

```
1: package com.delta.generics;
2:
3: import android.graphics.drawable.Drawable;
4:
5: /**
6:  * Created by spawrks on 11/15/13.
7:  */
8: public interface Adoptable {
9:
10:     public int getRating();
11:     public String getNumber();
12:     public boolean isAvailable = false;
13:
14: }
```

```
1: package com.delta.generics;
2:
3: import android.app.Activity;
4: import android.widget.ImageView;
5: import android.widget.RatingBar;
6: import android.widget.TextView;
7:
8: /**
9:  * Created by learnovate on 2/26/14.
10: */
11: public class AdoptAdapter<T extends Animal & Adoptable> {
12:
13:     private Activity activity;
14:     private TextView name;
15:     private TextView description;
16:     private RatingBar ratingBar;
17:     private ImageView imageView;
18:
19:     //T stands for "type"
20:     private T t;
21:
22:     public AdoptAdapter(Activity aActivity, TextView aName, TextView aDescription, RatingBar aBar, ImageView anImageView){
23:         this.activity = aActivity;
24:         this.name = aName;
25:         this.description = aDescription;
26:         this.ratingBar = aBar;
27:         this.imageView = anImageView;
28:     }
29:
30:     public void set(T t){
31:         this.t = t;
32:         updateView();
33:     }
34:
35:     public T get(){
36:         return t;
37:     }
38:
39:     private void updateView(){
40:         int resID = t.getImageResourceId(activity);
41:         imageView.setImageResource(resID);
42:         name.setText(t.getName());
43:         description.setText(t.getDescription());
44:         ratingBar.setNumStars(5);
45:         ratingBar.setRating(t.getRating());
46:     }
47:
48:
```

```
49: }
```

```
1: package com.delta.generics;
2:
3: import java.util.ArrayList;
4: import java.util.List;
5:
6: /**
7:  * Created by learnovate on 2/25/14.
8:  */
9: public class AdoptData {
10:     public static List<Dog> mDogList = new ArrayList<Dog>(){
11:         {
12:             add(new Dog("Weston",
13:                 "Weston was a big rockstar in the 90's, but has since settled down and lived a quite life of digging holes and chewing shoes.",
14:                 "david",
15:                 "+1-202-555-0114"));
16:             add(new Dog("Gizmo",
17:                 "So adorable that this dog won the 2014 award for \"Most Likely To Break Your Heart And Pee A Little While Doing It\"",
18:                 "gizmo",
19:                 "+1-202-555-0189"));
20:             add(new Dog("MogWai",
21:                 "This dog basically is an artist at heart, he knows how to paint with his tail, but often just craps on the floor",
22:                 "mogwai",
23:                 "+1-202-555-0190"));
24:             add(new Dog("Pepper",
25:                 "She is a little skeptical of cats, Rubik's cubes, and people using cameras",
26:                 "pepper",
27:                 "+1-202-555-0109"));
28:             add(new Dog("Shadow",
29:                 "This dog is incredible at cuddling up in a blanket. So good that he has been mistaken for a burrito numerous times.",
30:                 "shadow",
31:                 "+1-202-555-0141"));
32:             add(new Dog("Vegas",
33:                 "Not much is know about this dog, except that he stole a pizza in 1997",
34:                 "vegas",
35:                 "+1-202-555-0141"));
36:         }
37:     };
38:
39:     public static List<Cat> mCatList = new ArrayList<Cat>(){
40:         {
41:             add(new Cat("Barry & Levon","Barry and Levon are quite a pair, they love pudding!","barrylevon",20));
42:             add(new Cat("Flash Gordon","We think this cat is sleeping, but its also possible he's stuck. ","flashgordon",12));
43:             add(new Cat("Kif","This cat is very nice and very judgemental. (note: box not included)","kif",8));
```

```
44:         add(new Cat("Mochi","This cat needs a very special diet of only the finest minced tuna and caviar.","mochi",11)
);
45:         add(new Cat("Kitty","A great mouser, she has a habit of bringing you presents year-round","kitty",10));
46:         add(new Cat("Mufassa","Not recommended for tame owners. Not housetrained. Don't tell him I said that.","mufassa
",2));
47:     }
48: }
49: }
```

```
1: package com.delta.generics;
2:
3: import android.app.Activity;
4: import android.content.Context;
5:
6: /**
7:  * Created by learnovate on 2/24/14.
8:  */
9: public interface Animal {
10:
11:     public String getName();
12:     public String getImageName();
13:     public String getDescription();
14:     public int getImageResourceId(Activity a);
15:
16: }
```

```
1: package com.delta.generics;
2:
3: import android.app.Activity;
4: import android.util.Log;
5:
6: import java.lang.reflect.Field;
7:
8: /**
9:  * Created by learnovate on 2/25/14.
10: */
11: public class Cat implements Animal, Adoptable {
12:
13:
14:     public static final int MAX_SHELTER_DAYS = 30;
15:
16:     private String whatYouCallYourCat = "";
17:     private String shelterDescription = "";
18:     private String catPic = "";
19:
20:     private int daysAtShelter = 1;
21:     private String phoneNumber = "";
22:
23:     public Cat(String aName, String aDescription, String anImageName, int aDaysAtShelter) {
24:         this.whatYouCallYourCat = aName;
25:         this.shelterDescription = aDescription;
26:         this.catPic = anImageName;
27:         this.daysAtShelter = aDaysAtShelter;
28:     }
29:
30:     @Override
31:     public int getRating() {
32:         int rating = 5 * daysAtShelter / MAX_SHELTER_DAYS;
33:         return rating;
34:     }
35:
36:     @Override
37:     public String getNumber() {
38:         //All cat adoptions go through one number
39:         return "+1-202-555-0171";
40:     }
41:
42:     @Override
43:     public String getName() {
44:         return whatYouCallYourCat;
45:     }
46:
47:     @Override
48:     public String getImageName() {
```

```
49:         return catPic;
50:     }
51:
52:     @Override
53:     public String getDescription() {
54:         return shelterDescription;
55:     }
56:
57:     @Override
58:     public int getImageResourceId(Activity a) {
59:
60:         int drawableId = 0;
61:         try {
62:             Class res = R.drawable.class;
63:             Field field = res.getField(getImageName());
64:             drawableId= field.getInt(null);
65:         }
66:         catch (Exception e) {
67:             Log.e("MyTag", "Failure to get drawable id.", e);
68:         }
69:         return drawableId;
70:     }
71: }
```



```
1: package com.delta.generics;
2:
3: import android.app.Activity;
4: import android.widget.ImageView;
5: import android.widget.RatingBar;
6: import android.widget.TextView;
7:
8: /**
9:  * Created by learnovate on 2/25/14.
10: */
11: public class CatAdapter {
12:
13:     private Activity activity;
14:     private TextView name;
15:     private TextView description;
16:     private RatingBar ratingBar;
17:     private ImageView imageView;
18:
19:     private Cat t;
20:
21:     public CatAdapter(Activity aActivity, TextView aName, TextView aDescription, RatingBar aBar, ImageView aImageView){
22:         this.activity = aActivity;
23:         this.name = aName;
24:         this.description = aDescription;
25:         this.ratingBar = aBar;
26:         this.imageView = aImageView;
27:     }
28:
29:     public void set(Cat t) {
30:         this.t = t;
31:         updateView();
32:     }
33:
34:     public Cat get() {
35:         return t;
36:     }
37:
38:     private void updateView(){
39:         int resID = t.getImageResourceId(activity);
40:         imageView.setImageResource(resID);
41:         name.setText(t.getName());
42:         description.setText(t.getDescription());
43:         ratingBar.setNumStars(5);
44:         ratingBar.setRating(t.getRating());
45:     }
46: }
```

```
1: package com.delta.generics;
2:
3: import android.app.Activity;
4: import android.util.Log;
5:
6: /**
7:  * Created by learnovate on 2/25/14.
8:  */
9:
10: public class Dog implements Adoptable,Animal{
11:
12:     private String dogName = "";
13:     private String ownersDescription = "";
14:     private String imageName = "";
15:     private int Rating;
16:     private String phoneNumber = "";
17:
18:     public Dog(String aName, String aDesc, String anImageName, String aPhoneNumber){
19:         dogName = aName;
20:         ownersDescription = aDesc;
21:         imageName = anImageName;
22:         phoneNumber = aPhoneNumber;
23:     }
24:
25:     @Override
26:     public int getRating() {
27:         return 9; //all dogs are awesome
28:     }
29:
30:     @Override
31:     public String getNumber() {
32:         return phoneNumber;
33:     }
34:
35:     @Override
36:     public String getDescription() {
37:         return ownersDescription;
38:     }
39:
40:     @Override
41:     public String getName() {
42:         return dogName;
43:     }
44:
45:     @Override
46:     public String getImageName() {
47:         return imageName;
48:     }
```

```
49:
50:     @Override
51:     public int getImageResourceId(Activity anActivity) {
52:         int resID = anActivity.getResources().getIdentifier(getImageName(), "drawable", anActivity.getPackageName());
53:         return resID;
54:     }
55:
56:     public int getNumberOfBallsICanCatch(){
57:         return 5;
58:     }
59:
60:     public void beTheBestDogEver(){
61:         for(int i = 0; i < 1000;i++){
62:             Log.e("DOG", getName() + "says Woof!");
63:         }
64:     }
65:
66:
67: }
```

```
1: package com.delta.generics;
2:
3: import android.app.Activity;
4: import android.net.wifi.p2p.WifiP2pDevice;
5: import android.net.wifi.p2p.WifiP2pManager;
6: import android.os.Bundle;
7: import android.service.dreams.DreamService;
8: import android.util.StringBuilderPrinter;
9: import android.view.Menu;
10: import android.view.MenuItem;
11: import android.view.View;
12: import android.widget.AdapterView;
13: import android.widget.Button;
14: import android.widget.ImageView;
15: import android.widget.RatingBar;
16: import android.widget.TextView;
17:
18: import java.util.ArrayList;
19: import java.util.List;
20:
21: public class GenericsActivity extends Activity {
22:
23:     public TextView nameTextView = null;
24:     public TextView descriptionTextView = null;
25:     public RatingBar ratingView = null;
26:     public ImageView portraitView = null;
27:     public Button nextButton = null;
28:
29:     private int currentSelection = 0;
30:
31:     CatAdapter catAdapter;
32:     AdoptAdapter<Cat> catAdoptAdapter;
33:     AdoptAdapter<Dog> dogAdoptAdapter;
34:
35:     @Override
36:     protected void onCreate(Bundle savedInstanceState) {
37:         super.onCreate(savedInstanceState);
38:         setContentView(R.layout.activity_generics);
39:
40:         nameTextView = (TextView) findViewById(R.id.nameTextView);
41:         descriptionTextView = (TextView) findViewById(R.id.descriptionTextView);
42:         ratingView = (RatingBar) findViewById(R.id.ratingView);
43:         portraitView = (ImageView) findViewById(R.id.portraitView);
44:         nextButton = (Button) findViewById(R.id.nextButton);
45:         nextButton.setOnClickListener(new View.OnClickListener() {
46:             @Override
47:             public void onClick(View v) {
48:                 showNext();
```

```
49:         }
50:     });
51:
52:     //         catAdapter = new CatAdapter(this,nameTextView,descriptionTextView,ratingView,portraitView);
53:     //         catAdapter.set(AdoptData.mCatList.get(0));
54:
55:     //         catAdoptAdapter = new AdoptAdapter<Cat>(this,nameTextView,descriptionTextView,ratingView,portraitView);
56:     dogAdoptAdapter = new AdoptAdapter<Dog>(this,nameTextView,descriptionTextView,ratingView,portraitView);
57:     //
58:     //         catAdoptAdapter.set(AdoptData.mCatList.get(0));
59:     dogAdoptAdapter.set(AdoptData.mDogList.get(0));
60:
61:
62:
63:
64: }
65:
66: public void showNext(){
67:     int random = currentSelection;
68:     while(random == currentSelection){
69:         //avoid same selection twice.
70:         //         random = (int )(Math.random() * AdoptData.mCatList.size());
71:         random = (int) (Math.random() * AdoptData.mDogList.size());
72:     }
73:     currentSelection = random;
74:     //         Cat c = AdoptData.mCatList.get(random);
75:     Dog d = AdoptData.mDogList.get(random);
76:     //         catAdapter.set(c);
77:     //         catAdoptAdapter.set(c);
78:     dogAdoptAdapter.set(d);
79: }
80:
81:
82: @Override
83: public boolean onCreateOptionsMenu(Menu menu) {
84:
85:     // Inflate the menu; this adds items to the action bar if it is present.
86:     getMenuInflater().inflate(R.menu.generics, menu);
87:     return true;
88: }
89:
90: @Override
91: public boolean onOptionsItemSelected(MenuItem item) {
92:     // Handle action bar item clicks here. The action bar will
93:     // automatically handle clicks on the Home/Up button, so long
94:     // as you specify a parent activity in AndroidManifest.xml.
95:     int id = item.getItemId();
96:     if (id == R.id.action_settings) {
```

```
97:         return true;
98:     }
99:     return super.onOptionsItemSelected(item);
100: }
101:
102:
103:
104: }
```

1	generics/src/main/java/com/delta/generics/Adoptable.java	1	pages	14	lines	14/02/25	14:02:53
2	generics/src/main/java/com/delta/generics/AdoptAdapter.java	2	pages	49	lines	14/09/18	16:13:21
3	generics/src/main/java/com/delta/generics/AdoptData.java	2	pages	49	lines	14/09/18	16:13:21
4	generics/src/main/java/com/delta/generics/Animal.java	1	pages	16	lines	14/02/25	14:02:53
5	generics/src/main/java/com/delta/generics/Cat.java	2	pages	71	lines	14/02/25	19:50:40
6	generics/src/main/java/com/delta/generics/CatAdapter.java	1	pages	46	lines	14/04/02	14:18:23
7	generics/src/main/java/com/delta/generics/Dog.java	2	pages	67	lines	14/02/25	14:13:07
8	generics/src/main/java/com/delta/generics/GenericsActivity.java	3	pages	104	lines	14/09/18	16:13:21