example of method changing:

public class Method\_chaining {

public Method\_chaining method (){

System.out.println("method #1");

return this;

}

public Method\_chaining method\_2 (){

System.out.println("method #1");

return this;

}

public Method\_chaining method\_3 (){

System.out.println("method #1");

return this;

}

public static void main(String [] args){

Method\_chaining methodchaining = new Method\_chaining();

methodchaining.method().method\_2().method\_3();

//If the return type of the method is object and we want to call different method of the same object we call the method directly, without mentioning the object.

}

}

We may want to get rid of the object of the class that we create and use in a main method, for that we need to change

signature of the first method that we call to be a static, because if it will be static we can call this method without creating an object, and the method should return ab object of the class. So since it returns the object of the class we may use method changing.

public class Method\_chaining {

public static Method\_chaining method (){ <- - we change it to be a static in order to call it without create object

System.out.println("method #1");

return new Method\_chaining(); <- - return object so we can call directly the next method without need to create object.

}

public Method\_chaining method\_2 (){

System.out.println("method #2");

return this;

}

public Method\_chaining method\_3 (){

System.out.println("method #3");

return this;

}

public static void main(String [] args){

method().method\_2().method\_3();

//If the return type of the method is object and we want to call different method of the same object we can use

//method chaining, line #21.

}

}