

Отчёт по индивидуальному практическому заданию

№3 Цель контрольной работы.

Изучить основные элементы пользовательского интерфейса и размещение их на различных типах слоев. Описание интерфейса с помощью языка разметки в ресурсах приложения. Работа со списками и адапторами.

Задание

Приложение должно поддерживать 2 варианта пользовательского интерфейса: для портретной и ландшафтной ориентации экрана. Также должна быть реализована поддержка как минимум 2 языков для текстовых строк на экране: русский и английский (активируются путем выбора языковой локали в свойствах смартфона\планшета)

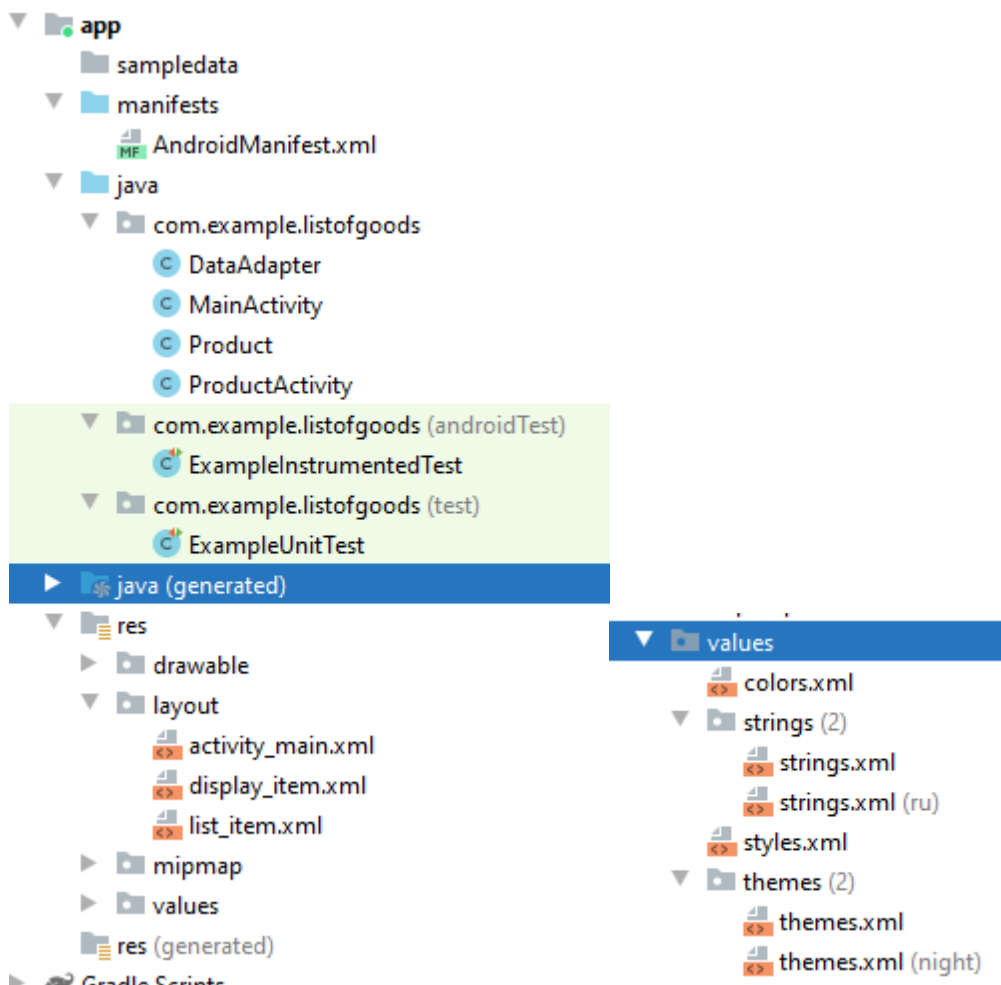
Вариант 7.

Список товаров. Каждый товар содержит картинку, название, цену. Возможность добавлять\удалять\редактировать.

Исходный код и видео работы программы есть на гит-хабе:

<https://github.com/ElephantT/Android>

Структура приложения:



DataAdapter – работа с данными товара

MainActivity - работа с главной страницей, где показаны все товары, можно открыть каждый товар и посмотреть его, можно добавить новый, и там же происходит инициализация стартовых товаров.

Product - описание класса товара

ProductActivity - основной функционал приложения по работе с товарами

Values

Цвета:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#6200EE</color>
    <color name="colorPrimaryDark">#3700B3</color>
    <color name="colorAccent">#03DAC5</color>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
</resources>
```

Русский и английский переводы:

```
<resources>
    <string name="app_name">ListOfGoods</string>
    <string name="save">Save</string>
    <string name="delete">Delete</string>
    <string name="cancel">Cancel</string>
    <string name="create">Create</string>
    <string name="hintName">Name</string>
    <string name="hintStore">Store</string>
    <string name="hintDate">Date</string>
</resources>
```

```
<resources>
    <string name="app_name">ListOfGoods</string>
    <string name="save">Сохранить</string>
    <string name="delete">Удалить</string>
    <string name="cancel">Отмена</string>
    <string name="create">Создать</string>
    <string name="hintName">Название</string>
    <string name="hintStore">Магазин</string>
    <string name="hintDate">Год</string>
</resources>
```

MainActivi переменные

```
public class MainActivity extends AppCompatActivity {  
  
    private List<Product> products = new ArrayList();  
    private DataAdapter productAdapter;  
    private static final int REQUEST_CHANGE_TYPE = 1;  
    private static final int REQUEST_CREATE_TYPE = 2;  
    private Product selectedProduct;
```

```
    ListView productList;
```

Начальная инициализация списка, затем получаем элемент ListView, создаём адаптер и делаем возможность слушать выбор в списке

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    setInitialData();  
    productList = (ListView) findViewById(R.id.albumList);  
    productAdapter = new DataAdapter( context: this, R.layout.list_item, products);  
    productList.setAdapter(productAdapter);  
    AdapterView.OnItemClickListener itemListener = new AdapterView.OnItemClickListener() {  
        @Override  
        public void onItemClick(AdapterView<?> parent, View v, int position, long id) {  
            selectedProduct = (Product) parent.getItemAtPosition(position);  
  
            Intent intent = new Intent( packageContext: MainActivity.this, ProductActivity.class);  
            intent.putExtra( name: "create", value: false);  
            intent.putExtra( name: "name", selectedProduct.getName());  
            intent.putExtra( name: "store", selectedProduct.getStore());  
            intent.putExtra( name: "image", selectedProduct.getImage());  
            intent.putExtra( name: "date", selectedProduct.getDate());  
            startActivityForResult(intent, REQUEST_CHANGE_TYPE);  
        }  
    };  
    productList.setOnItemClickListener(itemListener);  
}
```

```

public void onAddButtonClick(View v) {
    Intent intent = new Intent( packageContext: MainActivity.this, ProductActivity.class);
    intent.putExtra( name: "create", value: true);
    startActivityForResult(intent, REQUEST_CREATE_TYPE);
}

private void setInitialData(){
    products.add(new Product( name: "Apple", store: "Евроопт", R.drawable.p1, date: "2020"));
    products.add(new Product( name: "Cucumber", store: "Евроопт", R.drawable.p2, date: "2020"));
    products.add(new Product( name: "Orange", store: "Евроопт", R.drawable.p3, date: "2021"));
    products.add(new Product( name: "Banana", store: "Корона", R.drawable.p4, date: "2020"));
}

```

Обработка результата активности по изменению или
добавлению товара

```

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data){
    if(requestCode == REQUEST_CHANGE_TYPE) {
        if(resultCode == RESULT_OK) {
            if (data.getExtras().getBoolean( key: "delete"))
            {
                productAdapter.remove(selectedProduct);
                productAdapter.notifyDataSetChanged();
                return;
            }
            selectedProduct.setName(data.getExtras().getString( key: "name"));
            selectedProduct.setStore(data.getExtras().getString( key: "store"));
            selectedProduct.setDate(data.getExtras().getString( key: "date"));
            selectedProduct.setImage(data.getExtras().getInt( key: "image"));
            productAdapter.notifyDataSetChanged();
        }
    }
    if (requestCode == REQUEST_CREATE_TYPE) {
        if (resultCode == RESULT_OK) {
            Product newProduct = new Product(data.getExtras().getString( key: "name"),
                data.getExtras().getString( key: "store"),
                data.getExtras().getInt( key: "image"), data.getExtras().getString( key: "date"));
            productAdapter.add(newProduct);
            productAdapter.notifyDataSetChanged();
        }
    }
    else{
        super.onActivityResult(requestCode, resultCode, data);
    }
}

```

Активация Windows
Чтобы активировать Wind

Xml-код

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <Button
            android:id="@+id/addButton"
            android:text="@string/create"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:onClick="onAddButtonClick"/>
        </LinearLayout>
        <ListView android:id="@+id/albumList"
            android:layout_height="match_parent"
            android:layout_width="match_parent" />
    </LinearLayout>
```

Product

```
public class Product {  
    private int image;  
    private String name;  
    private String store;  
    private String date;  
  
    public Product(String name, String store, int image, String date){  
        this.name = name;  
        this.store = store;  
        this.image = image;  
        this.date = date;  
    }  
  
    public String getName() { return this.name; }  
  
    public void setName(String name) { this.name = name; }  
  
    public String getStore() { return this.store; }  
  
    public void setStore(String store) { this.store = store; }  
  
    public int getImage() { return this.image; }  
  
    public void setImage(int image) { this.image = image; }  
  
    public String getDate() { return this.date; }  
  
    public void setDate(String date) { this.date = date; }  
}
```

Xml-код отображения товара отдельно

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <EditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Name"
            android:id="@+id/editName" />

        <EditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Store"
            android:id="@+id/editStore" />

        <EditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Date"
            android:id="@+id/editDate" />

        <ImageButton
            android:layout_width="100dp"
```



```
<ImageButton
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:scaleType="fitCenter"
    android:id="@+id/imageButton"
    android:onClick="onImageButtonClick"/>

<Button
    android:id="@+id/button1"
    android:text="Save"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="onButton1Click"/>

<Button
    android:id="@+id/button2"
    android:text="Delete"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="onButton2Click"/>

<Button
    android:id="@+id/cancel"
    android:text="Cancel"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="onCancelClick" />

</LinearLayout>
</ScrollView>
```

Xml-код отображения товара в списке

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="16dp">
    <ImageView
        android:id="@+id/image"
        android:layout_marginRight="16dp"
        android:layout_marginEnd="16dp"
        android:layout_width="100dp"
        android:layout_height="100dp" />

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TextView
            android:id="@+id/name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />

        <TextView
            android:id="@+id/store"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />

        <TextView
            android:id="@+id/date"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />

    </LinearLayout>
</LinearLayout>
```

DataAdapter r

Переменные и конструктор

```
class DataAdapter extends ArrayAdapter<Product> {

    private LayoutInflater inflater;
    private int layout;
    private List<Product> products;

    public DataAdapter(Context context, int resource, List<Product> products) {
        super(context, resource, products);
        this.products = products;
        this.layout = resource;
        this.inflater = LayoutInflater.from(context);
    }

    public View getView(int position, View convertView, ViewGroup parent) {
        ViewHolder viewHolder;

        if(convertView==null){
            convertView = inflater.inflate(this.layout, parent, attachToRoot: false);
            viewHolder = new ViewHolder(convertView);
            convertView.setTag(viewHolder);
        }
        else{
            viewHolder = (ViewHolder) convertView.getTag();
        }
        Product product = products.get(position);

        viewHolder.imageView.setImageResource(product.getImage());
        viewHolder.nameView.setText(product.getName());
        viewHolder.storeView.setText(product.getStore());
        viewHolder.dateView.setText(product.getDate());

        return convertView;
    }
}
```

```

private class ViewHolder {
    final ImageView imageView;
    final TextView nameView, storeView, dateView;
    ViewHolder(View view){
        imageView = (ImageView) view.findViewById(R.id.image);
        nameView = (TextView) view.findViewById(R.id.name);
        storeView = (TextView) view.findViewById(R.id.store);
        dateView = (TextView) view.findViewById(R.id.date);
    }
}
}

```

ProductActivity

Просмотр товара или добавление нового

```

public class ProductActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display_item);

        Intent intent = getIntent();
        if (intent.getExtras().getBoolean( key: "create") == false) {
            EditText editName = (EditText) findViewById(R.id.editName);
            editName.setText(intent.getExtras().getString( key: "name"));
            EditText editStore = (EditText) findViewById(R.id.editStore);
            editStore.setText(intent.getExtras().getString( key: "store"));
            EditText editDate = (EditText) findViewById(R.id.editDate);
            editDate.setText(intent.getExtras().getString( key: "date"));
            ImageButton imageButton = (ImageButton) findViewById(R.id.imageButton);
            imageButton.setImageResource(intent.getExtras().getInt( key: "image"));
            iter = intent.getExtras().getInt( key: "image");
        }
        else {
            Button button = (Button) findViewById(R.id.button2);
            button.setVisibility(View.GONE);
            ImageButton imageButton = (ImageButton) findViewById(R.id.imageButton);
            imageButton.setImageResource(iter);
        }
    }
}

```

Отмена действия и итерация по картинкам (показано как работает в видео работы приложения, можно посмотреть на гит хабе)

```
public void onCancelClick(View v) {
    setResult(RESULT_CANCELED);
    finish();
}

private int iter = R.drawable.p0;
private int max = R.drawable.p4;
public void onImageButtonClick(View v) {
    ImageButton imageButton = (ImageButton) findViewById(R.id.imageButton);
    if (iter < max) iter++;
    else iter = R.drawable.p0;
    imageButton.setImageResource(iter);
}

public void onButton1Click(View v) {
    Intent data = new Intent();
    data.putExtra( name: "delete", value: false);
    EditText editName = (EditText) findViewById(R.id.editName);
    data.putExtra( name: "name",editName.getText().toString());
    EditText editStore = (EditText) findViewById(R.id.editStore);
    data.putExtra( name: "store",editStore.getText().toString());
    EditText editDate = (EditText) findViewById(R.id.editDate);
    data.putExtra( name: "date",editDate.getText().toString());
    data.putExtra( name: "image", iter);
    setResult(RESULT_OK,data);
    finish();
}

public void onButton2Click(View v) {
    Intent data = new Intent();
    data.putExtra( name: "delete", value: true);
    setResult(RESULT_OK,data);
    finish();
}
```

Достаточно много функционала, лучше всё
посмотреть в видео, но пару скриншотов всё же и тут
есть:

13:48



ListOfGoods

СОЗДАТЬ



Apple
Евроопт
2020



Cucumber
Евроопт
2020



Orange
Евроопт
2021



Ванана
Корона
2020

ListOfGoods

Apple

Евроопт

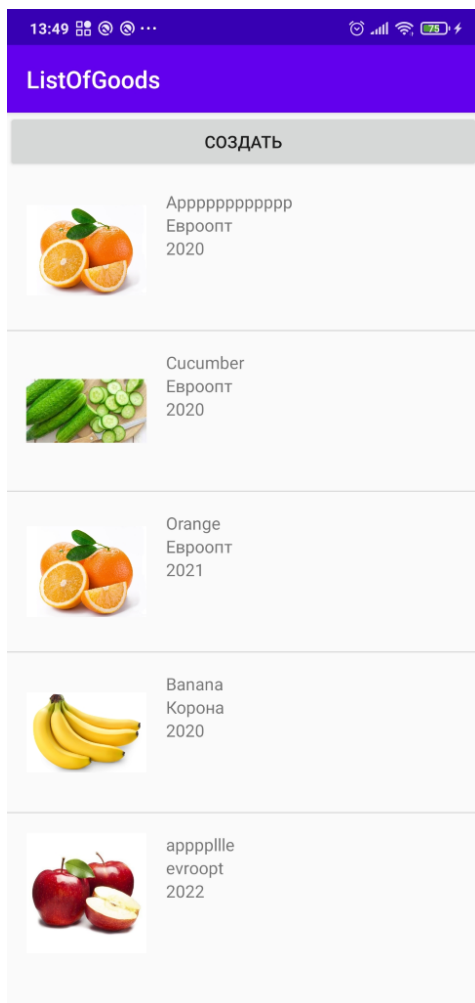
2020



СОХРАНИТЬ

УДАЛИТЬ

OTMEHA



Александр Дубейковский, студент 893551