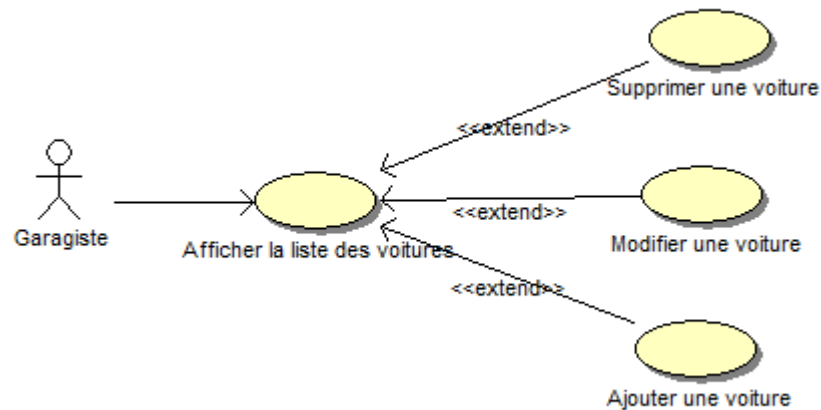


NFE114 : TD Java Web

Sujet : Un garagiste souhaite créer un système d'information lui permettant, entre autre, de gérer la liste des voitures en cours de réparation.

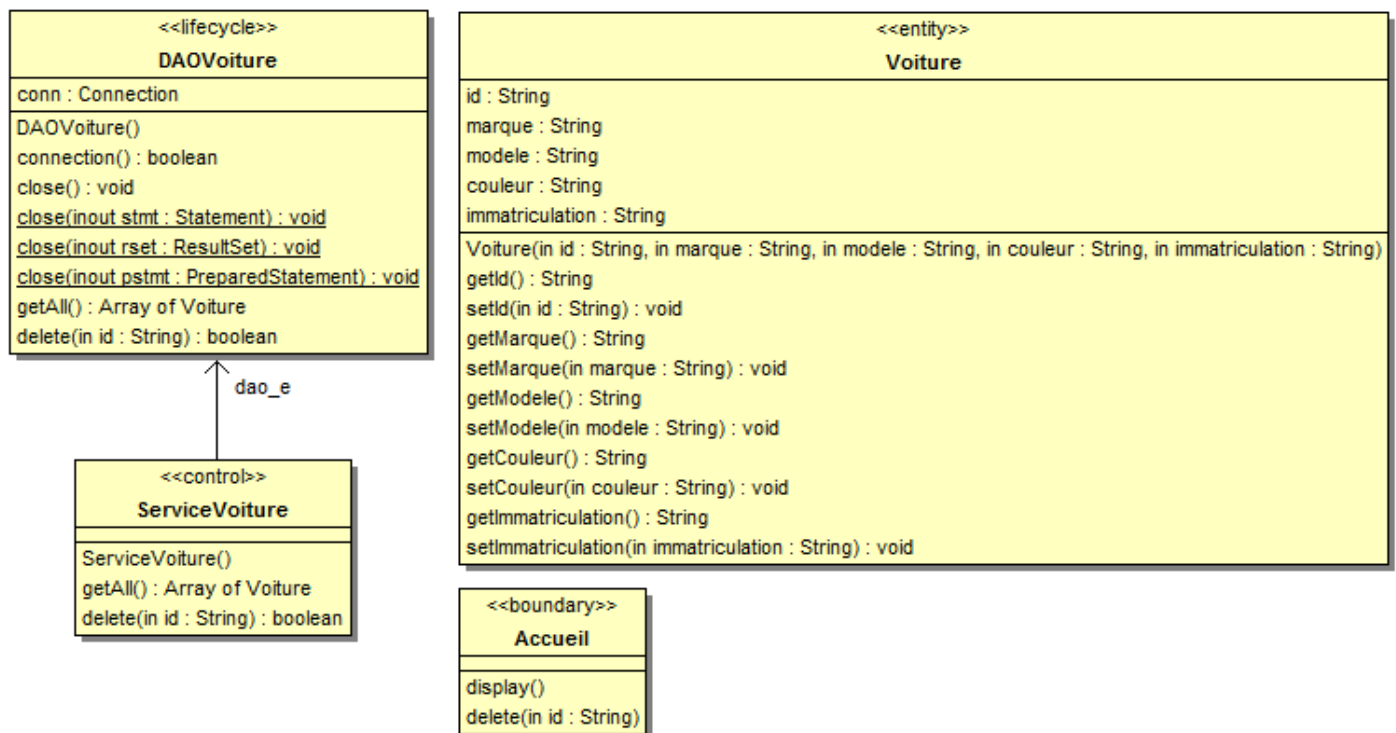
Son système comportera une base de données 'garage' comportant notamment une table 'voiture' précisant la marque, le modèle, la couleur et le numéro d'immatriculation de chaque voiture.

Soit le diagramme des cas d'utilisation suivant :



Pour les questions qui suivent, on ne traitera que les cas d'utilisation « Afficher la liste des voitures » et « Supprimer une voiture ».

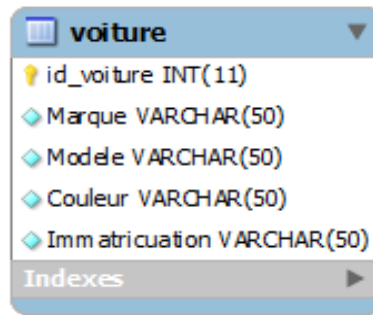
On propose le diagramme des classes suivant :



Question 1)

Proposer un exemple de schéma pour la table 'voiture' et le script SQL permettant de la créer.

Réponse :



```
CREATE TABLE IF NOT EXISTS `voiture` (  
  `id_voiture` int(11) NOT NULL AUTO_INCREMENT,  
  `Marque` varchar(50) CHARACTER SET utf8 NOT NULL,  
  `Modele` varchar(50) CHARACTER SET utf8 NOT NULL,  
  `Couleur` varchar(50) CHARACTER SET utf8 NOT NULL,  
  `Immatriculation` varchar(50) CHARACTER SET utf8 NOT NULL,  
  PRIMARY KEY (`id_voiture`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;
```

Question 2)

Écrire le code de la classe entité 'Voiture':

Réponse :

```
public class Voiture{
    private String id;
    private String marque;
    private String modele;
    private String couleur;
    private String immatriculation;

    public Voiture(String id, String marque, String modele, String couleur, String
immatriculation) {
        this.id=id;
        this.marque = marque;
        this.modele = modele;
        this.couleur = couleur;
        this.immatriculation = immatriculation;
    }

    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getMarque() {
        return marque;
    }
    public void setMarque(String marque) {
        this.marque = marque;
    }
    public String getModele() {
        return modele;
    }
    public void setModele(String modele) {
        this.modele = modele;
    }
    public String getCouleur() {
        return couleur;
    }
    public void setCouleur(String couleur) {
        this.couleur = couleur;
    }
    public String getImmatriculation() {
        return immatriculation;
    }
    public void setImmatriculation(String immatriculation) {
        this.immatriculation = immatriculation;
    }
}
```

Question 3)

Écrire le code de l'objet d'accès aux données (DAO) 'DAOVoiture' (on ne traitera pas les méthodes close()).

Réponse :

```
public class DAOVoiture {
    private Connection conn;
    public DAOVoiture() {
        conn = null;
    }

    public boolean connection() throws ClassNotFoundException, InstantiationException,
    IllegalAccessException, SQLException {
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        DriverManager.setLoginTimeout(10);
        conn = DriverManager.getConnection("jdbc:mysql://localhost/garage", "root", "");
        return true;
    }

    public ArrayList<Voiture> getAll() {
        Statement stmt = null;
        ResultSet rset = null;
        ArrayList<Voiture> al = new ArrayList<Voiture>();
        Voiture v;
        try {
            connection();
            stmt = conn.createStatement();
            rset = stmt.executeQuery("SELECT * FROM voiture ORDER BY Marque");
            while(rset.next()) {
                v=new Voiture(rset.getString(1), rset.getString(2), rset.getString(3),
                rset.getString(4),rset.getString(5));
                al.add(v);
            }
        } catch (Exception e) {
            System.err.println(e.getMessage());
            close(rset); // fonction à écrire
            close(stmt); // fonction à écrire
            close(); // fonction à écrire
            return null;
        }
        close(rset);
        close(stmt);
        close();
        return al;
    }

    public boolean delete(String id) {
        PreparedStatement pstmt = null;
        try {
            connection();
            pstmt = conn.prepareStatement("DELETE FROM voiture WHERE id_voiture=?");
            pstmt.setString(1, id);
            pstmt.executeUpdate();
        } catch (Exception e) {
            System.err.println(e.getMessage());
            close(pstmt); // fonction à écrire
            close();
            return false;
        }
        close(pstmt);
        close();
        return true;
    }
}
```

Question 4)

Écrire le code de la classe control 'ServiceVoiture' correspondant.

Réponse :

```
public class ServiceVoiture {
    protected DAOVoiture dao_e;

    public ServiceVoiture() {
        dao_e=new DAOVoiture();
    }

    public ArrayList<Voiture> getAll() {
        ArrayList<Voiture> arr_v = dao_e.getAll();
        return arr_v;
    }

    public boolean delete(String id) {
        return dao_e.delete(id);
    }
}
```

Question 5)

Écrire la page d'accueil, affichant la liste des voitures et permettant d'en supprimer, sous la forme d'une JSP.

Réponse :

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">

<%@page import="fr.cnam.garage.entity.Voiture"%>
<%@page import="fr.cnam.garage.service.ServiceVoiture" %>
<%@page import="java.util.ArrayList"%>

<%! ServiceVoiture service_v=new ServiceVoiture(); %>
<%
String action= request.getParameter("action");
if((action!=null)&&(action.equals("delete"))){
    String id= request.getParameter("id");
    service_v.delete(id);
}
%>
<%! ArrayList<Voiture> al=service_v.getAll(); %>

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Table Voiture</title>
    </head>
    <body>
        <h1>Affichage Table Voiture</h1>
        <table border="1">
            <thead>
                <tr><th>ID</th><th>Marque</th><th>Modele</th><th>Couleur</th><th>Immatriculation</th><th>Commandes</th></tr>
            </thead>
            <%
for(Voiture v : al) {
%>

                <tr>
                    <td><%= v.getId() %></td><td><%= v.getMarque() %></td><td><%=
v.getModele() %></td><td><%= v.getCouleur() %></td><td><%= v.getImmatriculation() %></td>
                    <td>
                        <form method="POST" action="">
                            <input type="hidden" name="id" value="<%= v.getId() %>" />
                            <input type="hidden" name="action" value="delete" />
                            <input type="submit" value="Supprimer" name="Send" />
                        </form>
                    </td>
                </tr>
            <% } %>
        </table>
    </body>
</html>
```