7/1/2019

18.30 - 20.30pm

Basement 1, Kevin Street

DUBLIN INSTITUTE OF TECHNOLOGY

DT228 BSc. (Honours) Degree in Computer Science

Year 3

WINTER EXAMINATIONS 2018/19

MOBILE SOFTWARE DEVELOPMENT [CMPU3026]

DR. SUSAN MCKEEVER
DR. DEIRDRE LILLIS
MR PATRICK CLARKE

MONDAY 7TH JANUARY 6:30 P.M. – 8:30 P.M.

DURATION: 2 HOURS

INSTRUCTIONS TO CANDIDATES

QUESTION 1 IS COMPULSORY.

ANSWER QUESTION 1 AND TWO OF THE REMAINING THREE QUESTIONS

QUESTION 1 CARRIES 50 MARKS. ALL OTHER QUESTIONS CARRY 25 MARKS EACH.

Q1. **(a)** Write the XML code for the *row layout* of the list shown in **Figure 1 List of Order Items**. State any assumptions that you make.

(10 marks)

(b) The Android app to populate the list in Figure 1 has a local SQLlite database that contains order items on a dB table. Describe *in your own words* what Java classes and XML layouts the developer will need to create in order to implement the list shown on this screen, including the retrieval of data for the screen. As part of your answer, explain any need for class inheritance.

(10 marks)

Figure 1 List of Order Items

(c) Explain the *purpose* and *use* of Activity lifecycle methods. Use the example of the onCreate() method to explain their use.

(10 marks)

(d) Looking at **Figure 2 Code Sample**, answer, with explanation, the following:

(i) What does the `static` keyword mean as used on lines 017 to 020? (3 marks)
(ii) What superclass is used in the code? (3 marks)
(iii) What will trigger the onUpgrade() method to run? (3 marks)
(iv) What class does the execSQL() method on line 034 belong to (3 marks)
(v) What does the `context` parameter in line 025 mean ? (3 marks)
(vi) What will trigger the onCreate() method to run? (3 marks)
(vii) What does the keyword "`this`" refer to on line 040 (2 marks)

(20 marks)

(Q1 total 25 marks)

```
012 public class JCGSQLiteHelper extends SQLiteOpenHelper {
013
014     // database version
015     private static final int database_VERSION = 1;
016     // database name
017     private static final String database_NAME = "BookDB";
018     private static final String table_BOOKS = "books";
019     private static final String book_ID = "id";
020     private static final String book_TITLE = "title";
021     private static final String book_AUTHOR = "author";
022
023     private static final String[] COLUMNS = { book_ID, book_TITLE, book_AUTHOR };
024
025     public JCGSQLiteHelper(Context context) {
026         super(context, database_NAME, null, database_VERSION);
027     }
028
029    public void onCreate(SQLiteDatabase db) {
030       String CREATE_BOOK_TABLE = "CREATE TABLE books (" + "id +
032       "INTEGER PRIMARY KEY AUTOINCREMENT, " + "title TEXT,"+ " + "author
033       "author TEXT )";
034     db.execSQL(CREATE_BOOK_TABLE);
035
036     @Override
037     public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
038         // drop books table if already exists
039         db.execSQL("DROP TABLE IF EXISTS books");
040         this.onCreate(db);
041     }
```

Figure 2 Code sample

Q2. **(a)** In location tracking, explain how the *minTime* and *minDistance* attributes can impact power consumption and accuracy.

(10 marks)

(b) Answer the following questions about the code shown in **Figure 3 Location Code:**

(i) What is the `LocationListener`? (2 marks)
(ii) Why is the `onStatusChanged` method included, without any code? (3 marks)
(iii) What is the concept of "Criteria" as used in the code? (2 marks)
(iv) What does `R.id.TextView02` refer to? (2 marks)
(v) Explain the purpose of the `onPause()` method as implemented here? (2 marks)
(vi) What method is called by `super.onResume()`? (2 marks)
(vii) Pick out an example of casting below and explain why is it needed. (2 marks)

(15 marks)

(Q2 total 25 marks)

```
public class ShowLocationActivity extends Activity implements
LocationListener {
    private TextView latituteField;
    private TextView longitudeField;
    private LocationManager locationManager;
    private String provider;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        latituteField = (TextView) findViewById(R.id.TextView02);
        longitudeField = (TextView) findViewById(R.id.TextView04);

        locationManager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
        Criteria criteria = new Criteria();
        provider = locationManager.getBestProvider(criteria, false);
        Location location =
                locationManager.getLastKnownLocation(provider);

        if (location != null) {
            System.out.println("Provider " + provider + " selected.");
            onLocationChanged(location);
        } else {
            latituteField.setText("Location not available");
            longitudeField.setText("Location not available");
        }
    }
    @Override
    protected void onResume() {
```

```
        super.onResume();
        locationManager.requestLocationUpdates(provider, 400, 1, this);
    }
    @Override
    protected void onPause() {
        super.onPause();
        locationManager.removeUpdates(this);
    }
    @Override
    public void onLocationChanged(Location location) {
        int lat = (int) (location.getLatitude());
        int lng = (int) (location.getLongitude());
        latituteField.setText(String.valueOf(lat));
        longitudeField.setText(String.valueOf(lng));
    }
    @Override
    public void onStatusChanged(String provider, int status,
Bundle extras)
    {
        // TODO Auto-generated method stub
    }
    @Override
    public void onProviderEnabled(String provider) {
        Toast.makeText(this, "Enabled new provider " + provider,
                Toast.LENGTH_SHORT).show();

    }
    @Override
    public void onProviderDisabled(String provider) {
        Toast.makeText(this, "Disabled provider " + provider,
                Toast.LENGTH_SHORT).show();
    }
}
```

Figure 3 Location Code

Q3. **(a)** Explain the purpose of the MVC software architecture. How compliant is Android with this architecture?

(10 marks)

(b) Explain clearly how to implement *asynchronous processing* in Android so that short repetitive tasks such as network connections, file downloading or database connections can be executed as background tasks.

(10 marks)

(c) Explain the purpose of the *manifest file* in Android. Include three examples of its use.

(5 marks)

(Q3 total 25 marks)

Q4. **(a)** Explain the purpose of *interfaces* in Java (where an interface is implemented in a class using the "`implements`" keyword).

(5 marks)

(b) Explain *two* ways that interfaces such as the `View.OnClickListener` interface can be used to implement event programming in Android (such as responding to button clicks). Clearly explain any advantages or drawbacks of the approaches you describe.

(10 marks)

(c) Explain in your own words how to implement *switching from one screen* to another in Android – including how to bring data from the current screen to the new screen.

(5 marks)

(d) Describe two examples of where *nested classes* have been used in Android development during the course. Explain why nested classes were appropriate in these cases.

(5 marks)

(Q4 total 25 marks)