ASSIGNMENT 5
2021KUCP1104

1. Write a Java code snippet that demonstrates method overloading with three methods that have the same name but different number of parameters.

```
public class MethodOverloadingExample {

    public static void main(String[] args) {
    MethodOverloadingExample example = new MethodOverloadingExample();
    System.out.println(example.add(1, 2));
    System.out.println(example.add(1, 2, 3));
    System.out.println(example.add(1, 2, 3, 4));
    }

    public int add(int a, int b) {
    return a + b;
    }

    public int add(int a, int b, int c) {
    return a + b + c;
    }

    public int add(int a, int b, int c, int d) {
    return a + b + c + d;
    }

}
```

2 Write a Java code snippet that demonstrates method overriding with a superclass and a subclass. The subclass should call the overridden method of the superclass before printing its own message.

```
    public class Animal {

        public void speak() {

        System.out.println("The animal makes a sound.");

        }

    }
```

```
public class Cat extends Animal {


    public void speak() {
    super.speak();
    System.out.println("The cat meows.");
    }
}


public class Main {
    public static void main(String[] args) {
    Animal animal = new Animal();
    Cat cat = new Cat();


    animal.speak();
    cat.speak();
    }
}
```

3 Write a Java code snippet that demonstrates overloading constructors in a class with default values for some parameters.

```
public class Car {
    private String make;
    private String model;
    private int year;
    private int mileage;
    public Car() {
    this("Unknown", "Unknown", 0, 0);
    }

    public Car(String make, String model) {
    this(make, model, 0, 0);
    }

    public Car(String make, String model, int year) {
    this(make, model, year, 0);
    }

    public Car(String make, String model, int year, int mileage) {
    this.make = make;
    this.model = model;
    this.year = year;
    this.mileage = mileage;
    }

    public String getMake() {
    return make;
    }

    public String getModel() {
    return model;
    }

    public int getYear() {
    return year;
    }

    public int getMileage() {
    return mileage;
    }

    public static void main(String[] args) {
    Car car1 = new Car();
    Car car2 = new Car("Toyota", "Camry");
    Car car3 = new Car("Honda", "Accord", 2018);
    Car car4 = new Car("BMW", "X3", 2021, 5000);

    System.out.println(car1.getMake() + " " + car1.getModel() + " " + car1.getYear() + " " +
car1.getMileage());
```

```
        System.out.println(car2.getMake() + " " + car2.getModel() + " " + car2.getYear() + " " +
car2.getMileage());
        System.out.println(car3.getMake() + " " + car3.getModel() + " " + car3.getYear() + " " +
car3.getMileage());
        System.out.println(car4.getMake() + " " + car4.getModel() + " " + car4.getYear() + " " +
car4.getMileage());
        }
}
```

4. Write a Java code snippet that demonstrates how to prevent a class from being subclassed using the final keyword.

```
final class Bike{}

class Honda1 extends Bike{
 void run(){System.out.println("running safely with 100kmph");}

 public static void main(String args[]){
 Honda1 honda= new Honda1();
 honda.run();
 }
}
```

5 how to overload a static method in a class with multiple methods having the same name but different parameter types

```
public class MyClass {
        public static void myMethod(int x) {
        // implementation for int parameter
        }

        public static void myMethod(double y) {
        // implementation for double parameter
        }

        public static void myMethod(String s) {
        // implementation for string parameter
        }
}
```