ФГБОУ ВО «НИУ «МЭИ»

Кафедра ВМСС

Лабораторная работа №4

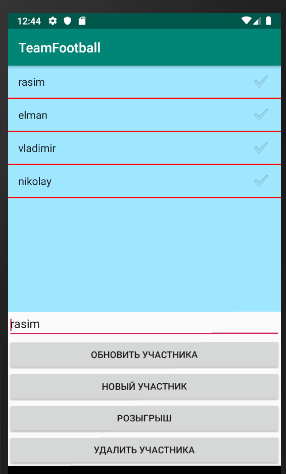
По курсу «Разработка мобильных приложений»

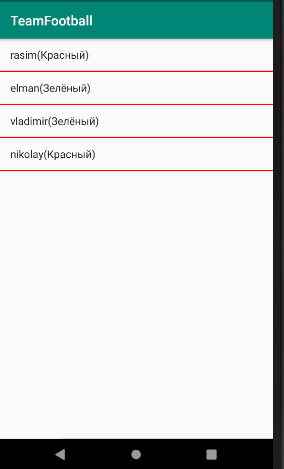
Разработка программы «Сформируй команду»

|  |  |
| --- | --- |
| Выполнил: | Самошкин Д.Ю. |
| Группа: | А-08м-19 |
| Вариант: | - |

Москва, 2020 г.

**Задание**

Написать программу, которая формирует состав участников двух команд «красных» и «зеленых» случайным образом. Имена игроков вводятся в программу и сохраняется. Пользователь отмечает присутствующих, нажимает кнопку сформировать состав. Программа выводит список красных и зеленых игроков.



После того как пользователь отмечает присутствующих игроков, нажимает на кнопку розыгрыш, появляется результаты розыгрыша:

**Выполнение работы**

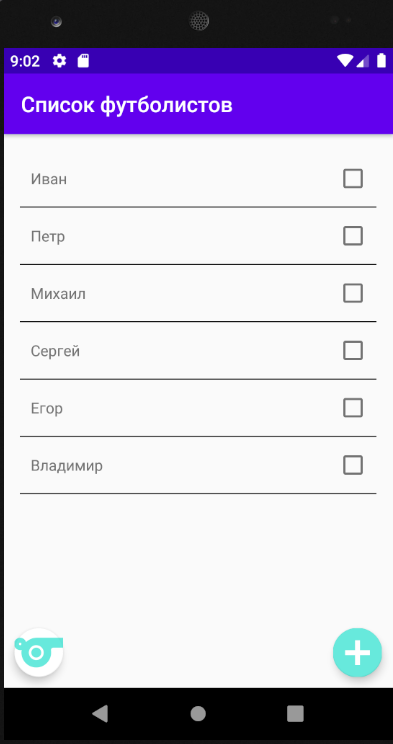
Лабораторная работа была сделана в 2 активностях, с одним классом, одним интерфейсом и одним адаптером данных.

|  |  |
| --- | --- |
| Название | Назначение |
| MainActivity | Главная активность, где пользователь может добавлять, изменять, удалять и отмечать футболистов |
| TeamsActivity | Вторая активность, в которой отображаются игроки, распределенный по командам случайным образом |
| Footballer | Класс, содержащий данные футболиста ( имя, команда, отметка о присутствии) |
| DataAdapter | Адаптер данных для работы с RecyclerView и отображения данных о футболистах в двух вышеупомянутых активностях |
| Constants | Интерфейс с константами, используемыми в приложении |

**Тестирование работы приложения**

|  |  |
| --- | --- |
| **Добавление игрока** | **Добавленный игрок** |
|  |  |
| **Изменение игрока** | **Удаление игрока** |
|  |  |
| **Вид меню «изменения и удаления»** | **Розыгрыш (Не отмечен ни один игрок)** |
|  |  |
| **Розыгрыш (Четное число игроков)** | **Розыгрыш (Нечетное число игроков)** |
|  |  |

**Активность MainActivty**



Активность состоит из списка игроков, кнопки розыгрыша (распределения игроков по командам случайным образом) и кнопки добавления нового игрока.

При зажатии (longItemClick) на одном из игроков, появляется меню, в котором можно удалить или изменить имя игрока. В последнем случае возникает диалоговое окно, предлагающее изменить имя игрока.

При нажатии на добавление игрока (правый нижний угол), появляется диалоговое окно, предлагающее ввести имя нового игрока.

При нажатии на розыгрыш (расположен в левом нижнем углу экрана) происходит переход в другую активность с передачей в нее всех отмеченных игроков.

Все игроки из списка этой активности сохраняются в памяти смартфона и доступны при следующем запуске приложения.

Код activity\_main.xml

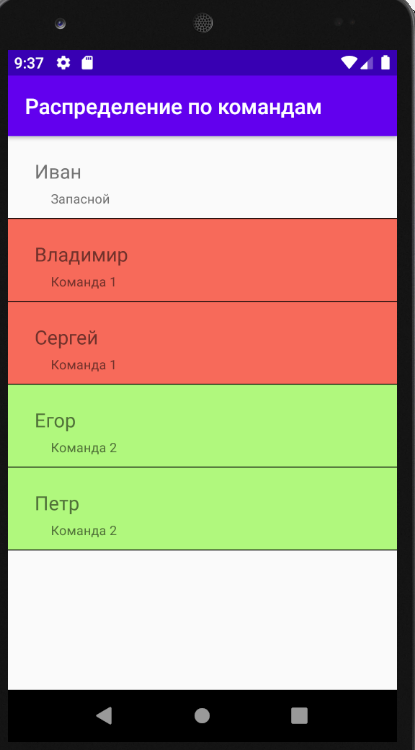
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:padding="15dp"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:id="@+id/recyclerViewFootballers" />  
  
 <com.google.android.material.floatingactionbutton.FloatingActionButton  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnAdd"  
 android:src="@drawable/ic\_btn\_add"  
 app:backgroundTint="@android:color/white"  
 android:tint="@color/colorAccent"  
 app:fabCustomSize="45dp"  
 app:maxImageSize="55dp"  
 android:elevation="10dp"  
 android:clickable="true"  
 android:layout\_alignParentBottom="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_margin="10dp"  
 android:onClick="onBtnAdd"  
 />  
  
 <com.google.android.material.floatingactionbutton.FloatingActionButton  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnMakeRandomTeams"  
 android:src="@drawable/ic\_btn\_make\_random\_team"  
 app:backgroundTint="@android:color/white"  
 android:tint="@color/colorAccent"  
 app:fabCustomSize="45dp"  
  
 app:maxImageSize="55dp"  
 android:elevation="10dp"  
 android:clickable="true"  
  
 android:layout\_alignParentBottom="true"  
 android:layout\_alignParentLeft="true"  
  
 android:layout\_margin="10dp"  
  
 android:onClick="onBtnMakeTeams"  
 />  
  
</RelativeLayout>

Код элемента списка footballer\_item.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:padding="5dp"  
 android:id="@+id/parentLayout"  
 xmlns:app="http://schemas.android.com/apk/res-auto">  
  
 <TextView  
 android:id="@+id/tvName"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="5dp"  
 android:layout\_marginStart="4dp"  
 android:layout\_weight="9"  
 android:hint="@string/hintName"  
 android:inputType="textPersonName"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <CheckBox  
 android:layout\_margin="5dp"  
 android:id="@+id/checkBox"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:checked="false"  
 android:visibility="visible"/>  
  
</LinearLayout>

MainActivity.java

package com.example.lab4;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.DividerItemDecoration;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import android.content.Context;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.content.SharedPreferences;  
import android.os.Bundle;  
import android.text.TextUtils;  
import android.util.Log;  
import android.view.LayoutInflater;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import java.util.ArrayList;  
  
public class MainActivity extends AppCompatActivity {  
  
 private static final String *TAG* = "MainActivity";  
   
 private ArrayList<Footballer>footballers=new ArrayList<>();  
 private RecyclerView recyclerView;  
 private DataAdapter adapter;  
   
 private SharedPreferences sharedPreferences;  
   
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 sharedPreferences = getApplicationContext().getSharedPreferences("com.example.lab4", Context.*MODE\_PRIVATE*);  
 getSupportActionBar().setTitle("Список футболистов");  
 initRecyclerView();  
 readFromSharedPreferences();  
  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 clearSharedPreferences();  
 writeToSharedPreferences();  
 }  
  
 private void readFromSharedPreferences() {  
 int i =0;  
 while (sharedPreferences.contains("footballer"+i)){  
 adapter.add(new Footballer(sharedPreferences.getString("footballer"+i,"empty")));  
 i++;  
 }  
 }  
  
 private void clearSharedPreferences(){  
 int i = 0;  
 while (sharedPreferences.contains("footballer"+i)){  
 sharedPreferences.edit().remove("footballer"+i).apply();  
 i++;  
 }  
 }  
  
 private void writeToSharedPreferences(){  
 footballers=adapter.getFootballers();  
  
 for (int i=0;i<footballers.size();i++){  
 sharedPreferences.edit().putString("footballer"+i,footballers.get(i).getName()).apply();  
 }  
 }  
  
  
 private void initRecyclerView(){  
 Log.*i*(*TAG*, "initRecyclerView: init");  
 recyclerView= findViewById(R.id.*recyclerViewFootballers*);  
 adapter = new DataAdapter(this,footballers,Constants.*VIEW\_TYPE\_1*);  
 recyclerView.setAdapter(adapter);  
 recyclerView.setLayoutManager(new LinearLayoutManager(this));  
  
 DividerItemDecoration dividerItemDecoration = new DividerItemDecoration(this, RecyclerView.*VERTICAL*);  
 dividerItemDecoration.setDrawable(getResources().getDrawable(R.drawable.*divider\_drawable*));  
 recyclerView.addItemDecoration(dividerItemDecoration);  
  
 }  
  
 public void onBtnAdd(View view) {  
 AlertDialog.Builder dialog = new AlertDialog.Builder(this);  
 dialog.setTitle("Введите имя");  
 //Объект для получение шаблона окна  
 LayoutInflater inflater = LayoutInflater.*from*(this);  
 //Получаем шаблон окна register\_window в переменную registerWindow  
 View dialogWindow = inflater.inflate(R.layout.*dialog\_window*,null);  
 //Устанавливаем шаблон для всплывающего окна  
 dialog.setView(dialogWindow);  
 final EditText editText = dialogWindow.findViewById(R.id.*etTextField*);  
  
 dialog.setNegativeButton("Отмена", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int i) {  
 dialogInterface.dismiss();  
 }  
 });  
 dialog.setPositiveButton("Подтвердить", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int i) {  
 if (TextUtils.*isEmpty*(editText.getText().toString())){  
 Toast.*makeText*(MainActivity.this,  
 "Пустое поле ввода",Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
 adapter.add(new Footballer(editText.getText().toString(),true));  
  
 }  
 });  
 dialog.show();  
  
 }  
  
 @Override  
 public boolean onContextItemSelected(@NonNull MenuItem item) {  
 super.onContextItemSelected(item);  
 final MenuItem copyItem = item;  
 switch (item.getItemId())  
 {  
  
 //Выбор удаления  
 case R.id.*action\_delete*:  
 adapter.removeAt(item.getGroupId());  
 Toast.*makeText*(this,"Игрок удален",Toast.*LENGTH\_SHORT*).show();  
 return true;  
  
 //Выбор изменения  
 case R.id.*action\_change*:  
  
 AlertDialog.Builder dialog = new AlertDialog.Builder(this);  
 dialog.setTitle("Введите имя");  
 //Объект для получение шаблона окна  
 LayoutInflater inflater = LayoutInflater.*from*(this);  
 //Получаем шаблон окна register\_window в переменную registerWindow  
 View dialogWindow = inflater.inflate(R.layout.*dialog\_window*,null);  
 //Устанавливаем шаблон для всплывающего окна  
 dialog.setView(dialogWindow);  
 final EditText editText = dialogWindow.findViewById(R.id.*etTextField*);  
  
 editText.setSelectAllOnFocus(true);  
 editText.setText(footballers.get(copyItem.getGroupId()).getName());  
  
 dialog.setNegativeButton("Отмена", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int i) {  
 dialogInterface.dismiss();  
 }  
 });  
 dialog.setPositiveButton("Подтвердить", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int i) {  
 if (TextUtils.*isEmpty*(editText.getText().toString())){  
 Toast.*makeText*(MainActivity.this,  
 "Пустое поле ввода",Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
  
 adapter.updateAt(copyItem.getGroupId(),  
 new Footballer(editText.getText().toString(),  
 footballers.get(copyItem.getGroupId()).check()));  
 Toast.*makeText*(MainActivity.this,  
 "Игрок изменен",Toast.*LENGTH\_SHORT*).show();  
  
 }  
 });  
 dialog.show();  
  
 return true;  
 default:  
 return false;  
 }  
  
 }  
  
 public void onBtnMakeTeams(View view) {  
 // footballers=adapter.getFootballers();  
 Intent intent = new Intent(MainActivity.this,TeamsActivity.class);  
 int countPresent=0;  
 //Перебираем футболистов и отправляем только тех, кто присутствует  
 for (int i=0;i<footballers.size();i++){  
 Footballer footballer = footballers.get(i);  
 if(footballer.check()){  
 intent.putExtra ("footballer"+countPresent,footballer.getName());  
 countPresent++;  
 }  
 }  
 Log.*i*(*TAG*, "onBtnMakeTeams: footballers size = "+footballers.size());  
 Log.*i*(*TAG*, "onBtnMakeTeams: countPresent "+String.*valueOf*(countPresent));  
 intent.putExtra("countPresent",countPresent);  
 startActivity(intent);  
 }  
}

**Активность TeamsActivity**

Активность состоит из игроков, которые были отмечены в прошлой активности. В результате розыгрыша (распределения игроков по командам случайным образом) игроки распределены поровну по 2ум командам. В случае нечетного числа игроков, один из игроков попадает в запас. Элементы списка отсортированы по командам, а затем по именам в алфавитном порядке.

Код activity\_teams.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
  
 tools:context=".TeamsActivity">  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/recyclerViewFootballersWithTeams"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 />  
  
</RelativeLayout>

Код элемента списка item\_footballer\_in\_team.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:id="@+id/parentLayoutWithTeams"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:padding="10dp"  
 >  
  
  
 <TextView  
 android:id="@+id/textViewName"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="15dp"  
 android:layout\_marginTop="10dp"  
 android:text="TextView"  
 android:textSize="18sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/textViewTeam"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="30dp"  
 android:layout\_marginTop="5dp"  
 android:text="TextView"  
 android:textSize="12sp"  
 app:layout\_constraintTop\_toBottomOf="@+id/textViewName"  
 app:layout\_constraintStart\_toStartOf="parent"  
 />  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

Код TeamsActivity.java

package com.example.lab4;  
  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.DividerItemDecoration;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.Toast;  
  
import java.util.ArrayList;  
import java.util.Collections;  
  
public class TeamsActivity extends AppCompatActivity {  
  
 private static final String *TAG* = "TeamsActivity";  
 private ArrayList<Footballer>footballers= new ArrayList<>();  
 private RecyclerView recyclerView;  
 private DataAdapter adapter;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_teams*);  
 getSupportActionBar().setTitle("Распределение по командам");  
 Intent intent=getIntent();  
 int count = intent.getIntExtra("countPresent",-1);  
 Log.*i*(*TAG*, "onCreate: count = "+count);  
 if (count<0){  
 Toast.*makeText*(TeamsActivity.this,"Ошибка передачи данных",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 if (count == 0){  
 Toast.*makeText*(TeamsActivity.this,"Не был отмечен ни один игрок",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 for (int i =0;i<count;i++){  
 Footballer footballer= new Footballer(intent.getStringExtra("footballer"+i));  
 footballers.add(footballer);  
 }  
 initRecyclerView();  
 distributeFootballers();  
  
   
 }  
  
 private void initRecyclerView() {  
 recyclerView=findViewById(R.id.*recyclerViewFootballersWithTeams*);  
 adapter = new DataAdapter(TeamsActivity.this,footballers,Constants.*VIEW\_TYPE\_2*);  
 recyclerView.setAdapter(adapter);  
 recyclerView.setLayoutManager(new LinearLayoutManager(TeamsActivity.this));  
  
 DividerItemDecoration dividerItemDecoration = new DividerItemDecoration  
 (this, RecyclerView.*VERTICAL*);  
 dividerItemDecoration.setDrawable(getResources().getDrawable(R.drawable.*divider\_drawable*));  
 recyclerView.addItemDecoration(dividerItemDecoration);  
  
 }  
  
 public void distributeFootballers(){  
  
 //Число игроков  
 int fullSize = adapter.getItemCount();  
 //Первая половина игроков  
 int half = fullSize/2;  
 //Нечетность числа игроков  
 int odd = fullSize%2;  
 //Счетчик игроков  
 int count = 0;  
  
 //В первую команду  
 while (count<half)  
 {  
 int rNum = (int) (Math.*random*()\*fullSize);  
 //Если еще не распределили  
 if (! footballers.get(rNum).check()){  
 footballers.get(rNum).setTeam(Constants.*TEAM1*);  
 footballers.get(rNum).setChecked(true);  
 count++;  
 }  
 }  
 //Во вторую команду  
 while (count<fullSize){  
 int rNum = (int) (Math.*random*()\*fullSize);  
 //Если еще не распределили  
 if (! footballers.get(rNum).check()){  
 //Если нечетное число игроков и это последний игрок распределения  
 //Игрок будет в запасе  
 if (odd == 1 && count == fullSize - 1){  
 footballers.get(rNum).setTeam(Constants.*RESERVE*);  
 }  
 else {  
 footballers.get(rNum).setTeam(Constants.*TEAM2*);  
 }  
 footballers.get(rNum).setChecked(true);  
 count++;  
 }  
  
 }  
 Collections.*sort*(footballers,Footballer.*COMPARE\_BY\_TEAM*);  
 adapter.notifyDataSetChanged();  
 }  
}

**Класс Footballer.java**

Класс, который представляет из себя модель футболиста. Состоит из полей имя, команда, отметка. Реализует Parcelable (для интента) и Serializable (для компаратора и сортировки в списке).

Код класса Footballer.java

package com.example.lab4;  
  
import android.content.Intent;  
import android.os.Parcel;  
import android.os.Parcelable;  
import android.text.TextUtils;  
import android.widget.TextView;  
  
import java.io.Serializable;  
import java.util.Comparator;  
  
public class Footballer implements Parcelable, Serializable {  
 String name;  
 String team = Constants.*NO\_TEAM*;  
 boolean isChecked =false;  
  
  
 public Footballer(String name, boolean isChecked) {  
 this.name = name;  
 this.isChecked = isChecked;  
 }  
  
 public Footballer() {  
 }  
  
 public Footballer(String name) {  
 this.name = name;  
 }  
  
  
 protected Footballer(Parcel in) {  
 name = in.readString();  
  
 }  
  
 public static final Creator<Footballer> *CREATOR* = new Creator<Footballer>() {  
 @Override  
 public Footballer createFromParcel(Parcel in) {  
 return new Footballer(in);  
 }  
  
 @Override  
 public Footballer[] newArray(int size) {  
 return new Footballer[size];  
 }  
 };  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public String getTeam() {  
 return team;  
 }  
  
 public void setTeam(String team) {  
 this.team = team;  
 }  
  
 public boolean check() {  
 return isChecked;  
 }  
  
 public void setChecked(boolean checked) {  
 isChecked = checked;  
 }  
  
 @Override  
 public int describeContents() {  
 return 0;  
 }  
  
 @Override  
 public void writeToParcel(Parcel parcel, int i) {  
 parcel.writeString(name);  
 //parcel.writeBoolean(isChecked);  
  
  
 }  
 public static final Comparator<Footballer> *COMPARE\_BY\_TEAM* = new Comparator<Footballer>() {  
 @Override  
 public int compare(Footballer f1, Footballer f2) {  
  
 //Сравниваем и по названию команды и по имени игрока  
 //Значение сравнения по команде имеет больший вес  
 return f1.getTeam().compareTo(f2.getTeam())\*100 + f1.getName().compareTo(f2.getName());  
 }  
 };  
}

**Класс DataAdapter**

Класс адаптера данных для отображения футболистов в списках активностей. Имеет внутри 2 различных класса ViewHolder для типов элементов (для каждой активности свой тип). Содержит в себе методы добавления, удаления и изменения футболиста.

Код класса DataAdapter.java

package com.example.lab4;  
  
import android.content.Context;  
import android.util.Log;  
import android.view.ContextMenu;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.CheckBox;  
import android.widget.CompoundButton;  
import android.widget.LinearLayout;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.constraintlayout.widget.ConstraintLayout;  
import androidx.recyclerview.widget.RecyclerView;  
  
import java.util.ArrayList;  
import java.util.Comparator;  
  
public class DataAdapter extends RecyclerView.Adapter{  
  
 public static final String *TAG* = "DataAdapter";  
 private ArrayList<Footballer>footballers = new ArrayList<>();  
 private int viewType;  
 private Context mContext;  
  
 public DataAdapter(Context mContext, ArrayList<Footballer> footballers, int viewType ) {  
 this.footballers = footballers;  
 this.mContext = mContext;  
 this.viewType= viewType;  
 }  
  
 @Override  
 public int getItemViewType(int position) {  
 return this.viewType;  
 }  
  
 @NonNull  
 @Override  
 public RecyclerView.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
  
 if(this.viewType == Constants.*VIEW\_TYPE\_1*){  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.*footballer\_item*,parent,false);  
 ViewHolder1 holder1 = new ViewHolder1(view);  
 return holder1;  
 }  
 else  
 if(this.viewType == Constants.*VIEW\_TYPE\_2*){  
 View view = LayoutInflater.*from*(parent.getContext())  
 .inflate(R.layout.*item\_footballer\_in\_team*,parent,false);  
 ViewHolder2 holder2 = new ViewHolder2(view);  
 return holder2;  
 }  
 return null;  
 }  
  
 @Override  
 public void onBindViewHolder(@NonNull RecyclerView.ViewHolder holder, int position) {  
  
 final int position\_copy = position;  
 Log.*i*(*TAG*, "onBindViewHolder: called");  
  
 if(this.viewType==Constants.*VIEW\_TYPE\_1*){  
  
 ViewHolder1 holder1= (ViewHolder1)holder;  
 holder1.tvName.setText(footballers.get(position).getName());  
 holder1.cbIsPresent.setChecked(footballers.get(position).check());  
 holder1.parentLayout.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Log.*i*(*TAG*,"onClick"+footballers.get(position\_copy).getName());  
 Toast.*makeText*(mContext,footballers.get(position\_copy).getName(),Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
 holder1.cbIsPresent.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
 @Override  
 public void onCheckedChanged(CompoundButton compoundButton, boolean b) {  
 try {  
 footballers.get(position\_copy).setChecked(b);  
 }  
 catch (Exception e){  
 Log.*i*(*TAG*, "onCheckedChanged: Exception caught. Doesn't exist anymore");  
 return;  
 }  
  
 }  
 });  
 }  
 else  
 if(this.viewType==Constants.*VIEW\_TYPE\_2*){  
  
 ViewHolder2 holder2 = (ViewHolder2) holder;  
 holder2.tvTeam.setText(footballers.get(position).getTeam());  
 holder2.tvName.setText(footballers.get(position).getName());  
 if(footballers.get(position).getTeam().equals(Constants.*TEAM1*)){  
 holder2.parentLayout.setBackgroundResource(R.color.*colorBackgroundTeam1*);  
 holder2.tvName.setBackgroundResource(R.color.*colorBackgroundTeam1*);  
 holder2.tvTeam.setBackgroundResource(R.color.*colorBackgroundTeam1*);  
 }  
 else  
 if(footballers.get(position).getTeam().equals(Constants.*TEAM2*)){  
 holder2.parentLayout.setBackgroundResource(R.color.*colorBackgroundTeam2*);  
 holder2.tvName.setBackgroundResource(R.color.*colorBackgroundTeam2*);  
 holder2.tvTeam.setBackgroundResource(R.color.*colorBackgroundTeam2*);  
 }  
 }  
  
 }  
  
 @Override  
 public int getItemCount() {  
 return footballers.size();  
 }  
  
 public void removeAt (int position){  
 footballers.remove(position);  
 notifyDataSetChanged();  
 }  
  
 public void updateAt(int position, Footballer footballer){  
 footballers.remove(position);  
 footballers.add(position,footballer);  
 notifyDataSetChanged();  
 }  
  
 public void add(Footballer footballer){  
 footballers.add(footballer);  
 notifyDataSetChanged();  
  
 }  
  
 public ArrayList<Footballer> getFootballers() {  
 return footballers;  
 }  
  
  
 class ViewHolder1 extends RecyclerView.ViewHolder implements View.OnCreateContextMenuListener {  
  
 TextView tvName;  
 CheckBox cbIsPresent;  
 LinearLayout parentLayout;  
  
 public ViewHolder1(@NonNull View itemView) {  
 super(itemView);  
 tvName=itemView.findViewById(R.id.*tvName*);  
 cbIsPresent=itemView.findViewById(R.id.*checkBox*);  
 parentLayout=itemView.findViewById(R.id.*parentLayout*);  
 parentLayout.setOnCreateContextMenuListener(this);  
 }  
  
 @Override  
 public void onCreateContextMenu(ContextMenu contextMenu, View v, ContextMenu.ContextMenuInfo contextMenuInfo) {  
  
 contextMenu.setHeaderTitle("Что вы хотите сделать?");  
 contextMenu.add(getAdapterPosition(),R.id.*action\_delete*,0,"Удалить");  
 contextMenu.add(getAdapterPosition(), R.id.*action\_change*,0,"Изменить");  
 }  
 }  
  
 class ViewHolder2 extends RecyclerView.ViewHolder{  
  
 TextView tvName;  
 TextView tvTeam;  
 ConstraintLayout parentLayout;  
  
 public ViewHolder2(@NonNull View itemView) {  
 super(itemView);  
  
 tvName=itemView.findViewById(R.id.*textViewName*);  
 tvTeam=itemView.findViewById(R.id.*textViewTeam*);  
 parentLayout=itemView.findViewById(R.id.*parentLayoutWithTeams*);  
 }  
  
 }  
}

Код интерфейса с константами Constans.java

package com.example.lab4;  
  
public interface Constants {  
 public static final int *VIEW\_TYPE\_1* = 1;  
 public static final int *VIEW\_TYPE\_2* = 2;  
 public static final String *NO\_TEAM*="Без команды";  
 public static final String *TEAM1*="Команда 1";  
 public static final String *TEAM2*="Команда 2";  
 public static final String *RESERVE*="Запасной";  
}