

GANPAT UNIVERSITY
B. TECH (COMPUTER ENGINEERING/INFORMATION TECHNOLOGY)
SEM-V REGULAR EXAMINATION– NOV-DEC 2020
2CEIT5PE5: MOBILE APPLICATION DEVELOPMENT

Time: 1.5 Hours]**[Total Marks: 30****Instructions:**

1. Figures to the right indicate full marks.
2. Be precise and to the point in your answer.
3. Assume the suitable data wherever necessary.

Q-1 As shown in Code of DatabaseHelper.kt, Write down only three methods: 1. deleteNote, 2. allNotes, 3. insertNote (No need to write given full code, just write down only three methods) Description of methods: **5**

1. **deleteNote:** It will delete single note from SQLite Database
2. **allNotes:** It will fetch all notes from SQLite Database
3. **insertNote:** It will insert single note in SQLite Database.

```
class DatabaseHelper(context: Context?) :
    SQLiteOpenHelper(context, "notes_db", null, 1) {
    override fun onCreate(db: SQLiteDatabase) {
        db.execSQL(NotesData.CREATE_TABLE)
    }
    override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) {
        db.execSQL("DROP TABLE IF EXISTS " + NotesData.TABLE_NAME)
        onCreate(db)
    }
    fun insertNote(note: Notes): Long {
        // Write Code Here
        // return newly inserted row id
        return id
    }
    private fun getValues(note: Notes): ContentValues{
        val values = ContentValues()
        // `id` will be inserted automatically.
        values.put(NotesData.COLUMN_NOTE_TITLE, note.title)
        values.put(NotesData.COLUMN_NOTE_SUB_TITLE, note.subTitle)
        values.put(NotesData.COLUMN_NOTE_DESCRIPTION, note.Description)
        values.put(NotesData.COLUMN_NOTE_REMINDER_TIME, note.remindertime)
        values.put(NotesData.COLUMN_NOTE_SET_REMINDER, note.isReminder)
        values.put(NotesData.COLUMN_TIMESTAMP, note.modifiedTime)
        return values
    }
    fun getNote(id: Long): Notes {
        val db = this.readableDatabase
        val cursor = db.query(
            NotesData.TABLE_NAME,
            arrayOf(NotesData.COLUMN_ID, NotesData.COLUMN_NOTE_TITLE,
                NotesData.COLUMN_NOTE_SUB_TITLE, NotesData.COLUMN_NOTE_DESCRIPTION,
                NotesData.COLUMN_NOTE_SET_REMINDER,
                NotesData.COLUMN_NOTE_REMINDER_TIME, NotesData.COLUMN_TIMESTAMP),
            NotesData.COLUMN_ID.toString() + "=?",
            arrayOf(id.toString()), null, null, null, null
        )
        cursor?.moveToFirst()
        val note = getNote(cursor)
        cursor!!.close()
        return note
    }
    private fun getNote(cursor: Cursor): Notes
    {
        val note = Notes(
            cursor.getString(cursor.getColumnIndex(NotesData.COLUMN_NOTE_TITLE)),
            cursor.getString(cursor.getColumnIndex(NotesData.COLUMN_NOTE_SUB_TITLE)),
            cursor.getString(cursor.getColumnIndex(NotesData.COLUMN_NOTE_DESCRIPTION)),
            cursor.getString(cursor.getColumnIndex(NotesData.COLUMN_TIMESTAMP))
        )
        note.isReminder =
        (cursor.getInt(cursor.getColumnIndex(NotesData.COLUMN_NOTE_SET_REMINDER))==1)
        note.remindertime =
        cursor.getLong(cursor.getColumnIndex(NotesData.COLUMN_NOTE_REMINDER_TIME))
        note.id = cursor!!.getInt(cursor.getColumnIndex(NotesData.COLUMN_ID))
        return note
    }
    val allNotes: ArrayList<Notes>
    get() {
        val notes = ArrayList<Notes>()
        // Write Code Here
        return notes
    }
    fun deleteNote(note: Notes) {
        // Write Code Here
    }
}
```

Q-2 What is the difference between SQLite & SharedPreferences? Write down code to fetch data by using SharedPreferences (write code in getPrefValue method) and store data by using SharedPreferences (write code in setPrefValue method) by using getDefaultSharedPreferences as it is shown in following code: **5**

```
class PrefUtil {
    companion object{
        fun getPrefValue(context: Context, key: String):String?{
            // Write Code Here
        }

        fun setPrefValue(context: Context, key: String, value: String)
        {
            // Write Code Here
        }
    }
}
```

Q-3 Write down xml file of *contact_item.xml* to build UI as given in image by using LinearLayout only. To Display All Contacts, ContactListAdapter is used. Write down code of method getView() of ContactListAdapter to display Contact inside ListView with the help of *contact_item.xml* file. (Contact class is also available in code). **5**

```
class Contact {
    var Name: String
    var EmailId: String
    var PhoneNo: String
    var Address: String
}

class ContactListAdapter(private val mContext: Context,
                        private val dataSet: ArrayList<Contact>) : BaseAdapter() {
    private class ViewHolder {
        lateinit var txtName: TextView
        lateinit var txtEmail: TextView
        lateinit var txtPhone: TextView
        lateinit var txtAddress: TextView
    }

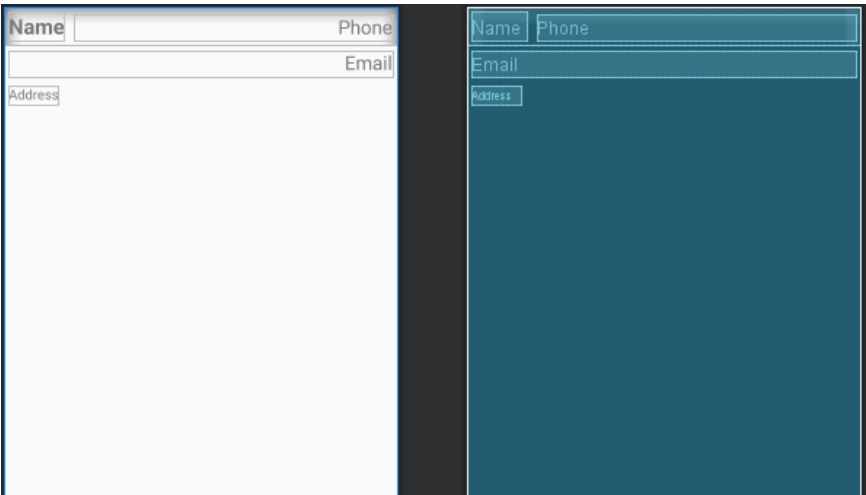
    override fun getCount(): Int {
        return dataSet.size
    }

    override fun getItem(position: Int): Contact {
        return dataSet[position]
    }

    override fun getItemId(position: Int): Long {
        return 0
    }

    override fun getView(position: Int, convertView: View, parent: ViewGroup): View {
        //Write down your code Here
    }
}
```

contact_item.xml layout File



Q-4 Write down code of SMSBroadcastReceiver to receive SMS in Android 6.0 or Above. Assume that editMessage & editPhoneNo is object of EditText where editMessage is used to write SMS message & editPhoneNo is used to write Phone Number to send SMS So write down code of send SMS by using editMessage & editPhoneNo. Write down code of SMS permissions & BroadcastReceiver inside manifest file. (Write down only code of SMS permissions & BroadcastReceiver). **5**

Q-5 Where JSON can be used? Assume that JSON data as shown in image is available to you. Create Employee Class which is used to store data. Create ParseJsonData() method to Parse JSON data and that method returns array of Employee object according to JSON data. **5**

```
[
  {
    "_id": "5f8d677c68d8ae7ceab6a732",
    "name": {
      "first": "Lloyd",
      "last": "York"
    },
    "email": "lloyd.york@undefined.net",
    "phone": "+1 (817) 545-3660",
    "address": "311 Livonia Avenue, Belva, Ohio, 6019"
  },
  {
    "_id": "5f8d677c09133e2398d0c710",
    "name": {
      "first": "Kendra",
      "last": "Blankenship"
    },
    "email": "kendra.blankenship@undefined.info",
    "phone": "+1 (973) 457-2386",
    "address": "782 Cox Place, Southmont, New Jersey, 7806"
  }
]
```

Q-6 Draw & compare Activity Lifecycle in Android with ViewController Lifecycle in iOS. **5**

----------*