/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package cta;

import java.io.Serializable;

import java.util.Collection;

import java.util.Date;

import javax.persistence.Basic;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.NamedQueries;

import javax.persistence.NamedQuery;

import javax.persistence.OneToMany;

import javax.persistence.Table;

import javax.persistence.Temporal;

import javax.persistence.TemporalType;

import javax.validation.constraints.NotNull;

import javax.validation.constraints.Size;

import javax.xml.bind.annotation.XmlRootElement;

import javax.xml.bind.annotation.XmlTransient;

/\*\*

\*

\* @author nafiy

\*/

@Entity

@Table(name = "USERS")

@XmlRootElement

@NamedQueries({

@NamedQuery(name = "Users.findAll", query = "SELECT u FROM Users u")

, @NamedQuery(name = "Users.findByUserId", query = "SELECT u FROM Users u WHERE u.userId = :userId")

, @NamedQuery(name = "Users.findByUserName", query = "SELECT u FROM Users u WHERE u.userName = :userName")

, @NamedQuery(name = "Users.findByUserSurname", query = "SELECT u FROM Users u WHERE u.userSurname = :userSurname")

, @NamedQuery(name = "Users.findByUserEmail", query = "SELECT u FROM Users u WHERE u.userEmail = :userEmail")

, @NamedQuery(name = "Users.findByUserAddress", query = "SELECT u FROM Users u WHERE u.userAddress = :userAddress")

, @NamedQuery(name = "Users.findByUserPostcode", query = "SELECT u FROM Users u WHERE u.userPostcode = :userPostcode")

, @NamedQuery(name = "Users.findByUserGender", query = "SELECT u FROM Users u WHERE u.userGender = :userGender")

, @NamedQuery(name = "Users.findByUserActivelevel", query = "SELECT u FROM Users u WHERE u.userActivelevel = :userActivelevel")

, @NamedQuery(name = "Users.findByUserSteppermile", query = "SELECT u FROM Users u WHERE u.userSteppermile = :userSteppermile")

, @NamedQuery(name = "Users.findByUserWeight", query = "SELECT u FROM Users u WHERE u.userWeight = :userWeight")

, @NamedQuery(name = "Users.findByUserHeight", query = "SELECT u FROM Users u WHERE u.userHeight = :userHeight")

, @NamedQuery(name = "Users.findByUserDob", query = "SELECT u FROM Users u WHERE u.userDob = :userDob")})

public class Users implements Serializable {

private static final long serialVersionUID = 1L;

@Id

@Basic(optional = false)

@NotNull

@Column(name = "USER\_ID")

private Short userId;

@Size(max = 10)

@Column(name = "USER\_NAME")

private String userName;

@Size(max = 10)

@Column(name = "USER\_SURNAME")

private String userSurname;

@Size(max = 25)

@Column(name = "USER\_EMAIL")

private String userEmail;

@Size(max = 30)

@Column(name = "USER\_ADDRESS")

private String userAddress;

@Size(max = 7)

@Column(name = "USER\_POSTCODE")

private String userPostcode;

@Size(max = 1)

@Column(name = "USER\_GENDER")

private String userGender;

@Column(name = "USER\_ACTIVELEVEL")

private Short userActivelevel;

@Column(name = "USER\_STEPPERMILE")

private Short userSteppermile;

// @Max(value=?) @Min(value=?)//if you know range of your decimal fields consider using these annotations to enforce field validation

@Column(name = "USER\_WEIGHT")

private Double userWeight;

@Column(name = "USER\_HEIGHT")

private Double userHeight;

@Column(name = "USER\_DOB")

@Temporal(TemporalType.DATE)

private Date userDob;

@OneToMany(mappedBy = "creUserid")

private Collection<Credential> credentialCollection;

@OneToMany(mappedBy = "repUserid")

private Collection<Report> reportCollection;

@OneToMany(mappedBy = "conUserid")

private Collection<Consumption> consumptionCollection;

public Users() {

}

public Users(Short userId) {

this.userId = userId;

}

public Short getUserId() {

return userId;

}

public void setUserId(Short userId) {

this.userId = userId;

}

public String getUserName() {

return userName;

}

public void setUserName(String userName) {

this.userName = userName;

}

public String getUserSurname() {

return userSurname;

}

public void setUserSurname(String userSurname) {

this.userSurname = userSurname;

}

public String getUserEmail() {

return userEmail;

}

public void setUserEmail(String userEmail) {

this.userEmail = userEmail;

}

public String getUserAddress() {

return userAddress;

}

public void setUserAddress(String userAddress) {

this.userAddress = userAddress;

}

public String getUserPostcode() {

return userPostcode;

}

public void setUserPostcode(String userPostcode) {

this.userPostcode = userPostcode;

}

public String getUserGender() {

return userGender;

}

public void setUserGender(String userGender) {

this.userGender = userGender;

}

public Short getUserActivelevel() {

return userActivelevel;

}

public void setUserActivelevel(Short userActivelevel) {

this.userActivelevel = userActivelevel;

}

public Short getUserSteppermile() {

return userSteppermile;

}

public void setUserSteppermile(Short userSteppermile) {

this.userSteppermile = userSteppermile;

}

public Double getUserWeight() {

return userWeight;

}

public void setUserWeight(Double userWeight) {

this.userWeight = userWeight;

}

public Double getUserHeight() {

return userHeight;

}

public void setUserHeight(Double userHeight) {

this.userHeight = userHeight;

}

public Date getUserDob() {

return userDob;

}

public void setUserDob(Date userDob) {

this.userDob = userDob;

}

@XmlTransient

public Collection<Credential> getCredentialCollection() {

return credentialCollection;

}

public void setCredentialCollection(Collection<Credential> credentialCollection) {

this.credentialCollection = credentialCollection;

}

@XmlTransient

public Collection<Report> getReportCollection() {

return reportCollection;

}

public void setReportCollection(Collection<Report> reportCollection) {

this.reportCollection = reportCollection;

}

@XmlTransient

public Collection<Consumption> getConsumptionCollection() {

return consumptionCollection;

}

public void setConsumptionCollection(Collection<Consumption> consumptionCollection) {

this.consumptionCollection = consumptionCollection;

}

@Override

public int hashCode() {

int hash = 0;

hash += (userId != null ? userId.hashCode() : 0);

return hash;

}

@Override

public boolean equals(Object object) {

// TODO: Warning - this method won't work in the case the id fields are not set

if (!(object instanceof Users)) {

return false;

}

Users other = (Users) object;

if ((this.userId == null && other.userId != null) || (this.userId != null && !this.userId.equals(other.userId))) {

return false;

}

return true;

}

@Override

public String toString() {

return "cta.Users[ userId=" + userId + " ]";

}

}