Lab Exercise-2

Mobile and Application Development Lab

Note: From this lab onwards, all the programs will be made and kept in a different working folder by each student and identified uniquely by the enrolment number suffixed with MADL2020. For example, 1102_MADL2020.

1. Test for Inheritance
   Make a class Mother with int x and void show ( ) as its members. Now create another class Child that is derived from the Mother class. Create another class Application to test the inheritance.

```
class Mother {.....}
class Child extends Mother {// Empty Body }  Note: "//" is used for single line comments
                                                    in java.
class Application {
 public static void main (String args[]){
    Mother m= new Mother ( );
    m.show( ); // show of Mother is called
    Child ch=new Child ( );
    ch. show ( ); // show ( ) inherited in Child from Mother is called
 }
}
```

2. Test for Overriding
   Make a function with the same return type , name of the function, number and type of arguments in the Child class as they are in the Mother class. Change the string to be displayed on the screen. For example, if Mother class version of show ( ) was displaying "Hello World" then the child class version of show ( ) will display "Hello JUET". Run the application class and discuss the results obtained with your instructor.

3. Test for Polymorphism
   Write another statement in Application class: Mother m1=new Child ( ); Now, call show () with reference variable m1 and '.' (dot) operator. Discuss the result with your instructor. Now, make the method show ( ) static in Mother only and check the results after executing Application. Repeat this by making show () static in Child only. Observe the errors. Test whether the static methods are inheritable or not. Now make show () static in both Mother as well as in Child and discuss the results with your instructor. Make a table to summarize your results.

4. Parameterized Constructors
   Create a class One and make a parameterized constructor One(int x){ } inside this class. Create a class Two derived from One: class Two extends One { }. Now, compile One and then Two. Discuss the results with your instructor. Hint: Class Two should not compile and must throw an error. Analyze the error and rectify it.