

```
1:  /*
2:
3:  Jared Dyreson
4:  CWID: 889546529
5:  ZipBackend.java ->
6:
7:  */
8:
9:  import java.io.*;
10: import java.util.*;
11: import java.util.zip.*;
12:
13: public class ZipBackend {
14:     // Auto generated with caffeine and accounts-daemon.service
15:
16:     private ZipOutputStream zip_stream_for_class;
17:     private String path = "";
18:
19:     //public ZipBackend(String zip_path){
20:         //this.path = zip_path;
21:     //}
22:
23:     public void zip_contents(List<String> file_contents, String path_to_file) throws IOException{
24:         FileOutputStream file_stream = new FileOutputStream(path_to_file);
25:         ZipOutputStream zip_out = new ZipOutputStream(file_stream);
26:         File destination_zip = new File(path_to_file);
27:         //if(destination_zip.exists()){
28:             //System.out.println("File already exists");
29:             //return;
30:         //}
31:
32:         for(String path : file_contents){
33:             File zip_this_file = new File(path);
34:             FileInputStream input_stream = new FileInputStream(zip_this_file);
35:             ZipEntry zip_entry = new ZipEntry(zip_this_file.getName());
36:             zip_out.putNextEntry(zip_entry);
37:
38:             byte[] bytes = new byte[1024];
39:             int len;
40:             while((len = input_stream.read(bytes)) >= 0){
41:                 zip_out.write(bytes, 0, len);
42:             }
43:             input_stream.close();
44:         }
45:         zip_out.close();
46:         file_stream.close();
47:     }
48:
```

```
49:      public void create_empty_archive(String path_to_file) throws IOException{
50:          File zip_file = new File(path_to_file);
51:          FileOutputStream stream = new FileOutputStream(zip_file);
52:          BufferedOutputStream buffered = new BufferedOutputStream(stream);
53:          ZipOutputStream zip_stream = new ZipOutputStream(stream);
54:
55:          //ZipEntry fake_entry = new ZipEntry("fake_meta");
56:          //zip_stream.putNextEntry(fake_entry);
57:          //zip_stream.closeEntry();
58:          zip_stream.close();
59:      }
60:
61:      //public void dump_contents(){
62:
63:      //}
64:
65:      //public void list_contents(String path_to_archive){
66:          ///// this will only print with depth of 1
67:
68:
69:      //}
70: }
```