Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра програмування та захисту інформації

**Звіт**

з виконаної лабораторної роботи № 5

дисципліни “ Кроссплатформені мови програмування ”

на тему

«ОСНОВИ РОБОТИ ЗІ СПИСКАМИ ДАНИХ»

Виконав :

студент академічної групи КІ-15

Аннаєв А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Перевірив :

Викладач

Смiрнова Н.В.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кропивницкий- 2017

**Лабораторная работа №5**

**Тема**: Основы разработки со списками данных.

**Цель**: Получить практические навыки разработки мобильных приложений под

управлением ОС ANDROID.

**ЗАДАНИЕ**

Используя имеющуюся электронную документацию создать мобильное ПО под управлением ОС Android. Состоящее из списка (ListView). В списке воспроизводится следующая информация (рис. 1, а): любые текстовые данные в правой части экрана строки и любая картинка в левой части экрана этой же строки. Список должен содержать не менее десяти строк с описанными выше данными.

1. При нажатии на строку или на один из его элементов (рис. 1, б) нужно показать пользователю информацию по выбранному строке (Toast message).



Рисунок 1 - Пример окна ПО

public class MainActivity extends AppCompatActivity {

private List<ListItem> mList = new ArrayList<>();

private RecyclerView recyclerView;

private Adapter mAdapter;

private Intent intent;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

intent = new Intent(this,InfoActivity.class);

recyclerView = (RecyclerView) findViewById(R.id.recycler\_view);

mAdapter = new Adapter(mList);

RecyclerView.LayoutManager mLayoutManager = new LinearLayoutManager(getApplicationContext());

recyclerView.setLayoutManager(mLayoutManager);

recyclerView.setItemAnimator(new DefaultItemAnimator());

recyclerView.setAdapter(mAdapter);

recyclerView.addOnItemTouchListener(new RecyclerItemClickListener(getApplicationContext(),

recyclerView, new OnItemClickListener() {

@Override

public void onItemClick(View view, int position) {

switch (position){

case 0:

Toast.makeText(MainActivity.this,

"You choose the game item ",

Toast.LENGTH\_SHORT).show();

break;

case 1:

Toast.makeText(MainActivity.this,

"You choose the flower item",

Toast.LENGTH\_SHORT).show();

break;

case 2:

Toast.makeText(MainActivity.this,

"You choose the car item",

Toast.LENGTH\_SHORT).show();

break;

case 3:

Toast.makeText(MainActivity.this,

"You choose the actor item",

Toast.LENGTH\_SHORT).show();

break;

case 4:

Toast.makeText(MainActivity.this,

"You choose the house item",

Toast.LENGTH\_SHORT).show();

break;

} }

}

@Override

public void onLongItemClick(View view, int position) { }

}));

prepareData();

}

private void prepareData() {

ListItem item = new ListItem(" The Game :", R.drawable.ic\_1);

mList.add(item);

item = new ListItem("The Flower :", R.drawable.ic\_2);

mList.add(item);

item = new ListItem("The Car: ", R.drawable.ic\_3);

mList.add(item);

item = new ListItem("The Actor :", R.drawable.ic\_4);

mList.add(item);

item = new ListItem("The House :", R.drawable.ic\_5);

mList.add(item);

mAdapter.notifyDataSetChanged();

}

}

public class Adapter extends RecyclerView.Adapter<Adapter.MyViewHolder> {

private List<ListItem> itemList;

public Adapter(List<ListItem> itemList) {

this.itemList = itemList;

}

public class MyViewHolder extends RecyclerView.ViewHolder {

public TextView title;

public ImageView img;

public MyViewHolder(View view) {

super(view);

title = (TextView) view.findViewById(R.id.title);

img = (ImageView) view.findViewById(R.id.imageView);

}

}

@Override

public MyViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {

View itemView = LayoutInflater.from(parent.getContext())

.inflate(R.layout.list\_item, parent, false);

return new MyViewHolder(itemView);

}

@Override

public void onBindViewHolder(MyViewHolder holder, int position) {

ListItem item = itemList.get(position);

holder.title.setText(item.getTitle());

holder.img.setImageResource(item.getImageId());

}

@Override

public int getItemCount() {

return itemList.size();

}

}

public class RecyclerItemClickListener implements RecyclerView.OnItemTouchListener {

private OnItemClickListener mListener;

public interface OnItemClickListener {

public void onItemClick(View view, int position);

public void onLongItemClick(View view, int position);

}

GestureDetector mGestureDetector;

public RecyclerItemClickListener(Context context, final RecyclerView recyclerView, OnItemClickListener listener) {

mListener = listener;

mGestureDetector = new GestureDetector(context, new GestureDetector.SimpleOnGestureListener() {

@Override

public boolean onSingleTapUp(MotionEvent e) {

return true;

}

@Override

public void onLongPress(MotionEvent e) {

View child = recyclerView.findChildViewUnder(e.getX(), e.getY());

if (child != null && mListener != null) {

mListener.onLongItemClick(child, recyclerView.getChildAdapterPosition(child));

}

}

});

}

@Override public boolean onInterceptTouchEvent(RecyclerView view, MotionEvent e) {

View childView = view.findChildViewUnder(e.getX(), e.getY());

if (childView != null && mListener != null && mGestureDetector.onTouchEvent(e)) {

mListener.onItemClick(childView, view.getChildAdapterPosition(childView));

return true;

}

return false;

}

@Override public void onTouchEvent(RecyclerView view, MotionEvent motionEvent) { }

@Override

public void onRequestDisallowInterceptTouchEvent (boolean disallowIntercept){}

}

public class ListItem {

private String title;

private int img;

public ListItem(String title, int image) {

this.title = title;

this.img = image;

}

public String getTitle() { return title; }

public int getImageId() { return img; }

}