TASK GUIDE (C1.05)

A. **Objectives.**

Student will create share preferences for user's data and book's data.

B. **Requirements.**

Hardware:

- 2 GB RAM minimum, 8 GB RAM recommended
- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution
- Intel processor with support for Intel VT-x, Intel EM64T (Intel 64), and Execute Disable (XD) Bit functionality

Software:

- Microsoft Windows 7/8/10 (32-bit or 64-bit)
- JDK 8
- Android Studio IDE (Minimum 3.2) with AndroidX library.

C. **Resources.**

Documents:

- Guide

Supplement files:

Test code:

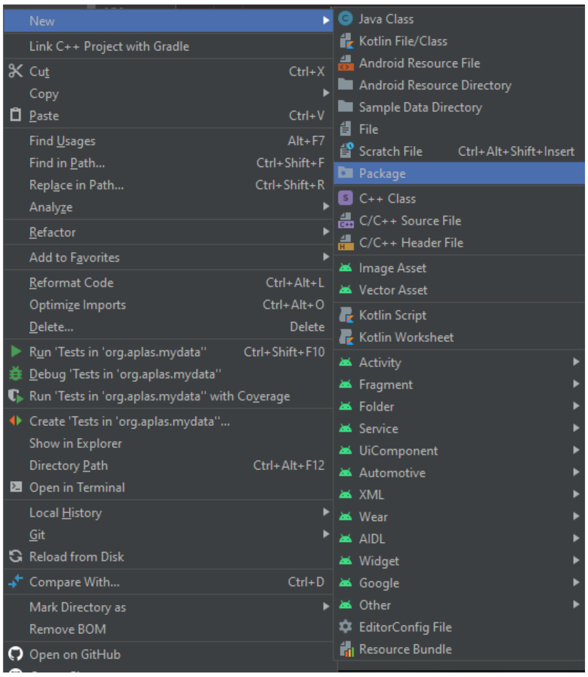
- TestC1MyLibrary071.java

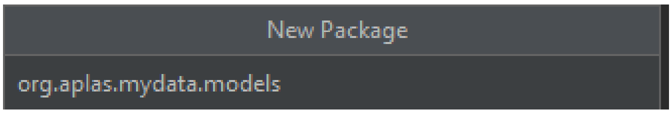
D. **Task Description.**

Student create class which save user's data and book's data. Then create method to call data with share preferences.

E. Specification.

1. Create package with name “models”

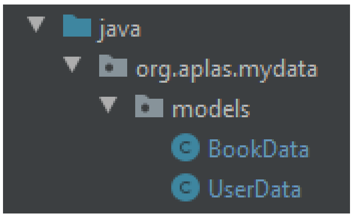




New Package

org.aplas.mydata.models

2. Create class with name “UserData.java” and “BookData.java” in “models” package



3. Open “UserData.java” and create final variables like table

```
“accessModifier” final “type” “name” = "value";
```

Access Modifier	Type	Name	Value
private	String	SHARED_ID	userdata
private	String	tag_name	name
private	String	tag_country	country
private	String	tag_phone	phone
private	String	tag_color	color

4. Create constructor that contain to getcollect share preferences. Share preferences use for save primitive data in internal file at application using key-value.

```
public UserData(Context c) {

        pref = c.getSharedPreferences(SHARED_ID, Context.MODE_PRIVATE);

    }
```

5. Create data binding method with name “getUserData” fill variable like table

```
Public User getUserData(){

       “type” “name” = “Preferences”.”getter”(“variable name”,
“unvalued”);

       return new User(“name”); }
```

Type	Name	Preferences	Getter	Variable Name	Unvalued
int	color	pref	getInt	tag_color	255*255*255
String	name	pref	getString	tag_name	
String	country	pref	getString	tag_country	
String	phone	pref	getString	tag_phone	

6. Create method with name “saveUserData” to catch the variable values like table

```
public void saveUserData(“type” “variable name”) {

        SharedPreferences.Editor editor = pref.edit();

        “share preferences editor”.”bundle”(“key value”,”variable
name”). ”function”();
```

Share Preferences Editor	Bundle	Key Value	Variable Name	Function
editor	putString	tag_name	name	apply
editor	putString	tag_country	country	apply
editor	putString	tag_phone	phone	apply
editor	putInt	tag_color	color	apply
editor				commit

7. Create method with name “clearUserData” to clear share preferences like below

```
public void clearUserData() {

        pref.edit().clear().apply();

    }
```

8. Create boolean method with name “isUserDataExist” to avoid duplication user like below

```
public boolean isUserDataExist() {

        return (pref.getAll().size() == 4) && (pref.contains(tag_name));

    }
```

9. Open “BookData.java” create variables like table

Access Modifier	**Key value**	**Value name**
private	Context	context
private	ArrayList<String>	fileList

And there are variables for save file in internal strotage

Access Modifier	Variable Type	Variable Name	Value
Private	String	FILE_EXT	.data
Private	String	DATA_LOCATION	

10. Create constructor that contains variable use to load file from internal storage like below

```
public BookData(Context c) {

        context=c;

        DATA_LOCATION = context.getFilesDir(). getAbsolutePath()
+File.separator;

        loadFileList();

    }
```

11. Create method with name “getBookData” for read data from internal storage and selection file if exist, like below

```
public Book getBookData(String fileName) {
        String fname = (fileName.isEmpty())?fileList.get(0):fileName;
        try {
            final File file = new File(DATA_LOCATION+fname);
            if (file.exists()) {
                BufferedReader reader;
                reader = new BufferedReader(new InputStreamReader(new
FileInputStream(file)));
                String line, title="", type="", year="";
                while((line = reader.readLine()) != null){
                    if (line.startsWith("Title:")) {
                        title = line.split(":")[1];
                    } else if (line.startsWith("Type:")) {
                        type = line.split(":")[1];
                    } else if (line.startsWith("Year:")) {
                        year = line.split(":")[1];
                    }
                }
                int number = fileList.indexOf(fileName);
                return new Book(number,title.trim(), type.trim(),
year.trim());
            } else {
                return null;}
        } catch (Exception e) {
            e.printStackTrace();
            return null; }
    }
```

12. Create method with name “getBookList” for load list of file in “BookList”

13.Create Boolean method to save book data with name “saveBookData”

```
public boolean saveBookData(String title, String type, String year) {

        String fname = getFileName(title);

        try {

            FileOutputStream fos = context.openFileOutput(fname,
Context.MODE_PRIVATE);

            OutputStreamWriter outputWriter = new
OutputStreamWriter(fos);

            outputWriter.write("Title: " +title+ "\nType: "

                    +type+ "\nYear: " +year);

            outputWriter.close();

            loadFileList();

            return true;

        } catch (IOException e) {

            e.printStackTrace();

            return false;

        }

    }
```

14. Create method “getFileName” to get name of file and replace “ “ to be “_” in file name.

15. Create private method “loadFileList” to load file in array list and check file

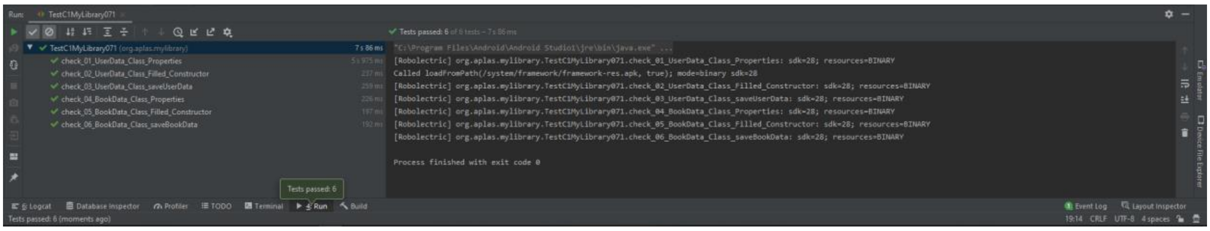
```
private void loadFileList() {
        fileList = new ArrayList<>();
        File directory = new File(DATA_LOCATION);
        File[] files = directory.listFiles();
        if (files != null){
            for (int x = 0; x < files.length; x++) {
                String fname = files[x].getName();
                if (fname.endsWith(FILE_EXT)) {
                    fileList.add(fname);
                }
            }
        }
```

Testing.

1. Copy “TestC1MyLibrary071.java” file to “org.aplas.Library (test)” folder.

2. Right click on the “TestC1MyLibrary071.java” file then choose Run. It may take long time to execute.

3. Get the result of your task. If passed you will get green check like below. If the test failed, you will get orange check get the messages and you must check your work again.



✓ Tests passed: 6 of 6 tests – 7 s 86 ms

```
✓ TestC1MyLibrary071 (org.aplas.mylibrary)            7 s 86 ms
    ✓ check_01_UserData_Class_Properties              5 s 975 ms
    ✓ check_02_UserData_Class_Filled_Constructor      237 ms
    ✓ check_03_UserData_Class_saveUserData            259 ms
    ✓ check_04_BookData_Class_Properties              226 ms
    ✓ check_05_BookData_Class_Filled_Constructor      197 ms
    ✓ check_06_BookData_Class_saveBookData            192 ms
```

```
"C:\Program Files\Android\Android Studio1\jre\bin\java.exe" ...
[Robolectric] org.aplas.mylibrary.TestC1MyLibrary071.check_01_UserData_Class_Properties: sdk=28; resources=BINARY
Called loadFromPath(/system/framework/framework-res.apk, true); mode=binary sdk=28
[Robolectric] org.aplas.mylibrary.TestC1MyLibrary071.check_02_UserData_Class_Filled_Constructor: sdk=28; resources=BINARY
[Robolectric] org.aplas.mylibrary.TestC1MyLibrary071.check_03_UserData_Class_saveUserData: sdk=28; resources=BINARY
[Robolectric] org.aplas.mylibrary.TestC1MyLibrary071.check_04_BookData_Class_Properties: sdk=28; resources=BINARY
[Robolectric] org.aplas.mylibrary.TestC1MyLibrary071.check_05_BookData_Class_Filled_Constructor: sdk=28; resources=BINARY
[Robolectric] org.aplas.mylibrary.TestC1MyLibrary071.check_06_BookData_Class_saveBookData: sdk=28; resources=BINARY

Process finished with exit code 0
```

Tests passed: 6

Tests passed: 6 (moments ago)

You have to try until get all green checkes and continue to the next task.