/\*

\* 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 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.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 = "FOOD")

@XmlRootElement

@NamedQueries({

@NamedQuery(name = "Food.findAll", query = "SELECT f FROM Food f")

, @NamedQuery(name = "Food.findByFoodId", query = "SELECT f FROM Food f WHERE f.foodId = :foodId")

, @NamedQuery(name = "Food.findByFoodName", query = "SELECT f FROM Food f WHERE f.foodName = :foodName")

, @NamedQuery(name = "Food.findByFoodCategory", query = "SELECT f FROM Food f WHERE f.foodCategory = :foodCategory")

, @NamedQuery(name = "Food.findByFoodCalorie", query = "SELECT f FROM Food f WHERE f.foodCalorie = :foodCalorie")

, @NamedQuery(name = "Food.findByFoodSeverunit", query = "SELECT f FROM Food f WHERE f.foodSeverunit = :foodSeverunit")

, @NamedQuery(name = "Food.findByFoodServeamount", query = "SELECT f FROM Food f WHERE f.foodServeamount = :foodServeamount")

, @NamedQuery(name = "Food.findByFoodFat", query = "SELECT f FROM Food f WHERE f.foodFat = :foodFat")})

public class Food implements Serializable {

private static final long serialVersionUID = 1L;

@Id

@Basic(optional = false)

@NotNull

@Column(name = "FOOD\_ID")

private Short foodId;

@Size(max = 25)

@Column(name = "FOOD\_NAME")

private String foodName;

@Size(max = 15)

@Column(name = "FOOD\_CATEGORY")

private String foodCategory;

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

@Column(name = "FOOD\_CALORIE")

private Double foodCalorie;

@Size(max = 10)

@Column(name = "FOOD\_SEVERUNIT")

private String foodSeverunit;

@Column(name = "FOOD\_SERVEAMOUNT")

private Double foodServeamount;

@Column(name = "FOOD\_FAT")

private Double foodFat;

@OneToMany(mappedBy = "conFoodid")

private Collection<Consumption> consumptionCollection;

public Food() {

}

public Food(Short foodId) {

this.foodId = foodId;

}

public Short getFoodId() {

return foodId;

}

public void setFoodId(Short foodId) {

this.foodId = foodId;

}

public String getFoodName() {

return foodName;

}

public void setFoodName(String foodName) {

this.foodName = foodName;

}

public String getFoodCategory() {

return foodCategory;

}

public void setFoodCategory(String foodCategory) {

this.foodCategory = foodCategory;

}

public Double getFoodCalorie() {

return foodCalorie;

}

public void setFoodCalorie(Double foodCalorie) {

this.foodCalorie = foodCalorie;

}

public String getFoodSeverunit() {

return foodSeverunit;

}

public void setFoodSeverunit(String foodSeverunit) {

this.foodSeverunit = foodSeverunit;

}

public Double getFoodServeamount() {

return foodServeamount;

}

public void setFoodServeamount(Double foodServeamount) {

this.foodServeamount = foodServeamount;

}

public Double getFoodFat() {

return foodFat;

}

public void setFoodFat(Double foodFat) {

this.foodFat = foodFat;

}

@XmlTransient

public Collection<Consumption> getConsumptionCollection() {

return consumptionCollection;

}

public void setConsumptionCollection(Collection<Consumption> consumptionCollection) {

this.consumptionCollection = consumptionCollection;

}

@Override

public int hashCode() {

int hash = 0;

hash += (foodId != null ? foodId.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 Food)) {

return false;

}

Food other = (Food) object;

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

return false;

}

return true;

}

@Override

public String toString() {

return "cta.Food[ foodId=" + foodId + " ]";

}

}