

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО”

Факультет інформатики та обчислювальної техніки

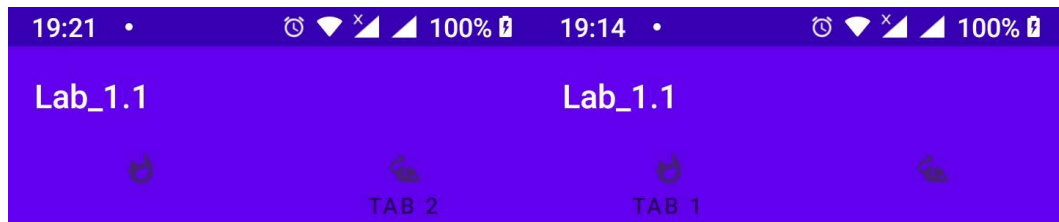
Кафедра обчислювальної техніки

Лабораторна робота №1.2
з дисципліни
“Програмування мобільних систем”

Виконала:
студентка групи ІО-82
ЗК ІО-8223
Трандашир Юлія

Київ 2021

Скріншот роботи додатку

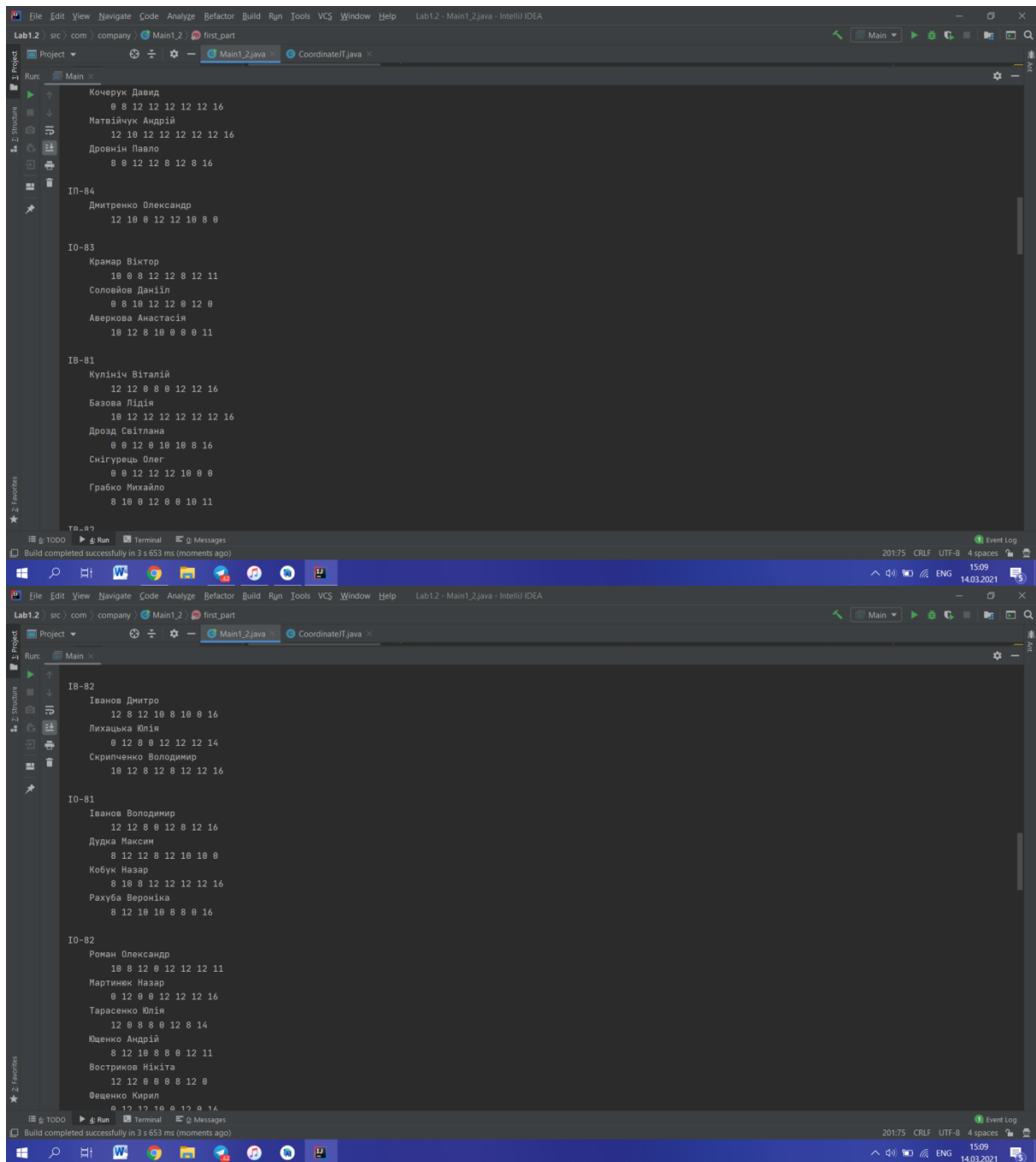


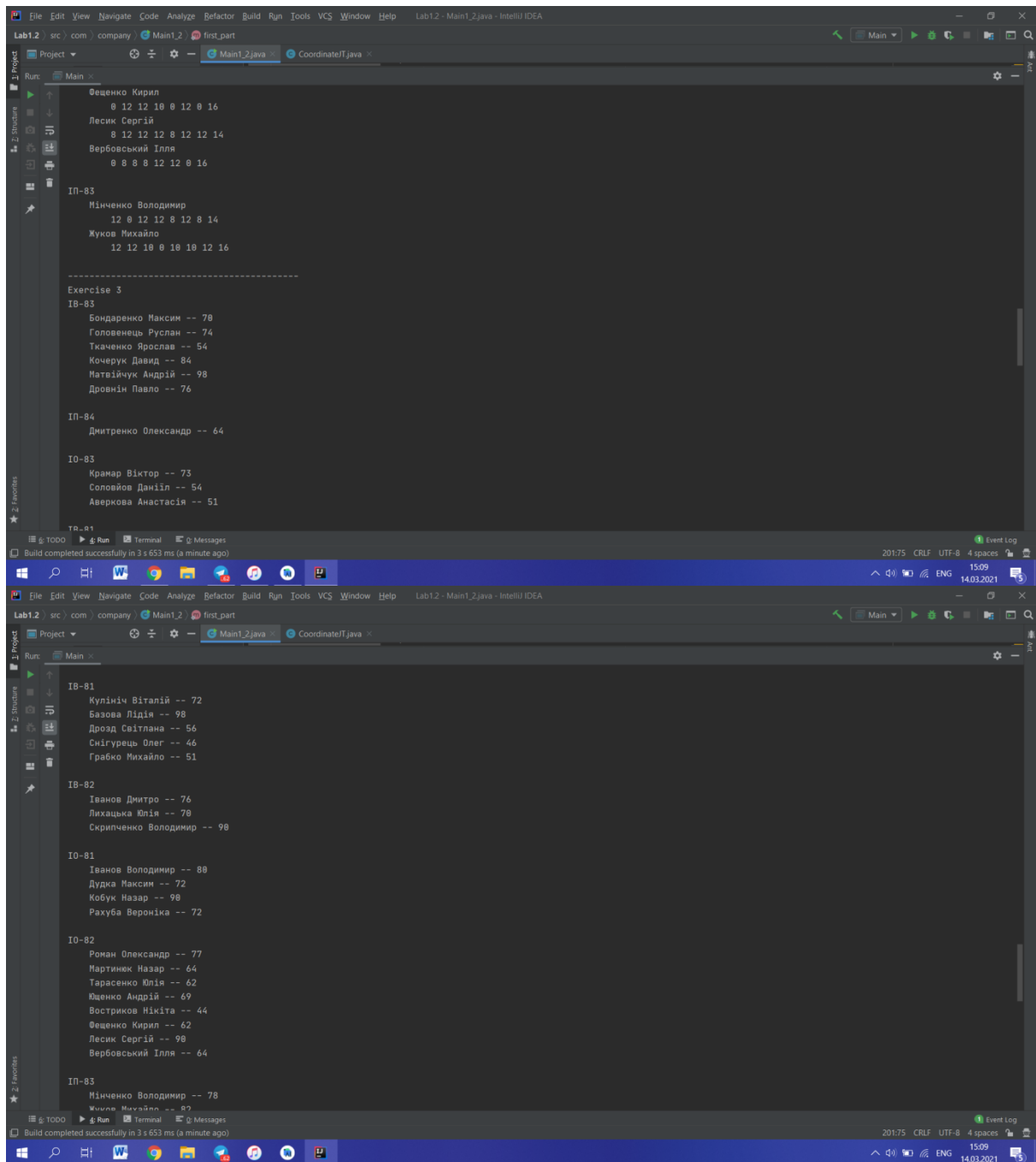
Трандашир Юлія
Група ІО-82
ЗК ІО-8223

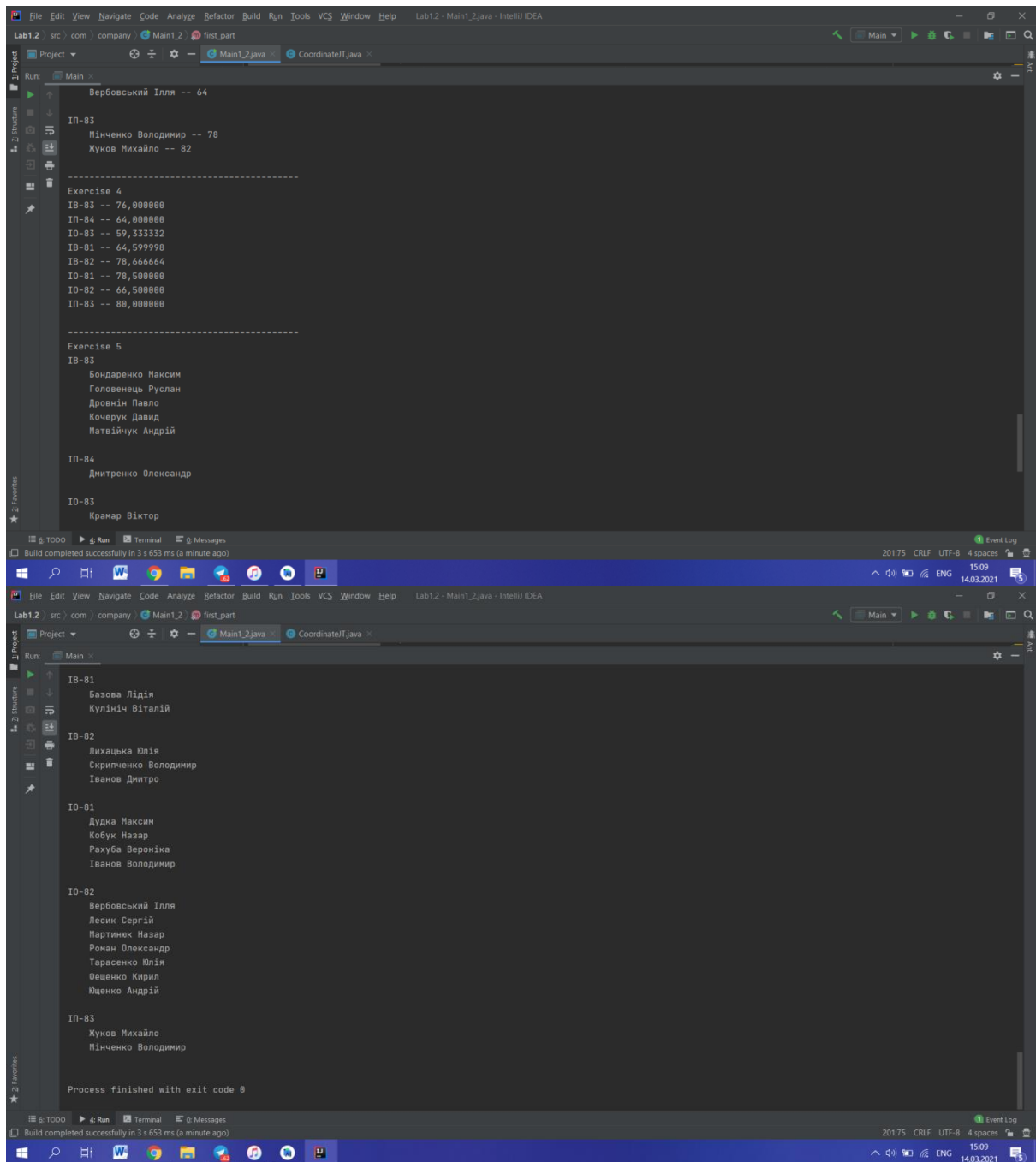


```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help Lab1.2 - Main1_2.java - IntelliJ IDEA
Lab1.2 | src | com | company | Main1_2 | first_part
Project | Main1_2.java | CoordinateT.java
Run: Main
C:\Users\Julia\.jdk\corretto-1.8.0_252\bin\java.exe ...
Exercise 1
IB-83
Бондаренко Максим
Головенець Руслан
Дровнін Павло
Кочерук Давид
Матвійчук Андрій
Ткаченко Ярослав
IP-84
Дмитренко Олександр
IO-83
Аверкова Анастасія
Крамар Віктор
Соловійов Даниїл
IB-81
Базова Лідія
Грабко Михайло
Дрозд Світлана
Купінін Віталій
Снігурець Олег
IB-82
Лихацька Юлія
Скрипченко Володимир
Іванов Дмитро
TB-81
Build completed successfully in 3 s 653 ms (moments ago)
201:75 CRLF UTF-8 4 spaces 15:09 14.03.2021
```

```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help Lab1.2 - Main1_2.java - IntelliJ IDEA
Lab1.2 | src | com | company | Main1_2 | first_part
Project | Main1_2.java | CoordinateT.java
Run: Main
IO-81
Дудка Максим
Кобук Назар
Рахуба Вероніка
Іванов Володимир
IO-82
Вербовський Ілля
Востриков Нікіта
Лесик Сергій
Мартинук Назар
Роман Олександр
Тарасенко Юлія
Фещенко Кирил
Юценко Андрій
IP-83
Жуков Михайло
Мінченко Володимир
-----
Exercise 2
IB-83
Бондаренко Максим
12 0 0 12 12 10 10 14
Головенець Руслан
12 10 0 8 10 10 8 16
Ткаченко Ярослав
12 8 0 0 8 12 0 14
Кочерук Давид
0 0 12 12 12 12 14
Build completed successfully in 3 s 653 ms (moments ago)
201:75 CRLF UTF-8 4 spaces 15:09 14.03.2021
```







The screenshot shows a Java application running in an IDE. The console output displays several coordinates (A, B, C, D) and the results of a 'mid' method. An exception is thrown: 'java.lang.Exception: Несумісний градус (190) в координатах довготи'. The stack trace points to 'CoordinateJT.java:43' and 'Main1_2.java:255'. The process ends with exit code 0.

```
Run: Main
C:\Users\Julia\.jdk\corretto-1.8.0_252\bin\java.exe ...

A: 0°0'0" N
B: 12,511111° N
C: 12°30'40" S
D: 150,924454° E

First mid method: 0°30'40" N
Second mid method: 0°30'40" N
Incorrect mid method: null

Несумісний градус (190) в координатах довготи
java.lang.Exception
    at com.company.CoordinateJT.<init>(CoordinateJT.java:43)
    at com.company.Main1_2.second_part(Main1_2.java:255)
    at com.company.Main1_2.main(Main1_2.java:10)

Process finished with exit code 0
```

Лістинг коду

Main1_2.java

```
package ua.kpi.comsys.io8223;

import android.annotation.SuppressLint;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.Random;

public class Main1_2 {
    public static void main(String[] args) {
        second_part();
    }

    @SuppressLint("DefaultLocale")
    private static void first_part(){

        // Part 1

        String studentsStr = "Дмитренко Олександр - ІП-84; Матвійчук Андрій - ІВ-83; Лесик Сергій - ІО-82; Ткаченко Ярослав - ІВ-83; Аверкова Анастасія - ІО-83; Соловійов Данііл - ІО-83; Рахуба Вероніка - ІО-81; Кочерук Давид - ІВ-83; Лихацька Юлія - ІВ-82; Головенець Руслан - ІВ-83; Ющенко Андрій - ІО-82; Вербовський Ілля - ІО-82; Мінченко Володимир - ІП-83; Мартинюк Назар - ІО-82; Базова Лідія - ІВ-81; Снігурець Олег - ІВ-81; Роман Олександр - ІО-82; Дудка Максим - ІО-81; Кулініч Віталій - ІВ-81; Жуков Михайло - ІП-83; Грабко Михайло - ІВ-81; Іванов Володимир - ІО-81; Востриков Нікіта - ІО-82; Бондаренко Максим - ІВ-83; Скрипченко Володимир - ІВ-82; Кобук Назар - ІО-81; Дровнін Павло - ІВ-83; Тарасенко Юлія - ІО-82; Дрозд Світлана - ІВ-81; Фещенко Кирил - ІО-82; Крамар Віктор - ІО-83; Іванов Дмитро - ІВ-82";

        // Ex1
        // Fill the ArrayList with
        // key - name of group
```

```

// definition - sorted list with students of this group

HashMap<String, ArrayList<String>> studentsGroups = new HashMap<>();

for (String st_gr :
    studentsStr.split("; ?")) {
    String[] st_gr_arr = st_gr.split(" ?- ");

    if (!studentsGroups.containsKey(st_gr_arr[1]))
        studentsGroups.put(st_gr_arr[1], new ArrayList<>());
    studentsGroups.get(st_gr_arr[1]).add(st_gr_arr[0]);
}

for (String group :
    studentsGroups.keySet()) {
    Collections.sort(studentsGroups.get(group),
String.CASE_INSENSITIVE_ORDER);
}

System.out.println("-----");
System.out.println("Exercise 1");
for (String key :
    studentsGroups.keySet()) {
    System.out.println(key);
    for (String student :
        studentsGroups.get(key)) {
        System.out.println("\t"+student);
    }
    System.out.println();
}

// list with max rating

int[] points = {12, 12, 12, 12, 12, 12, 12, 16};

// Ex2
// Fill the ArrayList with
// key - name of group
// definition - ArrayList with:
//     - key - student of this group
//     - definition - list with rating of student (fill the list with
random definitions using function `randomValue(maxValue: Int) -> Int`)

HashMap<String, HashMap<String, ArrayList<Integer>>> studentPoints = new
HashMap<>();

for (String group:
    studentsGroups.keySet()){

    if (!studentPoints.containsKey(group))
        studentPoints.put(group, new HashMap<>());

    for (String student :
        studentsGroups.get(group)) {

```



```

        studentPoints.get(group).put(student, new ArrayList<>());
        for (int point :
            points) {
            studentPoints.get(group).get(student).add(randomValue(point));
        }
    }
}

System.out.println("-----");
System.out.println("Exercise 2");
for (String key :
    studentPoints.keySet()) {
    System.out.println(key);
    for (String student :
        studentPoints.get(key).keySet()) {
        System.out.print("\t"+student+"\n\t\t");
        for (int p :
            studentPoints.get(key).get(student)) {
            System.out.print(p + " ");
        }
        System.out.println();
    }
    System.out.println();
}

// Ex3
// Fill the ArrayList with
// key - name of group
// definition - ArrayList with:
//     - key - student of this group
//     - definition - sum of rating of student

HashMap<String, HashMap<String, Integer>> sumPoints = new HashMap<>();

for (String group:
    studentsGroups.keySet()){

    if (!sumPoints.containsKey(group))
        sumPoints.put(group, new HashMap<>());

    for (String student :
        studentsGroups.get(group)) {
        int sum = 0;
        for (int point :
            studentPoints.get(group).get(student)) {
            sum += point;
        }

        sumPoints.get(group).put(student, sum);
    }
}

System.out.println("-----");
System.out.println("Exercise 3");
for (String key :

```

```

        sumPoints.keySet()) {
    System.out.println(key);
    for (String student :
        sumPoints.get(key).keySet()) {
        System.out.println(String.format("\t%s -- %d", student,
sumPoints.get(key).get(student)));
    }
    System.out.println();
}

// Ex4
//Fill the ArrayList with
// key - name of group
// definition- average rating of all students of group

HashMap<String, Float> groupAvg = new HashMap<>();

for (String group:
    studentsGroups.keySet()){
    int sum = 0, num = 0;
    for (String student :
        sumPoints.get(group).keySet()) {
        num++;
        sum += sumPoints.get(group).get(student);
    }
    groupAvg.put(group, (float)sum/num);
}

System.out.println("-----");
System.out.println("Exercise 4");
for (String key :
    groupAvg.keySet()) {
    System.out.println(String.format("%s -- %f", key, groupAvg.get(key)));
}
System.out.println();

// Ex5
//Fill the ArrayList with
// key - name of group
// definition - list of students, that have rating >= 60

HashMap<String, ArrayList<String>> passedPerGroup = new HashMap<>();

for (String group:
    studentsGroups.keySet()){

    if (!passedPerGroup.containsKey(group))
        passedPerGroup.put(group, new ArrayList<>());

    for (String student :
        studentsGroups.get(group)) {
        if (sumPoints.get(group).get(student) >= 60){
            passedPerGroup.get(group).add(student);
        }
    }
}

```

```

    }
}

System.out.println("-----");
System.out.println("Exercise 5");
for (String key :
    passedPerGroup.keySet()) {
    System.out.println(key);
    for (String student :
        passedPerGroup.get(key)) {
        System.out.println("\t"+student);
    }
    System.out.println();
}

}

private static int randomValue(int maxValue){
    Random rand = new Random();
    switch(rand.nextInt(6)) {
        case 1:
            return (int) (maxValue * 0.7);
        case 2:
            return (int) (maxValue * 0.9);
        case 3:
        case 4:
        case 5:
            return maxValue;
        default:
            return 0;
    }
}

private static void second_part(){
    CoordinateJT a = new CoordinateJT();
    CoordinateJT b, c, d;

    {
        try {
            b = new CoordinateJT(12, 30, 40, Direction.LATITUDE);
            c = new CoordinateJT(-12, 30, 40, Direction.LATITUDE);
            d = new CoordinateJT(150, 55, 28, Direction.LONGITUDE);
            System.out.println();

            System.out.println("A: " + a.getIntCoordinate());
            System.out.println("B: " + b.getFloatCoordinate());
            System.out.println("C: " + c.getIntCoordinate());
            System.out.println("D: " + d.getFloatCoordinate());

            System.out.println();

            System.out.println("First mid method: " + a.getMiddleCoordinate(b,
c).getIntCoordinate());
            System.out.println("Second mid method: " +
b.getMiddleCoordinate(c).getIntCoordinate());
            System.out.println("Incorrect mid method: " +
b.getMiddleCoordinate(d));

```

```

        System.out.println();

        CoordinateJT err = new CoordinateJT(190, 60, 60,
Direction.LONGITUDE);

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
}

```

CoordinateJT.java

```

package ua.kpi.comsys.io8223;

import android.annotation.SuppressLint;

public class CoordinateJT {

    private final Direction currentDir;
    private String worldLetter;
    private final int degree;
    private final int minute;
    private final int second;

    public CoordinateJT(){
        currentDir = Direction.LATITUDE;
        worldLetter = "N";
        degree = minute = second = 0;
    }

    @SuppressLint("DefaultLocale")
    public CoordinateJT(int deg, int min, int sec, Direction dir) throws Exception
    {
        currentDir = dir;
        if (dir == Direction.LATITUDE){
            if (deg >= -90 && deg <= 90){
                degree = deg;
                if (deg >= 0)
                    worldLetter = "N";
                else
                    worldLetter = "S";
            }
            else {
                System.err.println(String.format("Несумісний градус (%d) в координатах широти", deg));
                throw new Exception();
            }
        }
        else {
            if (deg >= -180 && deg <= 180){
                degree = deg;
                if (deg >= 0)
                    worldLetter = "E";
                else
                    worldLetter = "W";
            }
        }
    }
}

```

```

        else {
            System.err.println(String.format("Несумісний градус (%d) в
координатах довготи", deg));
            throw new Exception();
        }
    }

    if (min >= 0 && min <= 59){
        minute = min;
    }
    else {
        System.err.println(String.format("Несумісні мінути (%d) в
координатах", min));
        throw new Exception();
    }

    if (sec >= 0 && sec <= 59){
        second = sec;
    }
    else {
        System.err.println(String.format("Несумісні секунди (%d) в
координатах", sec));
        throw new Exception();
    }
}

@SuppressLint("DefaultLocale")
public String getIntCoordinate(){
    return String.format("%d°%d'%d\" %s", Math.abs(degree), minute, second,
worldLetter);
}

private float getFloatSigned(){
    return (Math.abs(degree) + (float)minute/60 + (float)second/3600) *
(degree >= 0? 1: -1);
}

@SuppressLint("DefaultLocale")
public String getFloatCoordinate(){
    return String.format("%f° %s", Math.abs(getFloatSigned()), worldLetter);
}

public CoordinateJT getMiddleCoordinate(CoordinateJT a, CoordinateJT b) throws
Exception {
    if (a.getCurrentDir() == b.getCurrentDir()){
        return new CoordinateJT((a.getDegree() + b.getDegree()) / 2,
(a.getMinute() + b.getMinute()) / 2,
(a.getSecond() + b.getSecond()) / 2,
a.getCurrentDir());
    }
    else {
        return null;
    }
}

public CoordinateJT getMiddleCoordinate(CoordinateJT second) throws Exception
{
    return getMiddleCoordinate(this, second);
}

```

```

    }

    public Direction getCurrentDir() {
        return currentDir;
    }

    public int getDegree() {
        return degree;
    }

    public int getMinute() {
        return minute;
    }

    public int getSecond() {
        return second;
    }
}

enum Direction {
    LATITUDE,
    LONGITUDE
}

```

PageViewModel.java

```

package ua.kpi.comsys.io8223.ui.main;

import androidx.arch.core.util.Function;
import androidx.lifecycle.LiveData;
import androidx.lifecycle.MutableLiveData;
import androidx.lifecycle.Transformations;
import androidx.lifecycle.ViewModel;

public class PageViewModel extends ViewModel {

    private MutableLiveData<Integer> mIndex = new MutableLiveData<>();
    private LiveData<String> mText = Transformations.map(mIndex, new
Function<Integer, String>() {
        @Override
        public String apply(Integer input) {
            return "Hello world from section: " + input;
        }
    });

    public void setIndex(int index) {
        mIndex.setValue(index);
    }

    public LiveData<String> getText() {
        return mText;
    }
}

```

PlaceholderFragment.java

```

package ua.kpi.comsys.io8223.ui.main;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.Nullable;
import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.lifecycle.Observer;
import androidx.lifecycle.ViewModelProvider;
import ua.kpi.comsys.io8223.R;

/**
 * A placeholder fragment containing a simple view.
 */
public class PlaceholderFragment extends Fragment {

    private static final String ARG_SECTION_NUMBER = "section_number";

    private PageViewModel pageViewModel;

    public static PlaceholderFragment newInstance(int index) {
        PlaceholderFragment fragment = new PlaceholderFragment();
        Bundle bundle = new Bundle();
        bundle.putInt(ARG_SECTION_NUMBER, index);
        fragment.setArguments(bundle);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        pageViewModel = new ViewModelProvider(this).get(PageViewModel.class);
        int index = 1;
        if (getArguments() != null) {
            index = getArguments().getInt(ARG_SECTION_NUMBER);
        }
        pageViewModel.setIndex(index);
    }

    @Override
    public View onCreateView(
        @NonNull LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        View root = inflater.inflate(R.layout.fragment_main, container, false);

        final TextView mainLable = root.findViewById(R.id.textView_identifier);
        pageViewModel.getText().observe(this, new Observer<String>() {
            @Override
            public void onChanged(@Nullable String s) {
            }
        });
        return root;
    }

```

```
}  
}
```

SectionsPagerAdapter.java

```
package ua.kpi.comsys.io8223.ui.main;  
  
import android.content.Context;  
import android.graphics.drawable.Drawable;  
import android.text.SpannableStringBuilder;  
import android.text.Spanned;  
import android.text.style.DynamicDrawableSpan;  
import android.text.style.ImageSpan;  
  
import androidx.annotation.Nullable;  
import androidx.annotation.StringRes;  
import androidx.fragment.app.Fragment;  
import androidx.fragment.app.FragmentManager;  
import androidx.fragment.app.FragmentPagerAdapter;  
  
import ua.kpi.comsys.io8223.FirstFragment;  
import ua.kpi.comsys.io8223.SecondFragment;  
import ua.kpi.comsys.io8223.R;  
  
/**  
 * A [FragmentPagerAdapter] that returns a fragment corresponding to  
 * one of the sections/tabs/pages.  
 */  
public class SectionsPagerAdapter extends FragmentPagerAdapter {  
  
    @StringRes  
    private static final int[] TAB_TITLES = new int[]{R.string.tab_text_1,  
R.string.tab_text_2};  
    private final Context mContext;  
    Drawable picture;  
    String name;  
  
    public SectionsPagerAdapter(Context context, FragmentManager fm) {  
        super(fm);  
        mContext = context;  
    }  
  
    @Override  
    public Fragment getItem(int position) {  
        // getItem is called to instantiate the fragment for the given page.  
        // Return a PlaceholderFragment (defined as a static inner class below).  
  
        Fragment fragment = null;  
        switch (position) {  
            case 0:  
                fragment = new FirstFragment();  
                break;  
            case 1:  
                fragment = new SecondFragment();  
                break;  
        }  
        return fragment;  
    }  
}
```



```

    }

    @Nullable
    @Override
    public CharSequence getPageTitle(int position) {
        switch (position) {
            case 0:
                picture = mContext.getResources().
                    getDrawable(R.drawable.ic_action_name);
                name = mContext.getResources().getString(TAB_TITLES[0]);
                break;
            case 1:
                picture = mContext.getResources().
                    getDrawable(R.drawable.ic_action_name2);
                name = mContext.getResources().getString(TAB_TITLES[1]);
                break;
            default:
                //TODO: handle default selection
                break;
        }

        SpannableStringBuilder sb = new SpannableStringBuilder(" \n" + name); //
        //space added before text for convenience

        picture.setBounds(5, 5, picture.getIntrinsicWidth(),
            picture.getIntrinsicHeight());
        ImageSpan span = new ImageSpan(picture,
            DynamicDrawableSpan.ALIGN_BASELINE);
        sb.setSpan(span, 0, 1, Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);
        return sb;
    }

    @Override
    public int getCount() {
        // Show 2 total pages.
        return 2;
    }
}

```

MainActivity.java

```

package ua.kpi.comsys.io8223;

import android.os.Bundle;

import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;
import com.google.android.material.tabs.TabLayout;

import androidx.viewpager.widget.ViewPager;
import androidx.appcompat.app.AppCompatActivity;

import android.view.Menu;
import android.view.MenuItem;
import android.view.View;

```

```

import ua.kpi.comsys.io8223.ui.main.SectionsPagerAdapter;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        SectionsPagerAdapter sectionsPagerAdapter = new SectionsPagerAdapter(this,
getSupportFragmentManager());
        ViewPager viewPager = findViewById(R.id.view_pager);
        viewPager.setAdapter(sectionsPagerAdapter);
        TabLayout tabs = findViewById(R.id.tabs);
        tabs.setupWithViewPager(viewPager);
        FloatingActionButton fab = findViewById(R.id.fab);

        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Snackbar.make(view, "Replace with your own action",
Snackbar.LENGTH_LONG)
                    .setAction("Action", null).show();
            }
        });
    }
}

```

FirstFragment.java

```

package ua.kpi.comsys.io8223;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;
public class FirstFragment extends Fragment{

    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable
ViewGroup container,
                           @Nullable Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_main, container, false);
    }
}

```

SecondFragment.java

```

package ua.kpi.comsys.io8223;

import android.os.Bundle;

```

```
import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public class SecondFragment extends Fragment {

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_second, container, false);
    }
}
```

Висновок

В результаті виконання лабораторної я додала 2 нових класи до проекту: `CoordinateJT` і `Main1_2`. `Main1_2` містить методи для роботи із заданим рядком, `CoordinateJT` описує координати і напрямки та містить методи для роботи із ними. Я перевірила працездатність додатку та коректність даних, що виводяться, на AVD та власному пристрої.