

FormFactory.java

```
1 package project;
2
3 import java.awt.Color;
21
22 /* This will create FormConfiguration and GenericForm objects.
23 * First it creates FormConfiguration objects for different forms by reading the
   form data
24 * from the config.txt file and then creates the respective GenericForm object */
25 public class FormFactory {
26     public static int unID=0;
27     private String fileName;
28
29     public FormFactory(String fileName){
32
33     /* Basically it takes a line of String and then parse into ConfigAttribute
   Object
34     * MapToConfig method should create different formIDs for different action.
35     * For an example, if the action is mapping then it should create formID=100
   or 200 or 300
36     * depend on how many different mappings blocks the text file have. in our case
   , there are
37     * 2 mapping blocks, so it should create attributes with formID=100,200
38     * if The action is form Parsing , it will create with formID=201,202,203
   .....
39     */
40     public static ConfigAttribute mapToConfig(String line) {
58
59     /* This will create Color object from Integer array with 3 values for RGB
   */
60     private Color mapToColor(String[] colorArray){
64
65     /*This will map the String into class */
66     private static Class<?> mapToClass(String str){
83
84     /* This will create List of String from ConfigAttribute object by converting
   Array of
85     * String into List of String */
86     private static List<String> mapToList (ConfigAttribute configAttribute){
91
92     /* This will convert Array of String into Array of class */
93     private Class<?>[] convertToClassArray(String[] array){
105
106     /* This method creates collection of ConfigAttributes Objects by scanning an
   external file line by line
107     * and then parse into ConfigAttribute Objects using "mapToConfig" method. The
   external file is the storage
108     * where the data for the Forms stored*/
109     public List<ConfigAttribute> csvParser(){
125
126     /* This method create collection of FormConfiguration objects by using the
   "csvParser" Method
127     * There are several FormConfiguration objects will be created depend on the
   contents of the text file.
128     * If there are 5 form definitions defined in the file, there will be 5
   FormConfiguration
```

FormFactory.java

```
129     * objects will be created
130     * */
131     public List<FormConfiguration> createFormConfigurators (){
259
260     /* This method read the option values  for different fields (Radio
    buttons,Combobox etc) from
261     * the text file. It read as a pair from the "intArray" . A pair (1,11) tell
    the method to read the contents
262     * of the "Options1" attribute(ConfigAttribute) as a List and assign to "Field
    11" */
263     public Hashtable<Integer,List<String>> getOptionsValues(int[]
    intArray,List<ConfigAttribute> configAttributes,int formID){
284
285     /*This will Loop through the collection of FormConfiguration objects and
    create the appropriate
286     * Forms(Questionnaire,Serach,Logon,Participant etc) using the
    FormConfiguration object(form Data)*/
287     public Map<String,GenericForm> createForms(){
298
299     /* This will create the FormConfiguration of the form Name that is passed as a
    parameter
300     * For an example , if "Participant" is passed as a form Name it gives the
    FormConfiguration Object
301     * for that form */
302     public FormConfiguration createFormConfiguration(String formName){
313
314 }
315
```