

```

01 package DBLayer;
02 import java.sql.*;
03 import java.util.ArrayList;
04 import ModelLayer.*;
05
06 public class DBMink implements IFDBMink
07 {
08     private Connection con;
09
10     public int createMink(Mink mink) throws Exception
11     {
12         int rc = -1;
13         String query = "INSERT INTO mfMink(color, furLength,
14 furDensity, birthdate, qualityType, cageNr) VALUES ('"+
15 mink.getColor() + "', '" +
16 mink.getFurLength() + "', '" +
17 mink.getFurDensity() + "', '" +
18 mink.getBirthDate() + "', '" +
19 mink.getQualityType() + "', '" +
20 mink.getCageNr() + "')";
21         System.out.println("insert : " + query);
22
23         try
24         {
25             Statement stmt = con.createStatement();
26             stmt.setQueryTimeout(5);
27             rc = stmt.executeUpdate(query);
28             stmt.close();
29         }
30         catch(SQLException ex)
31         {
32             System.out.println("Mink er ikke oprettet");
33             throw new Exception("Mink er ikke tilføjet");
34         }
35         return (rc);
36     }
37
38     public int updateMink(Mink mink)
39     {
40         Mink minkobj = mink;
41         int rc= -1;
42
43         String qry = "UPDATE mfMink SET" +
44 "color = '" + minkobj.getColor()+"', " +
45 "furLength ='" + minkobj.getFurLength() + "', " +
46 "furDensity ='" + minkobj.getFurDensity() + "', " +
47 "birthdate ='" + minkobj.getBirthDate() + "', " +
48 "qualityType ='" + minkobj.getQualityType() + "', " +
49 "cageNr ='" + minkobj.getCageNr() + "',";
50
51
52         System.out.println("Update query: " + qry);
53         try
54         {
55             Statement stmt = con.createStatement();
56             stmt.setQueryTimeout(5);
57             rc = stmt.executeUpdate(qry);
58
59             stmt.close();
60         }
61         catch(Exception ex)
62         {
63             System.out.println("Update exception in Mink db:
64 " + ex)

```

```

64         ;
65     }
66     return(rc);
67 }
68
69 public int deleteMink(int cageNr)
70 {
71     int rc = -1;
72
73     String qry = "DELETE FROM mfMink WHERE phoneno = '" +
74     cageNr + "'";
75     System.out.println(qry);
76     try
77     {
78         Statement stmt = con.createStatement();
79         stmt.setQueryTimeout(5);
80         rc = stmt.executeUpdate(qry);
81         stmt.close();
82     }
83     catch(Exception ex)
84     {
85         System.out.println("Delete exception in Mink db: "+ex)
86         ;
87     }
88     return(rc);
89 }
90
91 public String buildQuery(String wClause)
92 {
93     String qry = "SELECT color, furLength, furDensity,
94     birthDate, qualityType, cageNr";
95     if(wClause.length() > 0)
96         qry = qry + "WHERE " + wClause;
97     return qry;
98 }
99
100 public Mink singleWhere(String wClause, boolean
101 retrieveAssociation)
102 {
103     ResultSet results;
104     Mink minkobj = new Mink();
105     String query = buildQuery(wClause);
106     System.out.println(query);
107
108     try
109     {
110         Statement stmt = con.createStatement();
111         stmt.setQueryTimeout(5);
112         results = stmt.executeQuery(query);
113
114         if(results.next())
115         {
116             minkobj = buildMink(results);
117         }
118         stmt.close();
119     }
120     catch(Exception e)
121     {
122         System.out.println("Query exception: " + e);
123     }
124     return minkobj;
125 }
126
127 private Mink buildMink(ResultSet results)

```

```
128 {
129     Mink minkObj = new Mink();
130     try
131     {
132         minkObj.setColor(results.getString("color"));
133         minkObj.setFurLength(results.getInt("furLength"));
134         minkObj.setFurDensity(results.getInt("furDensity"));
135         minkObj.setBirthDate(results.getString("birthDate"));
136         minkObj.setQualityType(results.getString("qualityType"));
137         minkObj.setCageNr(results.getInt("cageNr"));
138     }
139     catch(Exception e)
140     {
141         System.out.println("Error in building cage");
142     }
143     return minkObj;
144 }
145
146 }
147
148 }
```