  - c. In your MyDriving class, add a line to call the method above on your FamilyCar object. (Make sure to pass a value to the function call!)
  - d. Try to compile & run your files. What happens? Did you get an error?
  - e. You should receive an error (FamilyCar). This usage of the keyword final prevents the value of the data member MAX\_SEATS from changing.



**<End of Lab>**