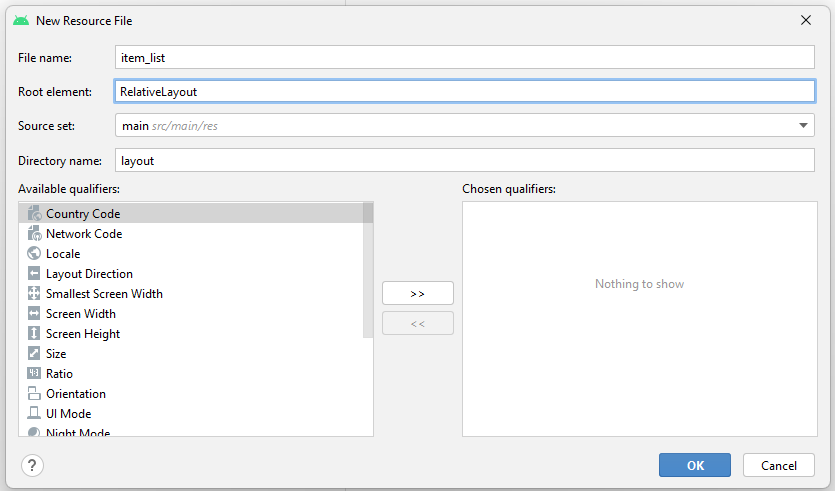
Задание

Пользователю предлагается перечень блюд. Он может выбрать одно или несколько. По окончанию выбора можно запросить счет.

Замечания:

1. Во всех файлах xml указывать orientation. Ошибка не указывается, но и вывода vможет не быть.
2. В файле разметки элемента указывать android:layout\_height="wrap\_content"> иначе будет выводиться только одна строка.
3. Создайте проект.
4. Откройте папку ресурсов (res) и вставьте из папки, прилагаемой к заданию, фотографии блюд в папку drawable.
5. Откройте файл activity\_main, удалите имеющийся там элемент TextView и вставьте из панели элементов вместо него элемент RecyclerView.
6. Для элемента RecyclerView задайте имя android:id="@+id/rv".
7. Откройте папку ресурсов (res), найдите в ней папку layout и щелкните правой кнопкой.
8. В открывшемся меню выберите позицию New, в следующем меню выберите позицию Layout Resource File и щелкните по ней.
9. В появившемся окне наберите имя файла item\_list, в позиции Root element удалите имеющийся там вид разметки и наберите разметку RelativeLayout.

их

1. Откройте файл и вставьте туда элемент ImageView и четыре элемента TextView.
2. Разместите элементы, выровняв их.
3. Установите для элементов имена: img, name, txtm, txtv и txtm.
4. Выполняйте импорт классов везде, где это требуется.
5. Добавьте класс для объектов списка (добавьте конструктор и методы getter and setter)

public class Meal {

int img; *// Изображение блюда*

String name; *// Название*

int price; *// Цена*

int quan; *// Количество*

int total; *// Итог по строке*

}

1. Для добавления в класс указанных позиций щелкните правой кнопкой по строке объявления класса и в появившемся меню выберите позицию Generate. Для добавления конструктора выберите позицию Constructor, для методов доступа позицию getter and setter. В появляющихся окнах выбирайте все перечисленные свойства класса.
2. Добавьте класс адаптера таблицы (MealAdapter)
3. Вставьте наследование

extends RecyclerView.Adapter<MealAdapter.ViewHolder>

MealAdapter – имя адаптера (может быть другим).

1. Вставьте члены класса адаптера (LayoutInflater и список желательно ArrayList, со списком List возможны проблемы)

private final LayoutInflater inflater;

private final ArrayList<Meal> meals;

Meal - имя класса для объектов списка.

1. Опишите конструктор адаптера (вручную, автоматизация приводит к созданию конструктора, с которым вывод списка не получается)

public MealAdapter( Context context, ArrayList<Meal> meals) {

this.inflater = LayoutInflater.*from*(context);

this.meals = meals;

}

Примечание: Частичная автоматизация возможна. Для этого щелкните по любому из членов класса (не по заголовку) и выберите позицию Generate. После этого измените параметры конструктора и строку с inflater, как показано.

1. Щелкните правой кнопкой по заголовку адаптера и в меню Show Context Actions выберите позицию Implements Methods. В адаптер будут добавлены три метода (onCreateViewHolde, onBindViewHolder и getItemCountывать).
2. Щелкните правой кнопкой в заголовке адаптера по лексеме ViewHolder и выберите позицию Create class с именем (ViewHolder). Можно указать похожее имя, например, MyViewHolder. Этот класс выдает в метод класса адаптера (onBindViewHolder) представление (View) текущего элемента списка.
3. Щелкните снова правой кнопкой в строке заголовка класса адаптера (MealAdapter) по ViewHolder и выберите позицию

Make ViewHolder extends android.recyclerview.widget.RecyclerView.ViewHolder

(Система добавит в наследование для класса ViewHolder только RecyclerView.ViewHolder)

1. Заголовок класса ViewHolder подчеркнут красной линией. Щелкните по ней правой кнопкой и выберите позицию Create constructor matching super. Будет создан конструктор в этом классе.
2. В класс ViewHolder добавьте описание элементов, которые будут обрабатываться в этом классе (порядок следования элементов должен совпадать с порядком их объявления в классе Meal)

ImageView imgView;:

TextView name, name, txtm, txtv, txtp;

Эти описания вставляются перед конструктором

1. В класс ViewHolder добавьте параметры для конструктора базового класса (после строки super(itemView). Не забудьте импортировать классы)

imgView = itemView.findViewById(R.id.*imgView*);

nameView= itemView.findViewById(R.id.*nameView*);

txtm = itemView.findViewById(R.id.*txtm*);

txtv = itemView.findViewById(R.id.*txtv*);

txtp = itemView.findViewById(R.id.*txtp*);

1. В метод getItemCount вставьте вместо 0 выражение meals.size();. Здесь meals – имя списка, объявленного как член класса адаптера.
2. В метод onCreateViewHolde вставьте вместо оператора return операторы (item\_list- это описание разметки для одной строки списка, имя задается пользователем при создании элемента Layout):

View view = inflater.inflate(R.layout.item\_list, parent, false);

return new ViewHolder(view);

1. В метод onBindViewHolder вставьте операторы:

Person person = people.get(position);

holder.imgView.setImageResource(people.get(position).getImg());

holder.nameView.setText(person.getName());

holder.statusView.setText(person.getStatus());

Здесь Person – имя класса для объектов списка

people – имя списка

nameView и statusView – имена элементов в файле item\_list (добавленный файл Layout для элементов списка.)

1. Перейдите в окно MainActivity и создайте метод инициализации списка:

private void setInitialData() {

people.add(new Person(R.drawable.*ab\_chekhov*),"Чехов A. П.",

"Писатель"));

people.add(new Person(R.drawable.*ab\_filatov*),"Филатов Л. А",

"Актер и писатель"));

people.add(new Person(R.drawable.*ab\_matroskin\_2*), "Матроскин",

"Кот из Простоквашино"));

people.add(new Person(R.drawable.*ab\_pes*), "Пес",

"Из польского юмора"));

people.add(new Person(R.drawable.*ab\_poruchik*), "Ржевский",

"Герой анекдотов"));

}

Метод вызывается между фигурными скобками (после метода OnCreate), закрывающими класс MainActivity.

1. В классе MainActivity этом объявите глобально список:

ArrayList<Person> people = new ArrayList<Person>();

1. В классе MainActivity после метода OnCreate) объявите элемент Recycler, объявите адаптер и закрепите его за элементом Recycler.

*// начальная инициализация списка*

setInitialData();

RecyclerView recyclerView = findViewById(R.id.*rv*);

recyclerView.setLayoutManager(new LinearLayoutManager(this));

*// создаем адаптер*

MealAdapter adapter = new MealAdapter(this, people);

*// устанавливаем для списка адаптер*

recyclerView.setAdapter(adapter);

1. Проверьте работу адаптера.

Activity\_main.xml

<LinearLayout

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="wrap\_content"

android:orientation="vertical"

tools:context=".MainActivity">

<HorizontalScrollView

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent">

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/rv"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

tools:layout\_editor\_absoluteX="6dp"

tools:layout\_editor\_absoluteY="2dp" />

</HorizontalScrollView>

*<!-- <LinearLayout-->*

*<!-- android:layout\_width="match\_parent"-->*

*<!-- android:layout\_height="wrap\_content"-->*

*<!-- android:orientation="horizontal">-->*

*<!-- <Button-->*

*<!-- android:id="@+id/button"-->*

*<!-- android:layout\_width="wrap\_content"-->*

*<!-- android:layout\_height="wrap\_content"-->*

*<!-- android:text="Button"-->*

*<!-- tools:layout\_editor\_absoluteX="43dp"-->*

*<!-- tools:layout\_editor\_absoluteY="72dp" />-->*

*<!-- <Button-->*

*<!-- android:id="@+id/button2"-->*

*<!-- android:layout\_width="wrap\_content"-->*

*<!-- android:layout\_height="wrap\_content"-->*

*<!-- android:text="Button"-->*

*<!-- android:layout\_marginLeft="40dp"/>-->*

*<!-- </LinearLayout>-->*

</LinearLayout>

tools:listitem="@layout/item\_list"/>

Item\_lisi.xml

<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:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
  
 <ImageView  
 android:id="@+id/imgView"  
 android:layout\_width="294dp"  
 android:layout\_height="250dp"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginStart="56dp"  
 android:layout\_marginTop="9dp"  
 app:srcCompat="@drawable/beefsteak\_1"  
 tools:ignore="ContentDescription" />  
  
 <TextView  
 android:id="@+id/mealView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="260dp"  
 android:text="TextView"  
 tools:ignore="HardcodedText" />  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginStart="55dp"  
 android:layout\_marginTop="296dp"  
 android:text="Цена"  
 tools:ignore="HardcodedText" />  
  
 <TextView  
 android:id="@+id/priceView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignParentEnd="true"  
 android:layout\_marginTop="296dp"  
 android:layout\_marginEnd="66dp"  
 android:text="TextView"  
 tools:ignore="HardcodedText" />  
  
 <TextView  
 android:id="@+id/quanView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="328dp"  
 android:text="TextView"  
 tools:ignore="HardcodedText" />  
</RelativeLayout>

CafeAdapter

import android.content.Context;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import androidx.annotation.NonNull;  
import androidx.recyclerview.widget.RecyclerView;  
  
import java.util.ArrayList;

public class CafeAdapter extends RecyclerView.Adapter<CafeAdapter.ViewHolder>{

private LayoutInflater inflater = null;

private final ArrayList<Meal> food;*// список для обработки*

public CafeAdapter(Context context, ArrayList<Meal> food) {

this.inflater = LayoutInflater.*from*(context);;

this.food = food;

}

@NonNull

@Override

public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

View view = inflater.inflate(R.layout.*item\_list*, parent, false);

return new ViewHolder(view);

}

@Override

public void onBindViewHolder(@NonNull ViewHolder holder, int position) {

Meal meal = food.get(position);

holder.imgView.setImageResource(meal.getImg());

holder.mealView.setText(meal.getName());

holder.textView.setText("Цена");

holder.priceView.setText(String.*valueOf*(meal.getPrice()));

holder.quanView.setText(String.*valueOf*(meal.getQuan()));

}

@Override

public int getItemCount() {

return food.size();

}

public class ViewHolder extends RecyclerView.ViewHolder {

ImageView imgView;

TextView mealView, priceView, quanView, textView;

public ViewHolder(@NonNull View itemView) {

super(itemView);

imgView=itemView.findViewById(R.id.*imgView*);

mealView=itemView.findViewById(R.id.*mealView*);

textView=itemView.findViewById(R.id.*textView*);

priceView=itemView.findViewById(R.id.*priceView*);

quanView=itemView.findViewById(R.id.*quanView*);

}

}

}

Main\_Activity

ArrayList<Meal> food = new ArrayList<Meal>();

RecyclerView recyclerView;

setInitialData();

recyclerView = findViewById(R.id.*rv*);

RecyclerView.LayoutManager layoutManager= new LinearLayoutManager(this);

recyclerView.setLayoutManager(layoutManager);

*// создаем адаптер*

CafeAdapter adapter = new CafeAdapter(this, food);

*// устанавливаем для списка адаптер*

recyclerView.setAdapter(adapter);

public void setInitialData() {

food.add(new Meal(R.drawable.*beefsteak\_1*,"Бифштекс",175, 2));

food.add(new Meal(R.drawable.*chachlyk*,"Шашлык", 273,6));

food.add(new Meal(R.drawable.*duck*, "Утка",122, 1));

food.add(new Meal(R.drawable.*fruit*, "Фрукты",56,1));

food.add(new Meal(R.drawable.*icecream*, "Мороженое",71,6));

food.add(new Meal(R.drawable.*pizza*, "Пицца",122, 1));

food.add(new Meal(R.drawable.*shaurma*, "Шаурма",56,1));

food.add(new Meal(R.drawable.*steak*, "Стейк",71,3));

}

public class Meal {  
 int img;  
 String name;  
 int price;  
 int quan;  
 int total;  
  
 public Meal(int img, String name, int price, int quan) {  
 this.img = img;  
 this.name = name;  
 this.price = price;  
 this.quan = quan;  
 this.total=price\*quan;  
 }  
  
 public int getImg() {  
 return img;  
 }  
  
 public void setImg(int img) {  
 this.img = img;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public int getPrice() {  
 return price;  
 }  
  
 public void setPrice(int price) {  
 this.price = price;  
 }  
  
 public int getQuan() {  
 return quan;  
 }  
  
 public void setQuan(int quan) {  
 this.quan = quan;  
 }  
  
 public int getTotal() {  
 return total;  
 }  
  
 public void setTotal(int total) {  
 this.total = total;  
 }  
}

public List<TwoNumbers> doStuff(List<TwoNumbers> list) {

List<TwoNumbers> returnList = new ArrayList<>();

for (int i = 0; i < list.size(); i++) {

boolean flag = true;

for (int j = i + 1; j < list.size(); j++) {

if (list.get(i).getNum1() == list.get(j).getNum1() && list.get(i).getNum2() == list.get(j).getNum2()) {

flag = false;

}

}

if (flag) {

returnList.add(list.get(i));

}

}

return returnList;

}