Answers

Question 1:

A)

a)

```
public void printSensors(){
   SensorManager mgr = (SensorManager) getSystemService(SENSOR_SERVICE);
   final List<Sensor> sensorList = mgr.getSensorList(Sensor.TYPE_ALL);

   final ArrayList<String> sensorNames = new ArrayList<String>();
   for (Sensor sensor : sensorList) {
       sensorNames.add(sensor.getName());
       System.out.println(sensor.getName());
   }
}
```

b)

```
public float getValueRange(int sensorId){
   Sensor mSensor = null;
   SensorManager mgr = (SensorManager) getSystemService(SENSOR_SERVICE);

   if (mgr.getDefaultSensor(Sensor.sensorId) != null){
     mSensor = mgr.getDefaultSensor(Sensor.sensorId);
     return mSensor.getMaximumRange();
   }
   else{
     throw new IllegalArgumentException("Sensor not available");
   }
}
```

c)

```
public void accelMaxRate(){
   mSensorManager = (SensorManager)getSystemService(SENSOR_SERVICE);
         mAccelerometer =
mSensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
         mSensorManager.registerListener(this, mAccelerometer,
SensorManager.SENSOR_DELAY_FASTEST);
   }
```

B)

A problem may occur when logging the values. The `onSensorChanged(SensorEvent event)` function is called all the time because the the values of the accelerometer change all the time. Thus when

the `log(int type, float[] values)` is called (which takes some time to complete) we could run into issues with things like the order in which the events are logged or some logs might get lost. Also because the `log` function takes some time and is called at every change of sensor values (especially accelerometer values) the phone could get very slow.

Question 2:

The state "Running" uses the callback function `onResume()`

The state "Paused" uses the callback function `onPause()`

The state "Stopped" uses the callback function `onStop()`

Question 3:

They should be defined in `res/values/strings.xml` The advantage is that we can easily swap from one language to another. Say we want to change the App from English to German: We would initially only have a `res/values/strings.xml`. Now we create another file `res/values/de_strings.xml` where all the translated strings are stored. We can now simply use the `de_strings.xml` in the entire app instead of having to change each string value manually.

Question 4:

From the android documentation:

> An Intent provides a facility for performing late runtime binding between the code in different applications. Its most significant use is in the launching of activities, where it can be thought of as the glue between activities. It is basically a passive data structure holding an abstract description of an action to be performed.
>
> --*Google and Open Handset Alliance n.d. Android API Guide. https://developer.android.com/reference/android/content/Intent.html. Accessed May Oct 6, 2017.*

Explicit intents are used to launch a new activity in an app and may send some data to the new activity

Implicit intents are used when we want to send data or switch to an inbuilt android activity. For that we must exactly specify which inbuilt activity we want to call. Examples for inbuilt activities are:

- Browser
- Camera
- Contact
- Dialpad

Question 5:

a) false

b) true

c) true

d) false

Question 6:

missing service entry:

```
<service  android:name=".LocationService."  android:exported="false"/>
```

missing location permission:

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

missing send sms permission

```
<uses-permission android:name="android.permission.SEND_SMS"/>
```