Карточки

Приложение со списком карточек автомобилей и их просмотр.

1. Создайте проект Android Studio (имя должно быть не MyApplication).
2. Откройте в папке app файл Manifests и в операторе со свойством label замените имя проекта текстом «Автомобили России».
3. Возьмите у преподавателя и скопируйте в папку darwable 2 папки изображений: автомобилей (папка auto) и их логотипов (папка icon).
4. Откройте файл activity\_main.xml.
5. Замените разметку ConstraintLayout на LinearLayout.
6. Добавьте ориентацию по вертикали
7. Удалите имеющийся там элемент TextView и вставьте элемент RecyclerView, указав его имя.

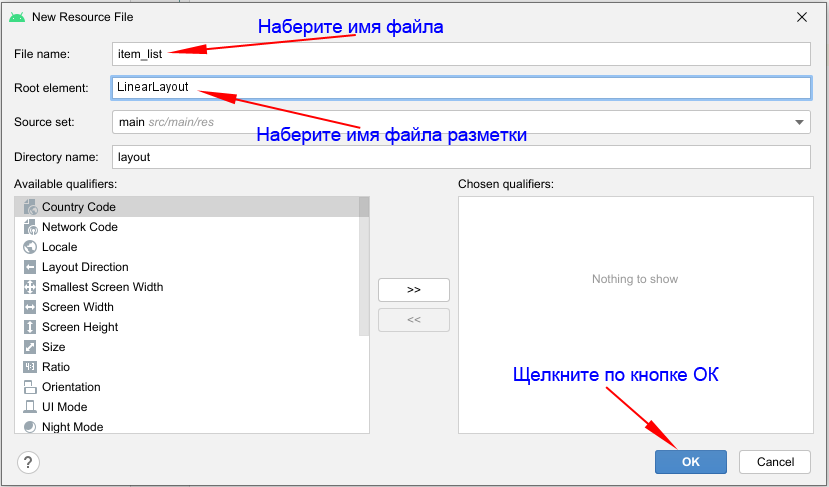
<androidx.recyclerview.widget.RecyclerView

android:id="@+id/rv"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

1. Добавьте файл разметки с именем item\_list типа LinearLayout.
   1. В ресурсах в папке res щелкните правой кнопкой по папке layout и выберите New → Layout Resource File → Окно нового ресурса.
   2. Выполните действия, указанные на скриншоте:



1. Вставьте в этот файл разметку:

<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="wrap\_content"

android:padding="1dp"

android:orientation="vertical">

<androidx.cardview.widget.CardView

android:id="@+id/cv"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginEnd="20dp"

app:cardCornerRadius="5dp"

app:cardBackgroundColor="@color/teal\_200"

android:elevation="3dp"

app:contentPadding="1dp">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:background="@android:color/white"

android:padding="8dp">

<ImageView

android:id="@+id/imgView"

android:layout\_width="87dp"

android:layout\_height="64dp"

android:layout\_marginStart="10dp"

android:layout\_marginTop="5dp"

app:srcCompat="@drawable/bmw"

tools:ignore="ContentDescription" />

<TextView

android:id="@+id/txtView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="TextView"

android:layout\_marginStart="20dp"

android:layout\_marginTop="20dp"

tools:ignore="HardcodedText" />

</LinearLayout>

</androidx.cardview.widget.CardView>

</RelativeLayout>

1. Добавьте класс SingleAuto и сгенерируйте конструктор и методы доступа

public class SingleAuto {

int img;

String name;

public SingleAuto(int img, String name) {

this.img = img;

this.name = name;

}

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;

}

}

1. Добавьте класс AutoAdapter и оформите адаптер.

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

private final LayoutInflater inflater;

private final ArrayList<Logo> mylst;

public AutoAdapter(Context context, ArrayList<Logo> mylst) {

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

this.mylst = mylst;

}

@NonNull

@Override

public AutoAdapter.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) {

Logo logo = mylst.get(position);

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

holder.txtView.setText(logo.getName());

}

@Override

public int getItemCount() {

return mylst.size();

}

public static class ViewHolder extends RecyclerView.ViewHolder {

ImageView imgView;

TextView txtView;

CardView cardView;

@SuppressLint("CutPasteId")

public ViewHolder(@NonNull View itemView) {

super(itemView);

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

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

cardView = itemView.findViewById(R.id.*cv*);

}

}

}

1. Импортируйте в адаптере все требуемые классы. Методы setImageResource и другие не будут подчеркнуты красным после импорта классов в классе ViewHolder.
2. В классе MainActivity объявите глобально список:

public ArrayList<Logo> logo;

1. В классе MainActivity вставьте метод для инициализации списка логотипов фирм (метод вставляется после фигурной скобки, закрывающей метод onCreate):

public void SetLogoInitial(){

logos.add(new Logo(R.drawable.*a\_bmwlogo*, "Германия"));

logos.add(new Logo(R.drawable.*a\_chevroletlogo*, "США"));

logos.add(new Logo(R.drawable.*a\_fordlogo*, "США"));

logos.add(new Logo(R.drawable.*a\_hundailogo*, "Южная Корея"));

logos.add(new Logo(R.drawable.*a\_ladalogo*, "Россия"));

}

1. Добавьте локальные объявления и инициализацию

*// Связь разметки с кодом*

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

recyclerView.setLayoutManager(new LinearLayoutManager(this));

SetLogoInitial ();

1. Подключите адаптер

AutoAdapter adapter = new AutoAdapter(MainActivity.this, mylst);

*// устанавливаем для списка адаптер*

recyclerView.setAdapter(adapter);

1. Оформите инициализацию списка

public void setInitialData(){

mylst.add(new SingleAuto(R.drawable.*bmw*, "BMW"));

mylst.add(new SingleAuto(R.drawable.*chevrolet*, "Шевроле"));

mylst.add(new SingleAuto(R.drawable.*ford*, "Форд"));

mylst.add(new SingleAuto(R.drawable.*hundai*, "Хундай"));

mylst.add(new SingleAuto(R.drawable.*lada*,"Лада"));

mylst.add(new SingleAuto(R.drawable.*mersedes*,"Мереседес"));

mylst.add(new SingleAuto(R.drawable.*toyota*, "Порше"));

}

1. Добавьте глобальные объявления

private final ArrayList<SingleAuto> mylst = new ArrayList<>();

RecyclerView recyclerView;

1. В адаптере создайте обработчик выбора позиции
   1. В адаптере объявить интерфейс

interface OnSingleAutoClickListener{

void onSingleAutoClick(SingleAuto sa, int position);

}

Здесь SingleAuto - имя класса для объектов списка, после описания и создания обработчика изменять нельзя.

* 1. Объявить переменную с именем интерфейса (имя произвольное, записанное со строчной буквы).

private final OnSingleAutoClickListener handle;

* 1. Объявленная строка подчеркнута красным. Щелкните правой кнопкой по ней и из списка Show Context Actions в появившемся меню выберите позицию:

Add constructor parameter (в появившемся затем окне щелкните по кнопке Refactor)

1. В адаптере (AutoAdapter) в метод onBindViewHolder вставить

holder.itemView.setOnClickListener(new View.OnClickListener(){

@Override

public void onClick(View v)

{

handle.onSingleAutoClick(SingleAuto, position);

}

});

Здесь имя метода onSingleAutoClick из объявленного интерфейса, остальные имена системные. Метод setOnClickListener принадлежит itemView. Метод onClick добавляется автоматически.

1. Параметр position в методе onBindViewHolder подчеркнут красным, по Show Context Actions выбрать Supress (и т. д.).
2. Имя (SingleAuto) параметра для имени объекта класса в методе onSingleAutoClick должно совпадать с объявленным в методе onBindViewHolder.
3. Если в пункте 2 было выбрано произвольное имя (handle) вставить его вместо onClickListener. (уже вставлено)
4. Перейти в MainActivity и вставить:

RecycleAdapter.OnSingleAutoClickListener vasja = new

RecycleAdapter.OnSingleAutoClickListener() {

@Override

public void onSingleAutoClick(SingleAuto blk, int position) {

Toast.*makeText*(getApplicationContext(), "Был выбран пункт " +

spisok.get(position).getText(), Toast.*LENGTH\_SHORT*).show();

}

};

*// создаем адаптер*

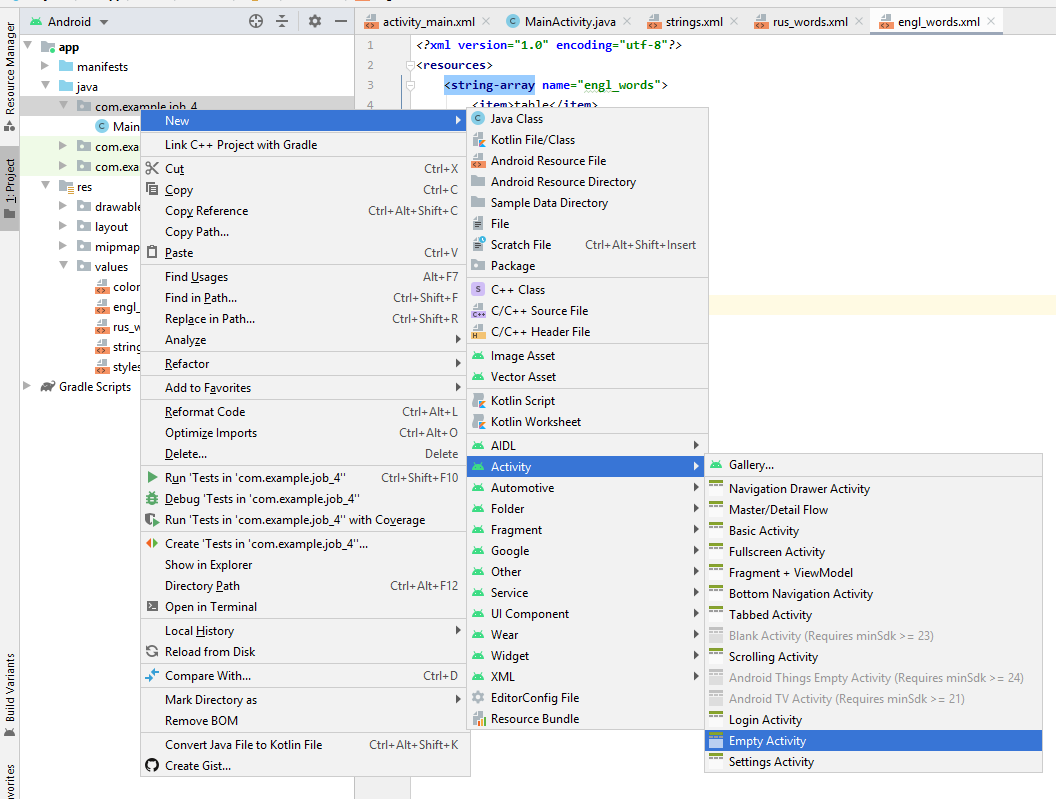
RecycleAdapter adapter = new RecycleAdapter(vasja, this, spisok);

*// устанавливаем для списка адаптер*

recyclerView.setAdapter(adapter);

Здесь RecycleAdapter - имя адаптера, spisok - имя обобщенного списка

1. Проверить работу
2. Для показа фотографии автомобилей добавим в приложение страницу. Для этого щелкните правой кнопкой по строке MainActivity, пройдите указанные меню и щелкните по позиции Empty Activity.



1. В появившемся окне наберите имя новой страницы (ShowActivity) и щелкните по кнопке Finish.
2. Перейдите в файл activity\_show.xml и выполните следующие действия:
   1. Замените разметку ConstraintLayout на RelativeLayout.
   2. Вставьте элементы ImageView, TextView и Button. Разметка должна иметь вид:

<ImageView

android:id="@+id/imageView"

android:layout\_width="387dp"

android:layout\_height="503dp"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_alignParentEnd="true"

android:layout\_alignParentBottom="true"

android:layout\_marginStart="14dp"

android:layout\_marginTop="26dp"

android:layout\_marginEnd="10dp"

android:layout\_marginBottom="202dp"

app:srcCompat="@drawable/bmw" />

<TextView

android:id="@+id/txtpoem"

android:layout\_width="wrap\_content"

android:layout\_height="106dp"

android:layout\_alignParentStart="true"

android:layout\_alignParentEnd="true"

android:layout\_alignParentBottom="true"

android:layout\_marginStart="55dp"

android:layout\_marginEnd="36dp"

android:layout\_marginBottom="73dp"

android:textColor="#272424" />

<Button

android:id="@+id/retbt"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentEnd="true"

android:layout\_alignParentBottom="true"

android:layout\_marginStart="96dp"

android:layout\_marginEnd="121dp"

android:layout\_marginBottom="21dp"

android:text="Возврат" />

Не забудьте в элементе ImageView указать первый автомобиль из списка (app:srcCompat="@drawable/bmw"), а в элементе TextView изменить цвет текста (цвет по умолчанию слишком бледный) android:textColor="#272424".

Примечание: свойство app:srcCompat="@drawable/bmw" в описании изображения ImageView обязательно. Это первое отображаемое в списке изображение (с индексом 0) для группы, заданной первыми символами, остальные изображения выбираются системой для этой группы в порядке их размещения в папке (по индексу).

1. В классе MainActivity Вставьте список. Для этого:
   1. Объявите глобально и инициализируйте массив строк списка:

String [] kinds; *// Массив строк*

* 1. Добавьте объявление адаптера и закрепление за списком:

ListView listView = (ListView) findViewById(R.id.*list\_item*);

ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,

android.R.layout.*simple\_list\_item\_1*, kinds);

listView.setAdapter(adapter);

1. В классе MainActivity добавьте обработчик щелчка по позиции списка:
   1. наберите приведенную ниже строку:

listR.setOnItemClickListener(this);

* 1. Красным цветом будет подчеркнуто this. Щелкните правой кнопкой по this и в меню Show Context Action выберите позицию:

Make ‘Main\_Activity’ implement ‘android.view.View.OnItemClickListener’

* 1. В появившемся окне щелкните по кнопке ОК:
  2. Появится обработчик, в который вставьте строки:
  3. ShowActivity

1. Добавьте на странице ShowActivity обработчик щелчка по кнопке Button:
   1. Объявите глобально кнопку Button.

Button retbt;

* 1. Закрепите за кнопкой ресурс (в методе onCreate):

retbt = (Button) findViewById(R.id.*retbt*);

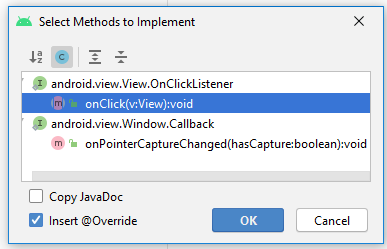
* 1. Наберите там же:

bt.setOnClickListener(this);

* 1. Красным цветом будет подчеркнуто this. Щелкнуть правой кнопкой по this и в меню Show Context Action выбрать

Make ‘Main\_Activity’ implement ‘android.view.View.OnClickListener’

* 1. В появившемся окне выбрать позицию onClick(v:View):void:



Щелкните по кнопке ОК.

1. В обработчик щелчка по выбранной позиции вставьте операторы перехода на вторую страницу:

Intent intent = new Intent(MainActivity.this, ShowActivity.class);

int pos = i; *// i – выбранная позиция, ее надо передать на вторую страницу*

intent.putExtra("pos", pos); *// “pos” имя для передачи*

this.startActivity(intent);

1. В обработчик щелчка по кнопке retbt вставьте операторы перехода обратно:

Intent intent = new Intent(ShowActivity.this, MainActivity.class);

this.startActivity(intent);

1. Проверьте работу приложения. Должен выполняться переход между страницами.
2. На странице ShowActivity объявите глобально:

Button retbt;

ImageView imgV;

TextView textView;

String [] str; *// Список текстов для автомобилей*

*// Список звуков*

String [] autotype = { "bmw.", "chevrolet.", "ford.",

"newlada.", "mersedes.", "toyota.", "solovey.", };

1. Оформите метод для отображения автомобилейы и текста по выбранной на первой странице позиции:

public void autochoose(int pos){

switch (pos){

case 0:

imgV.setImageResource(R.drawable.*bmw*);

textView.setText(str[pos]);

break;

case 1:

imgV.setImageResource(R.drawable.*chevrolet*);

textView.setText(str[pos]);

break;

case 2:

imgV.setImageResource(R.drawable.*ford*);

textView.setText(str[pos]);

break;

case 3:

imgV.setImageResource(R.drawable.*oriolus*);

textView.setText(str[pos]);

break;

case 4:

imgV.setImageResource(R.drawable.*mersedes*);

textView.setText(str[pos]);

break;

case 5:

imgV.setImageResource(R.drawable.*toyota*);

textView.setText(str[pos]);

break;

case 6:

imgV.setImageResource(R.drawable.*hundai*);

textView.setText(str[pos]);

break;

}

}

1. Обработка на второй странице:

*// Закрепление элементов за ресурсами*

imgV = (ImageView)findViewById(R.id.*imageView*);

textView = (TextView)findViewById(R.id.*txtpoem*);

retbt = (Button) findViewById(R.id.*retbt*);

str =getResources().getStringArray(R.array.*poem*); *// Заполнение списка*

retbt.setOnClickListener(this);

*// Прием выбранной позиции с предыдущей страницы*

Bundle arguments = getIntent().getExtras();

int ind = arguments.getInt("pos");

autochoose(ind);*// Вывод на экран выбранного автомобиля*

1. В обработчике для retbt:

Intent intent = new Intent(ShowActivity.this, MainActivity.class);

this.startActivity(intent);

1. Проверьте работу приложения и предъявите его преподавателю.