# Теоретическая часть

* Отображение данных в многостраничном режиме при помощи компонента **ViewPager2**
* Cell-анимация в Android
* Tween-анимация элементов интерфейса в Android

# Практическая часть

**Задача 1.** Разработайте Android -приложение для работы с существующей базой данных. Реализуйте репозитории для таблиц базы данных. Параметры запросов вводите в пользовательских диалоговых окнах на базе фрагментов.

|  |
| --- |
| *База данных****«Платный прием в поликлинике»*** |
| **Описание предметной области**  Платный прием пациентов (консультации специалистов) проводится врачами разных специальностей (хирург, терапевт, кардиолог, офтальмолог и т.д.). Несколько врачей могут иметь одну и ту же специальность. При оформлении приема должна быть сформирована квитанция об оплате приема, в которой указывается информация о пациенте, о враче, который консультирует пациента, о стоимости приема, о дате приема.  Пациент оплачивает за прием некоторую сумму, которая устанавливается персонально для каждого врача. За каждый прием врачу отчисляется фиксированный процент от стоимости приема. Процент отчисления от стоимости приема на зарплату врача также устанавливается персонально для каждого врача.  Размер начисляемой врачу заработной платы за каждый прием вычисляется по формуле: **Стоимость приема \* Процент отчисления от стоимости приема на зарплату врача**. Из этой суммы вычитается подоходный налог, составляющий 13% от суммы. |
| ***База данных должна включать таблицы ВРАЧИ, ПАЦИЕНТЫ, ПРИЕМ, содержащие следующую информацию:*** |
| Фамилия врача |
| Имя врача |
| Отчество врача |
| Специальность врача |
| Стоимость приема |
| Процент отчисления от стоимости приема на зарплату врача |
| Фамилия пациента |
| Имя пациента |
| Отчество пациента |
| Дата рождения пациента |
| Адрес пациента |
| Дата приема |

Кроме исходных базовых таблиц база данных должна содержать следующие объекты:

|  |  |  |
| --- | --- | --- |
| **ЗАПРОСЫ** | | |
| **Номер запроса** | **Тип запроса** | **Какую задачу решает запрос** |
| 1 | Запрос к данным | Выбирает информацию о пациентах с фамилиями, начинающимися на заданную параметром последовательность букв |
| 2 | Запрос к данным | Выбирает информацию о врачах, для которых значение в поле **Процент отчисления на зарплату**, больше 2.3% (задавать параметром) |
| 3 | Запрос к данным | Выбирает информацию о приемах за некоторый период, заданный параметрами |
| 4 | Запрос к данным | Выбирает информацию о докторах, специальность которых задана параметром |
| 5 | Запрос к данным | Вычисляет размер заработной платы врача за каждый прием. Включает поля **Фамилия врача, Имя врача, Отчество врача, Специальность врача, Стоимость приема, Зарплата**. Сортировка по полю **Специальность врача** |
|  |  |  |
| 6 | Итоговый запрос | Выполняет группировку по полю **Дата приема**. Для каждой даты вычисляет максимальную стоимость приема |
| 7 | Итоговый запрос | Выполняет группировку по полю **Специальность**. Для каждой специальности вычисляет средний **Процент отчисления** на зарплату от стоимости приема |

Разместите на активности запросов в горизонтальной ориентации экрана два фрагмента – в первом выводите список запросов, по клику на запрос во втором фрагменте выводите результаты запроса.

При вертикальной ориентации экрана выводить список запросов во фрагменте на одной активности, результаты запроса – во фрагменте на другой активности.

Каждый запрос выполняйте в асинхронной задаче, на базе класса **AsyncTask**.

*Выводите в отдельной активности с использованием* ***ViewPager2*** *все таблицы.* Для таблицы ПРИЕМ предусмотрите возможность добавления новой записи (поля записи вводите в пользовательском диалоговом окне на базе фрагментов), изменения записи (поля записи вводите в пользовательском диалоговом окне на базе фрагментов), удаления записи.

Реализуйте выгрузку таблиц ВРАЧИ, ПАЦИЕНТЫ, ПРИЕМ в файлы формата JSON. Для таблицы ПРИЕМ реализуйте загрузку из файла формат JSON. В файл таблицы ПРИЕМ выгружайте числовые значения внешних ключей*.*

Выгрузка всех таблиц должна выполняться в асинхронной задаче, загрузка таблицы ПРИЕМ также должна выполняться в асинхронной задаче.

В отдельной активности выводите в элемент **WebView** из HTML-файла сведения о разработчике и ненумерованный список – пункты теоретической части задания.

**Задача 2.** В приложение задачи 1 добавьте активность для получения данных из веб-сервиса [**https://jsonplaceholder.typicode.com/**](https://jsonplaceholder.typicode.com/). Используйте меню активности для запуска запросов. Запросы для реализации:

* **GET-запрос** [**https://jsonplaceholder.typicode.com/**](https://jsonplaceholder.typicode.com/)**albums/** – выводить информацию обо всех альбомах, использовать адаптер
* **GET-запрос** [**https://jsonplaceholder.typicode.com/**](https://jsonplaceholder.typicode.com/)**albums/id/** – выводить информацию об альбоме по заданному числовому идентификатору (идентификатор вводите в диалоге)
* ***GET-запрос*** [***https://jsonplaceholder.typicode.com/***](https://jsonplaceholder.typicode.com/)***comments/*** *– выводить все комментарии, использовать адаптер*
* ***GET-запрос*** [***https://jsonplaceholder.typicode.com/***](https://jsonplaceholder.typicode.com/)***posts/*** *– выводить все посты, использовать адаптер*

Альбомы, комментарии и посты выводить при помощи **ViewPager2**, в отдельные страницы.

# Дополнительно

Материалы занятия – в этом же архиве. Запись занятия можно скачать [**по этой ссылке**](https://cloud.mail.ru/public/jHKS/hBxch9bzu).