**Лабораторная работа №7. Меню и toast-уведомлениями**

**Цель работы:** Научиться создавать меню, управлять состоянием меню с помощью групп, использовать toast-уведомлениями.

**Задание:**

Создать приложение, состоящее из единственного Activity, отображающего в центре экрана текущее время (часы и минуты), и дату в формате DD/MM/YYYY.

Добавить контекстное меню, вызываемое «длинным» касанием, позволяющее переключить формат времени между 12-ти и 24-ти часовыми режимами, а, также, отключать отображение даты.

В каждый момент времени в меню должно присутствовать только два пункта (используйте группы меню).

Добавьте сразу после загрузки Activity долгое toast-уведомление, сообщающее что для изменения настроек можно использовать «длинное» касание. Уведомление не должно повторно отображаться при повороте экрана.

Реализуйте сохранение текущих настроек при повороте экрана.

В верхней части экрана разместите надпись с номером лабораторной работы, в нижней части фамилию и инициалы автора.

Ход работы:

public class MainActivity extends AppCompatActivity {

final byte MENU\_GROUP\_TIME = 1;

final byte MENU\_ITEM\_24H = 0;

final byte MENU\_ITEM\_12H = 1;

final byte MENU\_GROUP\_DATE = 2;

final byte MENU\_ITEM\_DATE\_ON = 0;

final byte MENU\_ITEM\_DATE\_OFF = 1;

private byte OPTION\_TIME\_FORMAT = MENU\_ITEM\_12H;

private byte OPTION\_DATE\_STATE = MENU\_ITEM\_DATE\_ON;

final Handler handler=new Handler();

private boolean toasted = false;

private String datetime\_format;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

handler.postDelayed(updateTime,0);

handler.postDelayed(show\_toast,0);

ConstraintLayout parentLayout = findViewById(R.id.rootLayout);

registerForContextMenu(parentLayout);

updateSettings();

}

private void updateSettings() {

String TIME\_FORMAT="", DATE\_STATE="";

switch (OPTION\_TIME\_FORMAT){

case MENU\_ITEM\_24H: TIME\_FORMAT="HH:MM"; break;

case MENU\_ITEM\_12H: TIME\_FORMAT="hh:MM aa"; break;

}

switch (OPTION\_DATE\_STATE){

case MENU\_ITEM\_DATE\_ON: DATE\_STATE=" dd/MM/YYYY"; break;

case MENU\_ITEM\_DATE\_OFF: DATE\_STATE=""; break;

}

datetime\_format = TIME\_FORMAT+DATE\_STATE;

set\_current\_time();

}

@Override

public void onCreateContextMenu(ContextMenu menu, View v,

ContextMenu.ContextMenuInfo menuInfo) {

MenuItem saved\_item;

SubMenu subMenuTime = menu.addSubMenu("Настройки времени");

subMenuTime.add(MENU\_GROUP\_TIME, MENU\_ITEM\_24H, Menu.NONE, "24ч формат");

subMenuTime.add(MENU\_GROUP\_TIME, MENU\_ITEM\_12H, Menu.NONE, "12ч формат");

subMenuTime.setGroupCheckable(MENU\_GROUP\_TIME, true, true);

saved\_item = subMenuTime.getItem(OPTION\_TIME\_FORMAT);

saved\_item.setChecked(true);

SubMenu subMenuDate = menu.addSubMenu("Настройки даты");

subMenuDate.add(MENU\_GROUP\_DATE, MENU\_ITEM\_DATE\_ON, Menu.NONE, "Вкл");

subMenuDate.add(MENU\_GROUP\_DATE, MENU\_ITEM\_DATE\_OFF, Menu.NONE, "Выкл");

subMenuDate.setGroupCheckable(MENU\_GROUP\_DATE, true, true);

saved\_item = subMenuDate.getItem(OPTION\_DATE\_STATE);

saved\_item.setChecked(true);

}

@Override

public boolean onContextItemSelected(MenuItem item)

{

if(item.getGroupId()!=Menu.NONE)

switch (item.getGroupId()){

case MENU\_GROUP\_TIME:

OPTION\_TIME\_FORMAT= (byte) item.getItemId();

break;

case MENU\_GROUP\_DATE:

OPTION\_DATE\_STATE = (byte) item.getItemId();

break;

}

item.setChecked(true);

updateSettings();

return true;

}

final Runnable show\_toast=new Runnable() {

@Override

public void run() {

if(toasted)

return;

toasted = true;

Toast.makeText(MainActivity.this, R.string.Toast,

Toast.LENGTH\_LONG).show();

}

};

private void set\_current\_time() {

DateFormat df = new SimpleDateFormat(datetime\_format);

TextView tv = findViewById(R.id.DateTime);

tv.setText(df.format(Calendar.getInstance().getTime()));

}

final Runnable updateTime=new Runnable() {

@Override

public void run() {

set\_current\_time();

handler.postDelayed(this,1000);

}

};

@Override

public void onSaveInstanceState(Bundle savedInstanceState) {

savedInstanceState.putBoolean("toasted", toasted);

savedInstanceState.putByte("OPTION\_TIME\_FORMAT", OPTION\_TIME\_FORMAT);

savedInstanceState.putByte("OPTION\_DATE\_STATE", OPTION\_DATE\_STATE);

super.onSaveInstanceState(savedInstanceState);

}

@Override

public void onRestoreInstanceState(Bundle savedInstanceState) {

super.onRestoreInstanceState(savedInstanceState);

toasted = savedInstanceState.getBoolean("toasted");

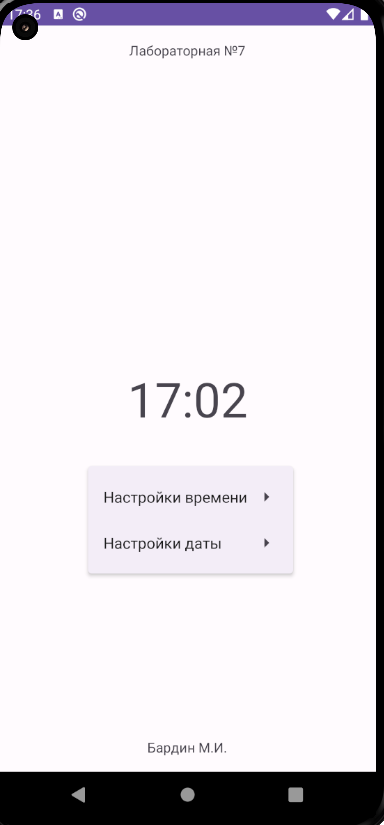
OPTION\_TIME\_FORMAT = savedInstanceState.getByte("OPTION\_TIME\_FORMAT");

OPTION\_DATE\_STATE = savedInstanceState.getByte("OPTION\_DATE\_STATE");

updateSettings();

}

}



**Вывод:** в ходе работы мы научились создавать меню, управлять состоянием меню с помощью групп, использовать toast-уведомлениями.