

**PROGRAMA AGENDA DE ALUMNOS XML**

**ASIGNATURA:**

Programación avanzada orientada a objetos

**CATEDRÁTICO:**

José Juan Hernández Mora

**ALUMNO:**

José Montoya Guzmán

**FECHA DE ASIGNACIÓN:**

7 de noviembre de 2016

**FECHA DE ENTREGA**:

21 de noviembre de 2016

**CÓDIGO**

**CLASE ES**

1 class ES {  
 2 static void escribe (String cad){  
 3   
 4 javax.swing.JOptionPane.showMessageDialog(null, cad);  
 5 }  
 6   
 7   
 8 int leeI(String cad){  
 9 int n;  
10 n=Integer.parseInt(javax.swing.JOptionPane.showInputDialog(cad));  
11 return n;  
12 }  
13   
14 String leeC (String cad){  
15 String n;  
16 n=javax.swing.JOptionPane.showInputDialog(cad);  
17 return n;  
18 }  
19   
20 Double leeD (String cad){  
21 double n;  
22 n=Double.parseDouble(javax.swing.JOptionPane.showInputDialog(cad));  
23 return n;  
24 }  
25 }

**CLASE ESCUELA**

1 import java.util.ArrayList;   
 2 import javax.xml.bind.annotation.\*;  
 3   
 4 @XmlRootElement(name = "escuela")  
 5 public class Escuela {  
 6   
 7 private String clave;  
 8 private String nombre;  
 9 private ArrayList<Alumnos> alumnos;  
10   
11 public String getClave() {  
12 return clave;  
13 }  
14   
15 public void setClave(String clave) {  
16 this.clave = clave;  
17 }  
18   
19 public String getNombre() {  
20 return nombre;  
21 }  
22   
23 public void setNombre(String nombre) {  
24 this.nombre = nombre;  
25 }  
26   
27 @XmlElementWrapper(name = "alumnos")  
28 @XmlElement(name = "alumno")  
29 public ArrayList<Alumnos> getAlumnos() {  
30 return alumnos;  
31 }  
32   
33 public void setAlumnos(ArrayList<Alumnos> alumnos) {  
34 this.alumnos = alumnos;  
35 }  
36 }

**CLASE ALUMNOS**

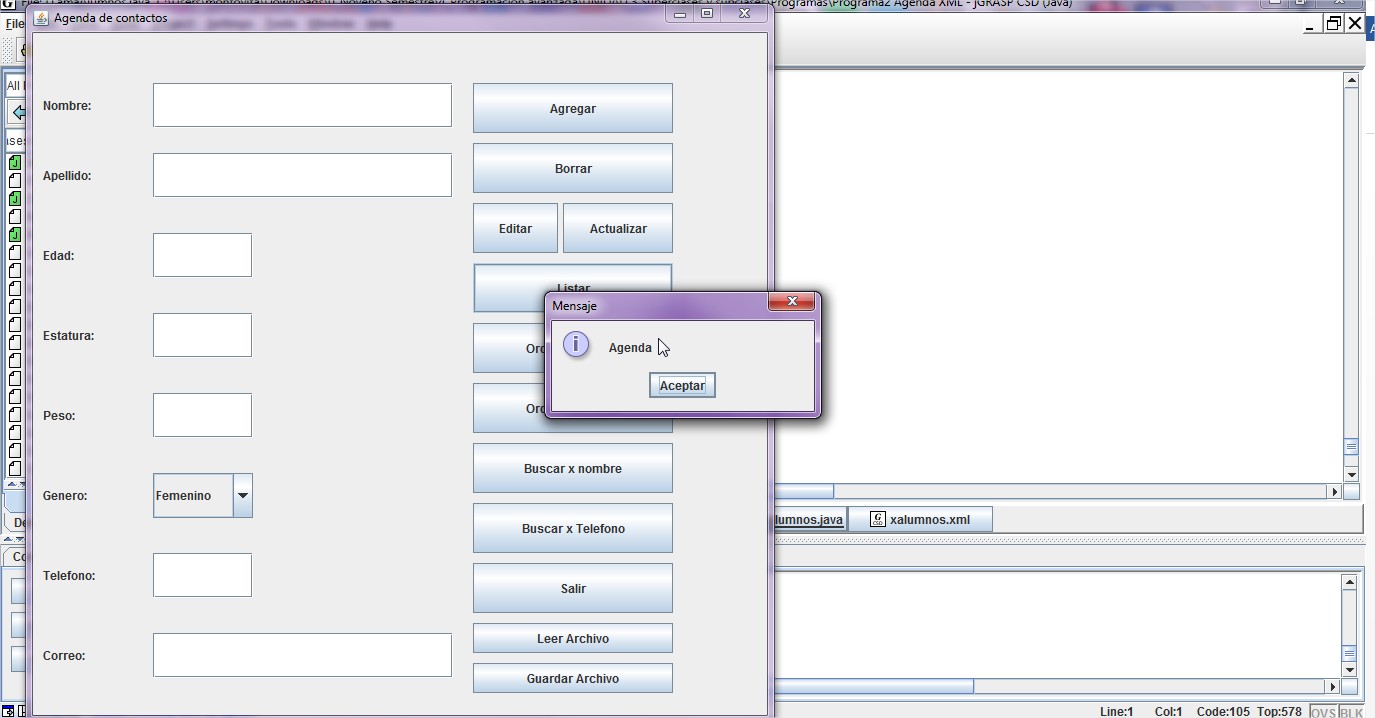
1 import javax.xml.bind.annotation.\*;  
 2   
 3 @XmlRootElement(name = "alumno")  
 4 //@XmlType(propOrder = {"id","nombre","apellido", "edad","estatura","peso","genero","telefono","correo" } )  
 5 public class Alumnos{  
 6   
 7 private int id;  
 8 private String nombre;  
 9 private String apellido;  
 10 private int edad;  
 11 private double estatura;  
 12 private int peso;  
 13 private String genero;  
 14 private String telefono;  
 15 private String correo;  
 16   
 17 Alumnos(){  
 18 id=1;  
 19 nombre="";  
 20 apellido="";  
 21 edad=0;  
 22 estatura=0.0;  
 23 peso=0;  
 24 genero="";   
 25 telefono="";  
 26 correo="";  
 27 }  
 28   
 29 Alumnos(int id, String nombre, String apellido, int edad, double estatura,int peso, String genero, String telefono, String correo){  
 30 this.id=id;  
 31 this.nombre=nombre;  
 32 this.apellido=apellido;  
 33 this.edad=edad;  
 34 this.estatura=estatura;  
 35 this.peso=peso;  
 36 this.genero=genero;  
 37 this.telefono=telefono;  
 38 this.correo=correo;  
 39 }  
 40   
 41 @XmlElement  
 42 void set\_Id(int id){   
 43 id=id;  
 44 }  
 45   
 46 @XmlElement  
 47 void set\_Nombre(String n){   
 48 nombre=n;  
 49 }  
 50   
 51 @XmlElement  
 52 void set\_Apellido(String a){   
 53 apellido=a;  
 54 }  
 55   
 56 @XmlElement  
 57 void set\_Edad(int e){   
 58 edad=e;  
 59 }  
 60   
 61 @XmlElement  
 62 void set\_Estatura(double e){   
 63 estatura=e;  
 64 }  
 65   
 66 @XmlElement  
 67 void set\_Peso(int p){   
 68 peso=p;  
 69 }  
 70   
 71 @XmlElement  
 72 void set\_Genero(String g){   
 73 genero=g;  
 74 }  
 75   
 76 @XmlElement  
 77 void set\_Telefono(String t){   
 78 telefono=t;  
 79 }  
 80   
 81 @XmlElement  
 82 void set\_Correo(String c){   
 83 correo=c;  
 84 }  
 85   
 86   
 87 int get\_Id(){  
 88 return id;  
 89 }  
 90   
 91 String get\_Nombre(){  
 92 return nombre;  
 93 }  
 94   
 95 String get\_Apellido(){  
 96 return apellido;  
 97 }  
 98   
 99 int get\_Edad(){  
100 return edad;  
101 }  
102   
103 double get\_Estatura(){  
104 return estatura;  
105 }  
106   
107 int get\_Peso(){  
108 return peso;  
109 }  
110   
111 String get\_Genero(){  
112 return genero;  
113 }  
114   
115 String get\_Telefono(){  
116 return telefono;  
117 }  
118   
119 String get\_Correo(){  
120 return correo;  
121 }  
122   
123   
124 String despliega(){  
125 return id+". "+"Nombre: "+nombre+  
126 " Apellido: "+apellido+  
127 " Edad: "+edad+  
128 " Estatura: "+estatura+  
129 " Peso: "+peso+  
130 " Genero: "+genero+  
131 " Telefono: "+telefono+  
132 " Correo: "+correo+"\n";  
133 }  
134 }

**CLASE LLAMAALUMNOS**

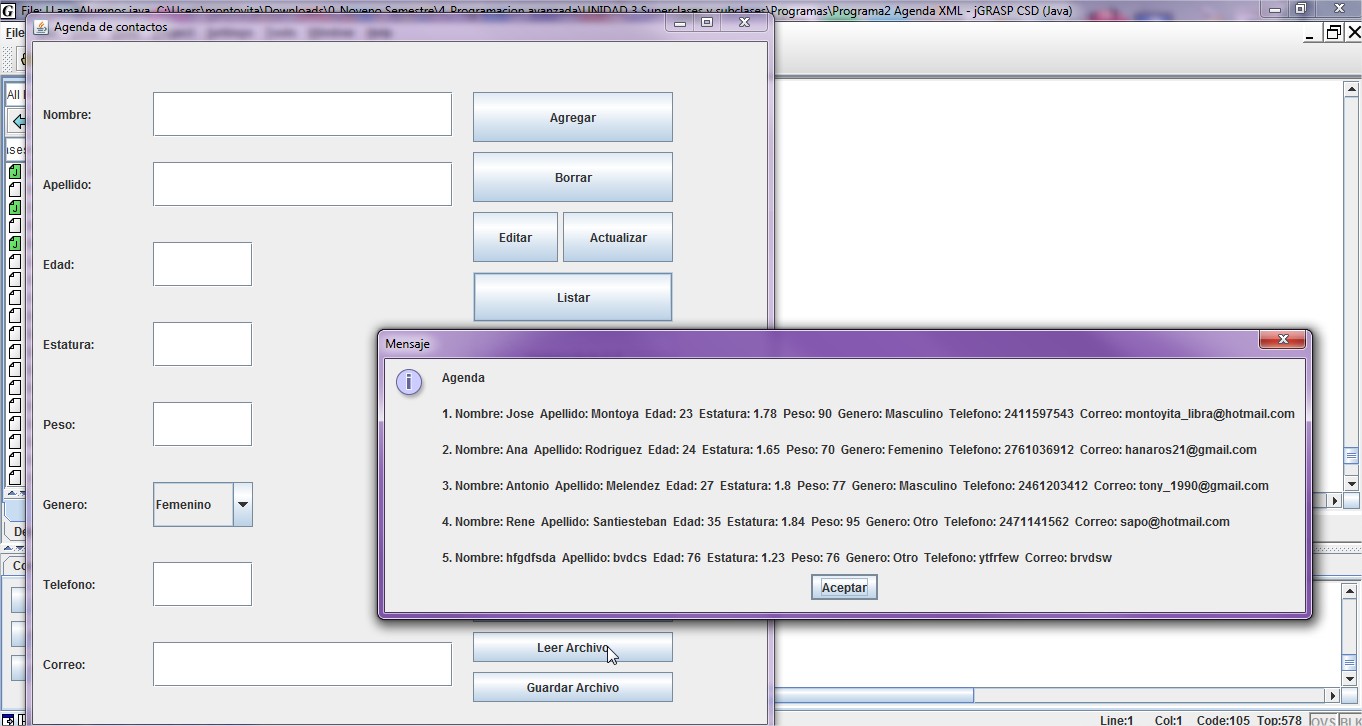
1 import java.awt.\*;  
 2 import java.awt.event.\*;  
 3 import javax.swing.\*;  
 4 import java.util.\*;  
 5 import java.io.\*;  
 6 import java.io.File;  
 7 import javax.xml.bind.\*;  
 8   
 9 public class LLamaAlumnos extends JFrame{  
 10   
 11 Alumnos alumno1 = new Alumnos();  
 12 int idv=1;  
 13 String nombrev="";  
 14 String apellidov="";  
 15 int edadv=0;  
 16 double estaturav=0.0;  
 17 int pesov=0;  
 18 String generov="";  
 19 String telefonov="";  
 20 String correov="";  
 21   
 22 int max;  
 23 ArrayList listaA;  
 24   
 25   
 26 private JLabel etiquetaNombre;  
 27 private JLabel etiquetaApellido;  
 28 private JLabel etiquetaEdad;  
 29 private JLabel etiquetaEstatura;  
 30 private JLabel etiquetaPeso;  
 31 private JLabel etiquetaGenero;  
 32 private JLabel etiquetaTelefono;  
 33 private JLabel etiquetaCorreo;  
 34   
 35 private JTextField nombre;  
 36 private JTextField apellido;  
 37 private JTextField edad;  
 38 private JTextField estatura;  
 39 private JTextField peso;  
 40 private JComboBox comboGenero;  
 41 private JTextField telefono;  
 42 private JTextField correo;  
 43   
 44 private JButton agregar;  
 45 private JButton borrar;  
 46 private JButton editar;  
 47 private JButton actualizar;  
 48 private JButton listar;  
 49 private JButton ordenarNombres;  
 50 private JButton ordenarEdades;  
 51 private JButton buscarNombre;  
 52 private JButton buscarTelefono;  
 53 private JButton salir;   
 54 private JButton leerArchivo;  
 55 private JButton guardarArchivo;  
 56   
 57 LLamaAlumnos()  
 58 {  
 59 super("Agenda de contactos");  
 60 Container co=getContentPane();  
 61 co.setLayout(null);  
 62   
 63 setSize(750,720);  
 64 setVisible(true);  
 65   
 66 listaA=new ArrayList();  
 67 max=0;  
 68   
 69 etiquetaNombre=new JLabel("Nombre: ");  
 70 etiquetaApellido=new JLabel("Apellido: ");  
 71 etiquetaEdad=new JLabel("Edad: ");  
 72 etiquetaEstatura=new JLabel("Estatura: ");  
 73 etiquetaPeso=new JLabel("Peso: ");  
 74 etiquetaGenero=new JLabel("Genero: ");  
 75 etiquetaTelefono=new JLabel("Telefono: ");  
 76 etiquetaCorreo=new JLabel("Correo: ");  
 77   
 78   
 79 nombre=new JTextField(20);  
 80 apellido=new JTextField(40);  
 81 edad=new JTextField(3);  
 82 estatura=new JTextField(5);  
 83 peso=new JTextField(2);  
 84 comboGenero=new JComboBox();  
 85 telefono=new JTextField(10);  
 86 correo=new JTextField(30);  
 87   
 88   
 89 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ETIQUETAS Y CAMPOS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 90   
 91 co.add(etiquetaNombre);  
 92 etiquetaNombre.setBounds(10,50,100,45);  
 93 co.add(nombre);  
 94 nombre.setBounds(120,50,300,45);  
 95   
 96 co.add(etiquetaApellido);  
 97 etiquetaApellido.setBounds(10,120,100,45);  
 98 co.add(apellido);  
 99 apellido.setBounds(120,120,300,45);  
100   
101 co.add(etiquetaEdad);  
102 etiquetaEdad.setBounds(10,200,100,45);  
103 co.add(edad);  
104 edad.setBounds(120,200,100,45);  
105   
106 co.add(etiquetaEstatura);  
107 etiquetaEstatura.setBounds(10,280,100,45);  
108 co.add(estatura);  
109 estatura.setBounds(120,280,100,45);  
110   
111 co.add(etiquetaPeso);  
112 etiquetaPeso.setBounds(10,360,100,45);  
113 co.add(peso);  
114 peso.setBounds(120,360,100,45);  
115   
116 co.add(etiquetaGenero);  
117 etiquetaGenero.setBounds(10,440,100,45);  
118 co.add(comboGenero);  
119 comboGenero.setBounds(120,440,100,45);  
120 comboGenero.addItem("Femenino");  
121 comboGenero.addItem("Masculino");  
122 comboGenero.addItem("Otro");  
123   
124 co.add(etiquetaTelefono);  
125 etiquetaTelefono.setBounds(10,520,100,45);   
126 co.add(telefono);  
127 telefono.setBounds(120,520,100,45);  
128   
129 co.add(etiquetaCorreo);  
130 etiquetaCorreo.setBounds(10,600,100,45);  
131 co.add(correo);  
132 correo.setBounds(120,600,300,45);  
133   
134 // INSTANCIANDO BOTONES  
135 agregar=new JButton("Agregar");  
136 co.add(agregar);  
137 agregar.setBounds(440,50,200,50);  
138   
139 borrar=new JButton("Borrar");  
140 co.add(borrar);  
141 borrar.setBounds(440,110,200,50);  
142   
143 editar=new JButton("Editar");  
144 co.add(editar);  
145 editar.setBounds(440,170,85,50);  
146   
147 actualizar=new JButton("Actualizar");  
148 co.add(actualizar);  
149 actualizar.setBounds(530,170,110,50);  
150   
151 listar=new JButton("Listar");  
152 co.add(listar);  
153 listar.setBounds(440,230,200,50);  
154   
155 ordenarNombres=new JButton("Ordena de la A-Z");  
156 co.add(ordenarNombres);  
157 ordenarNombres.setBounds(440,290,200,50);  
158   
159 ordenarEdades=new JButton("Ordena por Edad");  
160 co.add(ordenarEdades);  
161 ordenarEdades.setBounds(440,350,200,50);  
162   
163 buscarNombre=new JButton("Buscar x nombre");  
164 co.add(buscarNombre);  
165 buscarNombre.setBounds(440,410,200,50);  
166   
167 buscarTelefono=new JButton("Buscar x Telefono");  
168 co.add(buscarTelefono);  
169 buscarTelefono.setBounds(440,470,200,50);   
170   
171 salir=new JButton("Salir");  
172 co.add(salir);  
173 salir.setBounds(440,530,200,50);  
174   
175 leerArchivo=new JButton("Leer Archivo");  
176 co.add(leerArchivo);  
177 leerArchivo.setBounds(440,590,200,30);   
178   
179 guardarArchivo=new JButton("Guardar Archivo");  
180 co.add(guardarArchivo);  
181 guardarArchivo.setBounds(440,630,200,30);  
182   
183   
184 /\*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ LISTENERS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*/   
185   
186 agregar.addActionListener(  
187 new ActionListener() {  
188 public void actionPerformed(ActionEvent ev)  
189 {  
190 agrega();  
191 }  
192 }  
193 );  
194   
195 borrar.addActionListener(  
196 new ActionListener() {  
197 public void actionPerformed(ActionEvent ev)  
198 {  
199 borra();  
200 }  
201 }  
202 );  
203   
204 editar.addActionListener(  
205 new ActionListener() {  
206 public void actionPerformed(ActionEvent ev)  
207 {  
208 edita();  
209 }  
210 }  
211 );   
212   
213 actualizar.addActionListener(  
214 new ActionListener() {  
215 public void actionPerformed(ActionEvent ev)  
216 {  
217 actualiza();  
218 }  
219 }  
220 );  
221   
222 listar.addActionListener(  
223 new ActionListener() {  
224 public void actionPerformed(ActionEvent ev){   
225 lista();   
226 }  
227 }  
228 );  
229   
230 ordenarNombres.addActionListener(  
231 new ActionListener() {  
232 public void actionPerformed(ActionEvent ev)  
233 {  
234 ordenaNombres();  
235 }  
236 }  
237 );  
238   
239 ordenarEdades.addActionListener(  
240 new ActionListener() {  
241 public void actionPerformed(ActionEvent ev)  
242 {  
243 ordenaEdades();  
244 }  
245 }  
246 );  
247   
248 buscarNombre.addActionListener(  
249 new ActionListener() {  
250 public void actionPerformed(ActionEvent ev)  
251 {  
252 buscaNombre();  
253 }  
254 }  
255 );  
256   
257 buscarTelefono.addActionListener(  
258 new ActionListener() {  
259 public void actionPerformed(ActionEvent ev)  
260 {  
261 buscaTelefono();  
262 }  
263 }  
264 );   
265   
266 salir.addActionListener(  
267 new ActionListener() {  
268 public void actionPerformed(ActionEvent ev)  
269 {  
270 System.exit(0);  
271 }  
272 }  
273 );  
274   
275 leerArchivo.addActionListener(  
276 new ActionListener() {  
277 public void actionPerformed(ActionEvent ev)  
278 {  
279 try{  
280 leeDatos();  
281 }  
282 catch(JAXBException ec){  
283 System.out.print(ec);  
284 }  
285 catch(IOException ie){  
286 System.out.print(ie);  
287 }  
288   
289 }  
290 }  
291 );  
292   
293 guardarArchivo.addActionListener(  
294 new ActionListener() {  
295 public void actionPerformed(ActionEvent ev)  
296 {  
297 try{  
298 guardaDatos();  
299 }  
300 catch(JAXBException ec){  
301 System.out.print(ec);  
302 }  
303 catch(IOException ie){  
304 System.out.print(ie);  
305 }  
306 }   
307 }  
308 );  
309 }   
310   
311 //MAIN  
312 public static void main (String Ar[]){  
313 LLamaAlumnos ap=new LLamaAlumnos();  
314 ap.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  
315 }  
316   
317   
318 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_METODOS DE LA LISTA\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
319   
320 void agrega(){   
321   
322 for (int j=1; j<listaA.size(); j++){  
323 alumno1=(Alumnos)listaA.get(j);  
324 if (alumno1.get\_Id()==(idv)){  
325 idv++;  
326 }  
327 }   
328   
329 nombrev=nombre.getText();  
330 apellidov=apellido.getText();  
331 edadv=(int)Integer.parseInt(edad.getText());  
332 estaturav=(double)Double.parseDouble(estatura.getText());  
333 pesov=(int)Integer.parseInt(peso.getText());  
334 generov=(String)comboGenero.getSelectedItem();  
335 telefonov=telefono.getText();   
336 correov=correo.getText();  
337   
338 //System.out.println(idv);   
339 alumno1=new Alumnos(idv,nombrev,apellidov,edadv,estaturav,pesov,generov,telefonov,correov);  
340 listaA.add(alumno1);  
341 max++;  
342 idv++;  
343   
344 Collections.sort(listaA,  
345 new Comparator<Alumnos>() {  
346 @Override  
347 public int compare(Alumnos p1, Alumnos p2) {  
348 return new Integer(p1.get\_Id()).compareTo(new Integer(p2.get\_Id()));  
349 }  
350 });  
351 ES.escribe("Objeto agregado exitosamente!!");   
352 nombre.setText("");  
353 apellido.setText("");  
354 edad.setText("");  
355 estatura.setText("");  
356 peso.setText("");  
357 comboGenero.setSelectedIndex(0);  
358 telefono.setText("");  
359 correo.setText("");  
360   
361 }  
362   
363 void borra(){  
364 idv = (int)Integer.parseInt(JOptionPane.showInputDialog(null, "Ingresa Id a eliminar"));  
365 for (int j=0; j<listaA.size(); j++){  
366 alumno1=(Alumnos)listaA.get(j);  
367 if ((alumno1.get\_Id())==idv){  
368 listaA.remove(alumno1);  
369 max--;  
370 ES.escribe("Objeto eliminado exitosamente!!");  
371 }  
372 }  
373 }   
374   
375 void edita(){  
376 idv = (int)Integer.parseInt(JOptionPane.showInputDialog(null, "Ingresa ID de contacto que deseas editar"));  
377 for (int j=0; j<listaA.size(); j++){  
378 alumno1=(Alumnos)listaA.get(j);  
379 if ((alumno1.get\_Id())==idv){  
380 nombre.setText(alumno1.get\_Nombre());  
381 apellido.setText(alumno1.get\_Apellido());  
382 edad.setText(String.valueOf(alumno1.get\_Edad()));  
383 estatura.setText(String.valueOf((alumno1.get\_Estatura())));  
384 peso.setText(String.valueOf(alumno1.get\_Peso()));  
385 comboGenero.setSelectedItem(String.valueOf(alumno1.get\_Genero()));  
386 telefono.setText(String.valueOf(alumno1.get\_Telefono()));  
387 correo.setText(alumno1.get\_Correo());   
388 }  
389 }  
390 }   
391   
392 void actualiza(){  
393   
394 //Alumnos obt;  
395 nombrev=nombre.getText();  
396 apellidov=apellido.getText();  
397 edadv=(int)Integer.parseInt(edad.getText());  
398 estaturav=(double)Double.parseDouble(estatura.getText());  
399 pesov=(int)Integer.parseInt(peso.getText());  
400 generov=(String)comboGenero.getSelectedItem();  
401 telefonov=telefono.getText();   
402 String correov=correo.getText();  
403   
404 try{  
405 nombrev=nombre.getText();  
406 apellidov=apellido.getText();  
407 edadv=(int)Integer.parseInt(edad.getText());  
408 estaturav=(double)Double.parseDouble(estatura.getText());  
409 pesov=(int)Integer.parseInt(peso.getText());  
410 generov=(String)comboGenero.getSelectedItem();  
411 telefonov=telefono.getText();   
412 correov=correo.getText();  
413   
414 System.out.println(generov);  
415   
416 }catch(Exception e){   
417 System.out.print(e);  
418 }  
419   
420 Alumnos ob=new Alumnos(idv,nombrev,apellidov,edadv,estaturav,pesov,generov,telefonov,correov);  
421 listaA.set(idv-1,ob);  
422 ES.escribe("Datos del objeto actualizados exitosamente!!");  
423 nombre.setText("");  
424 apellido.setText("");  
425 edad.setText("");  
426 estatura.setText("");  
427 peso.setText("");  
428 comboGenero.setSelectedIndex(0);  
429 telefono.setText("");  
430 correo.setText("");   
431 }  
432   
433 void ordenaNombres(){  
434 Collections.sort(listaA,  
435 new Comparator<Alumnos>() {  
436 @Override  
437 public int compare(Alumnos p1, Alumnos p2) {  
438 return new String(p1.get\_Nombre()).compareTo(new String(p2.get\_Nombre()));  
439 }  
440 });  
441 }  
442   
443 void ordenaEdades(){  
444 Collections.sort(listaA,  
445 new Comparator<Alumnos>() {  
446 @Override  
447 public int compare(Alumnos p1, Alumnos p2) {  
448 return new Integer(p1.get\_Edad()).compareTo(new Integer(p2.get\_Edad()));  
449 }  
450 });   
451 }  
452   
453   
454   
455 void buscaNombre()  
456 {  
457 nombrev = JOptionPane.showInputDialog(null, "¿A quien buscas? (Ingresa nombre)");  
458 for (int j=0; j<listaA.size(); j++){  
459 alumno1=(Alumnos)listaA.get(j);  
460 if ((alumno1.get\_Nombre()).equals(nombrev)){  
461 idv=alumno1.get\_Id();  
462 nombrev=alumno1.get\_Nombre();   
463 apellidov=alumno1.get\_Apellido();  
464 edadv=alumno1.get\_Edad();  
465 estaturav=alumno1.get\_Estatura();  
466 pesov=alumno1.get\_Peso();  
467 generov=alumno1.get\_Genero();  
468 telefonov=alumno1.get\_Telefono();  
469 correov=alumno1.get\_Correo();  
470   
471 String cad="Catalogo \n\n"+"ID: "+idv  
472 +"\nNombre: "+nombrev  
473 +"\nApellido: "+apellidov  
474 +"\nEdad: "+edadv  
475 +"\nEstatura: "+estaturav  
476 +"\nPeso: "+pesov  
477 +"\nGenero: "+generov  
478 +"\nTelefono: "+telefonov  
479 +"\nCorreo: "+correov  
480 ;  
481 ES.escribe(cad);  
482 }  
483 }  
484 }  
485   
486 void buscaTelefono(){  
487 telefonov = JOptionPane.showInputDialog(null, "¿A quien buscas? (Ingresa telefono)");  
488 for (int j=0; j<listaA.size(); j++){  
489 alumno1=(Alumnos)listaA.get(j);  
490 if ((alumno1.get\_Telefono()).equals(telefonov)){  
491 idv=alumno1.get\_Id();  
492 nombrev=alumno1.get\_Nombre();   
493 apellidov=alumno1.get\_Apellido();  
494 edadv=alumno1.get\_Edad();  
495 estaturav=alumno1.get\_Estatura();  
496 pesov=alumno1.get\_Peso();  
497 generov=alumno1.get\_Genero();  
498 telefonov=alumno1.get\_Telefono();  
499 correov=alumno1.get\_Correo();  
500   
501 String cad="Catalogo \n\n"+"ID: "+idv  
502 +"\nNombre: "+nombrev  
503 +"\nApellido: "+apellidov  
504 +"\nEdad: "+edadv  
505 +"\nEstatura: "+estaturav  
506 +"\nPeso: "+pesov  
507 +"\nGenero: "+generov  
508 +"\nTelefono: "+telefonov  
509 +"\nCorreo: "+correov  
510 ;  
511 ES.escribe(cad);  
512 }   
513 }  
514 }  
515   
516 void leeDatos() throws JAXBException, IOException{  
517   
518 JAXBContext context = JAXBContext.newInstance(Escuela.class);  
519 Unmarshaller um = context.createUnmarshaller();  
520 Escuela escuela = (Escuela) um  
521 .unmarshal(new FileReader("xalumnos.xml"));  
522   
523 for (int i = 0; i < escuela.getAlumnos().toArray().length; i++){   
524 //idv=escuela.getAlumnos().get(i).get\_Id();  
525 nombrev=escuela.getAlumnos().get(i).get\_Nombre();  
526 apellidov=escuela.getAlumnos().get(i).get\_Apellido();  
527 edadv=escuela.getAlumnos().get(i).get\_Edad();  
528 estaturav=escuela.getAlumnos().get(i).get\_Estatura();  
529 pesov=escuela.getAlumnos().get(i).get\_Peso();  
530 generov=escuela.getAlumnos().get(i).get\_Genero();  
531 telefonov=escuela.getAlumnos().get(i).get\_Telefono();  
532 correov=escuela.getAlumnos().get(i).get\_Correo();  
533   
534 alumno1=new Alumnos(idv,nombrev,apellidov,edadv,estaturav,pesov,generov,telefonov,correov);  
535 listaA.add(alumno1);  
536 max++;  
537 idv++;  
538 }  
539 }  
540   
541 void guardaDatos() throws JAXBException, IOException{  
542   
543 ArrayList<Alumnos> alumnos = new ArrayList<Alumnos>();  
544   
545 Escuela escuela2 = new Escuela();  
546 escuela2.setClave("ITA2016");  
547 escuela2.setNombre("Instituto Tecnologico de Apizaco");  
548   
549 for (int l=0; l<listaA.size(); l++){  
550 alumno1=(Alumnos)listaA.get(l);  
551 idv=alumno1.get\_Id();  
552 nombrev=alumno1.get\_Nombre();  
553 apellidov=alumno1.get\_Apellido();  
554 edadv=alumno1.get\_Edad();  
555 estaturav=alumno1.get\_Estatura();  
556 pesov=alumno1.get\_Peso();  
557 generov=alumno1.get\_Genero();  
558 telefonov=alumno1.get\_Telefono();  
559 correov=alumno1.get\_Correo();  
560   
561 alumnos.add(alumno1);  
562 idv++;  
563   
564 escuela2.setAlumnos(alumnos);  
565   
566 JAXBContext context = JAXBContext.newInstance(Escuela.class);  
567 Marshaller m = context.createMarshaller();  
568 m.setProperty(Marshaller.JAXB\_FORMATTED\_OUTPUT, Boolean.TRUE);  
569 m.marshal(escuela2, System.out);  
570   
571 Writer w = null;  
572 try {  
573 w = new FileWriter("xalumnos.xml");  
574 m.marshal(escuela2, w);  
575 }  
576 finally{  
577 try {  
578 w.close();  
579 } catch (Exception e) {  
580 }  
581 }  
582 }  
583   
584   
585 }  
586   
587 void lista(){  
588 Alumnos ov;  
589 String cad="Agenda \n";  
590   
591 for (int j=0; j<listaA.size(); j++){  
592 ov=(Alumnos)listaA.get(j);  
593 cad+="\n"+ov.despliega();  
594 }  
595 ES.escribe(cad);  
596 }  
597 }

**CAPTURAS DE FUNCIONAMIENTO**

Si se consulta la agenda con el botón “Listar” al inicio de la aplicación no aparecerá ningún contacto. Debe primero presionarse el botón “Leer Archivo” para cargar los contactos almacenados en el XML.



Una vez leído el xml, los contactos son agregados a lista de objetos.



Podemos ver que nuestro XML se guarda de acuerdo a los cambios cada que presionamos el botón “Guardar Archivo”, este es un ejemplo de formato XML obtenido.

