1. Come up with some use cases in which a content provider would be helpful.

- Access to a central repository of data
- When we required to share data across applications
- Content providers let you centralize content in one place and have many different applications access it as needed.
- A content provider behaves very much like a database where you can query it, edit its content, as well as add or delete content using insert(), update(), delete(), and query() methods

2. How does file streams work in JAVA?

The goal of InputStream and OutputStream is to abstract different ways to input and output: whether the stream is a file, a web page, or the screen shouldn't matter. All that matters is that you receive information from the stream (or send information into that stream.)

InputStream is used for many things that you read from.

OutputStream is used for many things that you write to.

How work for write

```java
File file = new File("C:/text.bin");

file.createNewFile();

DataOutputStream stream = new DataOutputStream(new FileOutputStream(file));

stream.writeBoolean(true);

stream.writeInt(1234);

stream.close();
```

How work for read

```java
File file = new File("C:/text.bin");

DataInputStream stream = new DataInputStream(new FileInputStream(file));

 boolean isTrue = stream.readBoolean();

int value = stream.readInt();

stream.close();

System.out.printlin(isTrue + " " + value);
```

3. Explain the process of implementing a content provider, and to get the info from a content provider.

How to implement

1. Create a class that extends ContentProvider.
2. Create a contract class.
3. Create the UriMatcher definition.
4. Implement the onCreate() method.
5. Implement the getType() method.
6. Implement the CRUD methods.
7. Add the content provider to your AndroidManifest.xml.

How Access to the data

An application accesses the data from a content provider with a ContentResolver client object. This object has methods that call identically-named methods in the provider object, an instance of one of the concrete subclasses of ContentProvider. The ContentResolver methods provide the basic "CRUD" (create, retrieve, update, and delete) functions of persistent storage.

The ContentResolver object in the client application's process and the ContentProvider object in the application that owns the provider automatically handle inter-process communication. ContentProvider also acts as an abstraction layer between its repository of data and the external appearance of data as tables.

4. What is a vector drawable and how do we implement and use them in android?

What is?

It's a vector graphic defined in an XML file as a set of points, lines, and curves along with its associated color information. The major advantage of using a vector drawable is image scalability. It can be scaled without loss of display quality, which means the same file is resized for different screen densities without loss of image quality. This results in smaller APK files and less developer maintenance. You can also use vector images for animation by using multiple XML files instead of multiple images for each display resolution

Implement and use

Case: renders an image of a battery in the charging mode.

```xml
<!-- res/drawable/battery_charging.xml -->
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    <!-- intrinsic size of the drawable -->
    android:height="24dp"
    android:width="24dp"
    <!-- size of the virtual canvas -->
    android:viewportWidth="24.0"
    android:viewportHeight="24.0">
  <group
        android:name="rotationGroup"
        android:pivotX="10.0"
        android:pivotY="10.0"
        android:rotation="15.0" >
    <path
        android:name="vect"
        android:fillColor="#FF000000"
        android:pathData="M15.67,4H14V2h-4v2H8.33C7.6,4 7,4.6 7,5.33V9h4.93L13,7v2h4V5.33C17,4.6 16.4,4 15.67,4z"
        android:fillAlpha=".3"/>
```

```xml
        <path
            android:name="draw"
            android:fillColor="#FF000000"
            android:pathData="M13,12.5h2L11,20v-5.5H9L11.93,9H7v11.67C7,21.4
7.6,22 8.33,22h7.33c0.74,0 1.34,-0.6 1.34,-1.33V9h-4v3.5z"/>
    </group>
</vector>
```

5. Define the following:

Content Resolver

The Content Resolver is the single, global instance in your application that provides access to your (and other applications') content providers

Primary Key (Sql)

It is a field in a table which uniquely identifies each row/record in a database table. Primary keys must contain unique values. A primary key column cannot have NULL values. A table can have only one primary key, which may consist of single or multiple fields.

Foreign Key (Sql)

It is a column (or columns) that references a column (most often the primary key) of another table. The purpose of the foreign key is to ensure referential integrity of the data. In other words, only values that are supposed to appear in the database are permitted.

Relational Database

It is a set of formally described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables. The standard user and application programming interface (API) of a relational database is the Structured Query Language (SQL).

Dangerous Permissions

They are permissions which could potentially affect the user's privacy or the device's operation. The user must explicitly agree to grant those permissions. These include accessing the camera, contacts, location, microphone, sensors, SMS, and storage.

6. What is a ORM (Object Relational Mapper)?

It is Object Relational Mapping, which is to map an object with a relational world. It basically converts data between incompatible types in object oriented programming languages. What ORM does is wrap the implementation specific details of storage drivers in an API, and maps the relational fields/attributes to an object members. For eg if I have a table Address, it is mapped to a single object Address, with various methods associated with it.

7. Explain how you would upgrade a Table in your database with a new column while preserving the data already in said table.

I'll create a condition with IF, asking is the Data base below to the old version, get the old data, saved in a auxList of the object and insert in an Object with the new Model and addid empty in the string of my new column and later I just sent to the new DataBase .