



IT ШКОЛА SAMSUNG

# Индивидуальный проект Star English – Изучение английского языка

Город: Новосибирск

Площадка: МБОУ «Инженерный лицей НГТУ»

Учащийся: Пономарев Вячеслав

Преподаватель: Томилов И.Н.

Дата: 12 мая 2015г.



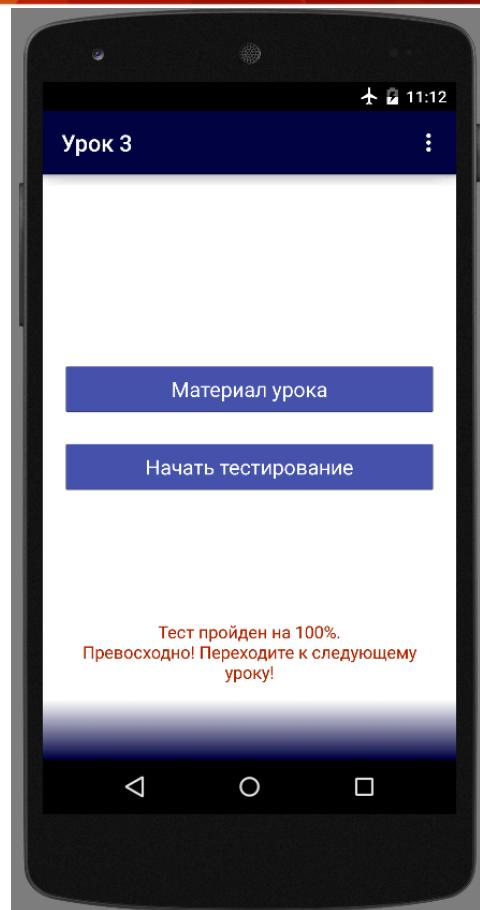
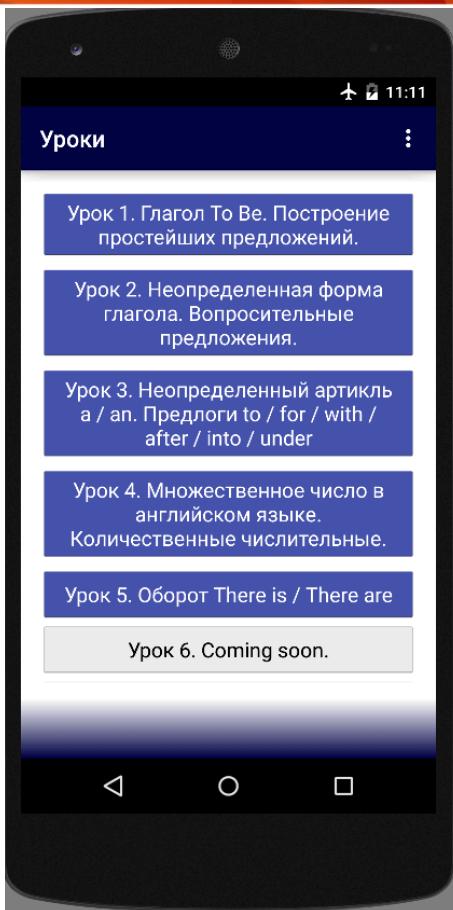
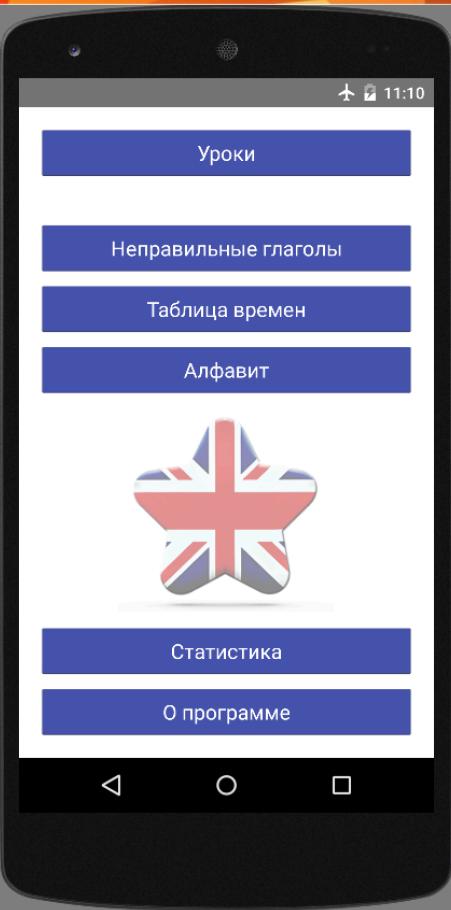
# Описание программы

- Программа **Star English** предназначена для изучения английского языка и закрепления полученных знаний.
- Курс обучения - модульный, в каждом модуле (уроке) представлены обучающие материалы, а также тестирующая система.
- Тест по каждому уроку состоит из 10 вопросов, которые выбираются случайным образом из базы, предоставляющей 50 уникальных вопросов на каждый тест.
- В каждом тесте имеется 4 типа вопросов: вопросы с ответом "да" или "нет", вопросы с ответом, который нужно написать самому, вопросы на сопоставление русских слов и картинок с их английскими аналогами.
- Чтобы перейти к следующему уроку, нужно завершить предыдущий, то есть пройти тест, как минимум на 60%. Первый урок доступен изначально.
- Свою статистику можно посмотреть и сбросить в окне "статистика", доступном с главного экрана.



IT ШКОЛА SAMSUNG

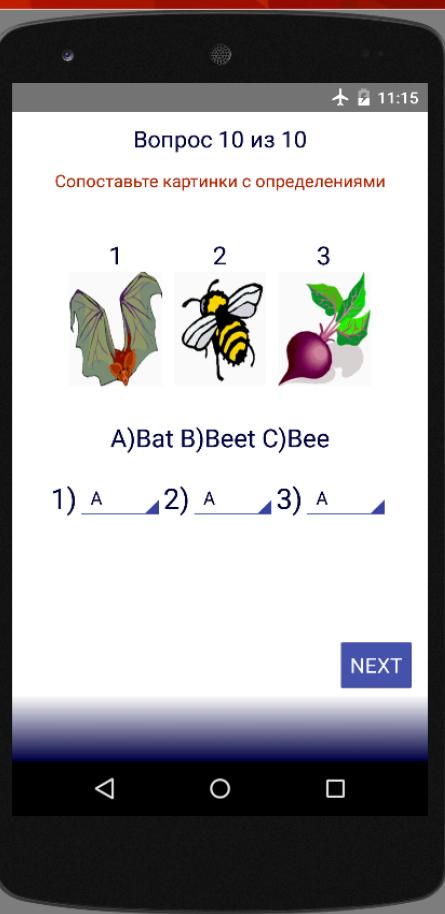
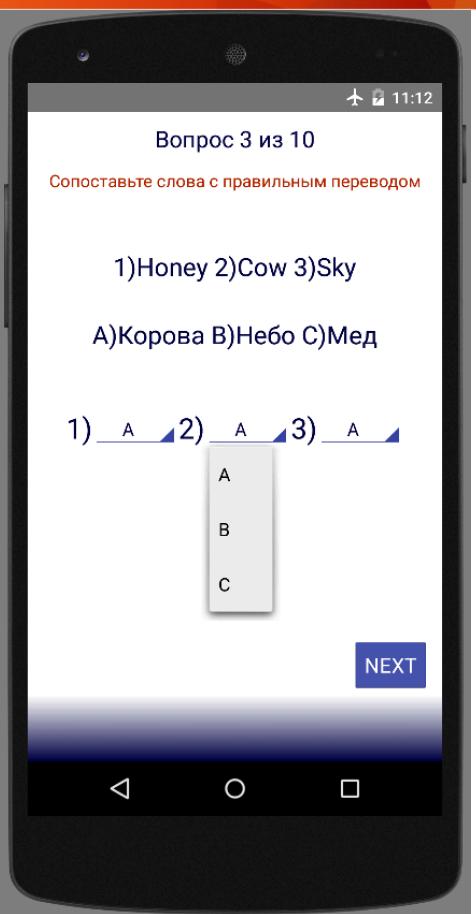
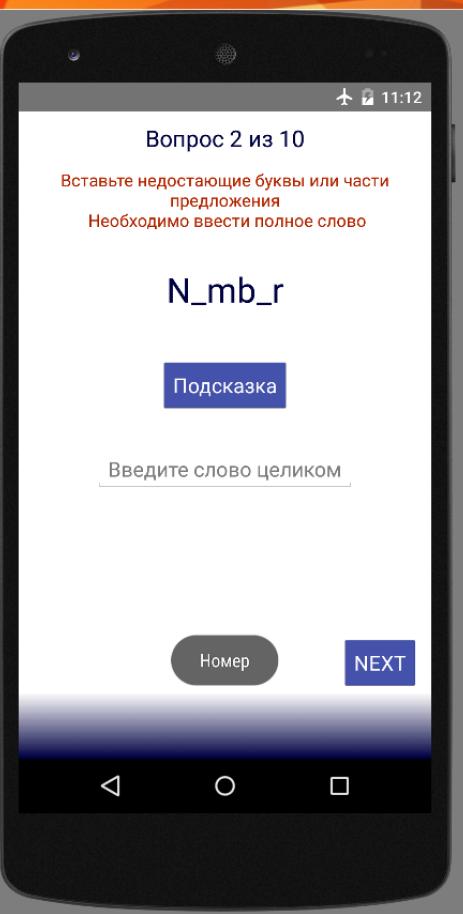
# Скриншоты





IT ШКОЛА SAMSUNG

# Скриншоты





# Скриншоты

Вопрос 3 из 10  
Сопоставьте слова с правильным переводом

1) Honey 2) Cow 3) Sky

A) Корова B) Небо C) Мед

Ваш ответ некорректный  
В этом вопросе не может быть два одинаковых ответа

OK

NEXT

Хорошо!

Вы правильно ответили на 7 вопросов из 10.  
Процент выполнения: 70%

Подробная статистика по каждому заданию:

Вопрос № 7	ПРАВИЛЬНО!
Вопрос № 8	ПРАВИЛЬНО!
Вопрос № 9	НЕВЕРНЫЙ ОТВЕТ!
Понятия на английском и перевод:	
1) Hot 2) Sweet 3) Cold	
A) Горячий B) Холодный C) Сладкий	
Ваши ответы:	
1) A 2) B 3) C	
Правильные ответы:	
1) A 2) C 3) B	
Вопрос № 10	
НЕВЕРНЫЙ ОТВЕТ!	
Предложение или слово на английском:	
Gard_ning	
Ваш ответ:	
GARDINING	
Правильный ответ:	
GARDENING	

Пройти заново Завершить

Статистика

сбросить

Вы успешно завершили 5 уроков из 5

Урок 1  
Лучший результат при прохождении теста:  
90%  
2015-05-10 19:42  
Всего попыток: 1

Урок 2  
Лучший результат при прохождении теста:  
100%  
2015-05-10 19:43  
Всего попыток: 1

Урок 3  
Лучший результат при прохождении теста:  
100%  
2015-05-10 19:45  
Всего попыток: 1

Урок 4  
Лучший результат при прохождении теста:  
100%  
2015-05-10 19:46  
Всего попыток: 1

Урок 5

Урок 5 - Оборот There is / There are  
Pronunciation - Произношение

[ʃ]	church, chair, cherry, chimes, beach, teacher
[ʃ]	ship, bush, negation, notion, sugar, Russian
[θ]	thing, nothing, both, moth, thistle, thigh, third
[ð]	that, mother, brother, with, they, them, bother

Word - Слово

Глагол **let** переводится как **позволять, разрешать**. Он используется во фразах типа **let me, let us**. Если буквально - **позвольте мне, позвольте нам**. Но более точным соответствием будут русские фразы **дайте, давайте, пустяк**.

**Let us** почти всегда сокращается до **let's** [lets].

Let me introduce Mr Bliss - Позовите мне



# Скриншоты

▼ ru.slayff.starenglish

▼ Activity

- С AboutActivity
- С AddInfoActivity
- С LessonPickerActivity
- С LessonShowerActivity
- С MainActivity
- С SplashActivity
- С StatisticsActivity
- С TestActivity

▼ QuizSystem

- С DataBaseHelper
- С QHolder
- С QTypeCOR
- С QTypeMATCH
- С QTypePIC
- С QTypeYN
- С Question
- С Session

## Общая структура

- В проекте использовано 8 активностей, создано 8 собственных классов, обеспечивающих работу тестирующей системы
- В **assets** размещены папки с картинками для уроков, **.html** файлы и БД.

▼ assets

- StarLesson1Pictures
- StarLesson2Pictures
- StarLesson3Pictures
- StarLesson4Pictures
- StarLesson5Pictures
- Alphabet.html
- Les\_database.db
- Lesson1.html
- Lesson2.html
- Lesson3.html
- Lesson4.html
- Lesson5.html
- TensesTable.html
- VerbsTable.html

- Всего в проекте ≈ **2170 строк** (без учета разметок, стилей и т.д.)
- Общий объем своего кода = **98%**. Были заимствованы несколько небольших механизмов.

## Справочные материалы

- Используется универсальная активность с **WebView**, который отображает нужный **html**-файл по идентификатору в **Intent**
- Все уроки и дополнительная информация сохранена в виде **.html** в каталоге **assets**



```
public class AddInfoActivity extends ActionBarActivity {
    int keyInfo; // getting extra from intent to know which .html should be opened
    final String EXTRA_ADD_INFO_KEY = "KEY";
    //Key in Intent to show which .html info-file should be opened
    //Negative values mean info about lesson should be opened
    //Positive values mean additional info should be opened
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_add_info);
        WebView webViewAddInfo = (WebView) findViewById(R.id.webViewAddInfo);
        keyInfo = (int) getIntent().getSerializableExtra(EXTRA_ADD_INFO_KEY);
        switch (keyInfo) {
            case 1 : webViewAddInfo.loadUrl("file:///android_asset/VerbsTable.html"); break;
            case 2 : webViewAddInfo.loadUrl("file:///android_asset/TensesTable.html"); break;
            case 3: webViewAddInfo.loadUrl("file:///android_asset/Alphabet.html"); break;
            case -1: webViewAddInfo.loadUrl("file:///android_asset/Lesson1.html"); break;
            case -2: webViewAddInfo.loadUrl("file:///android_asset/Lesson2.html"); break;
            case -3: webViewAddInfo.loadUrl("file:///android_asset/Lesson3.html"); break;
            case -4: webViewAddInfo.loadUrl("file:///android_asset/Lesson4.html"); break;
            case -5: webViewAddInfo.loadUrl("file:///android_asset/Lesson5.html"); break;
        }
    }
}
```



# Механизмы программы

## Вопросы для тестов

- Для хранения вопросов использована база данных.
- В базе имеется 4 таблицы, под каждый тип вопроса.
- Таблица хранит номер урока для данного вопроса, сам вопрос, варианты ответа (пути к картинкам), ответ(ы).



PICQ											
_id	lesson	path_pic1	path_pic2	path_pic3	pic_title	words_ru	ans_1	ans_2	ans_3		
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter		
7	16	1	StarLesson1...	StarLesson1...	StarLesson1...	A)Lion B)Mouse C)... C)Pig	1)Лев 2)Обезьяна ... 3)Свинья ...	A	C	B	
8	17	1	StarLesson1...	StarLesson1...	StarLesson1...	A)Bear B)Frog C)Dog	1)Медведь 2)Лягушка ... 3)Собака ...	A	B	C	
9	18	1	StarLesson1...	StarLesson1...	StarLesson1...	A)Cock B)Chicken C)... C)Duck	1)Цыпленок 2)Петух 3)... 3)Курица ...	B	A	C	
10	19	1	StarLesson1...	StarLesson1...	StarLesson1...	A)Giraffe B)Wolf C)Sn...	1)Жираф 2)Змей 3)Волк	A	C	B	
11	20	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Sheep B)Kangaroo ... C)Lizard	1)Овца 2)Кенгуру ... 3)Лягушка ...	C	A	B	
12	21	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Bear B)Chamaleo...	1)Мурзай 2)Хамелеон...	C	B	A	
13	22	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Pen B)Pencil C)Rainbow	1)Ручка 2)Карандаш ...	A	B	C	
14	23	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Star B)Wind C)Snow	1)Снег 2)Звезда 3)... 3)Снег ...	C	A	B	
15	24	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Coat B)Seals C)...	1)Носки 2)Галстук 3)... 3)Слоны ...	B	A	C	
16	25	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Village B)City C)Sea	1)Море 2)Город 3)... 3)Водопад ...	C	B	A	
17	26	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Potatoe B)Salad C)A...	1)Яблоко 2)Картошка ... 3)Салат ...	C	A	B	
18	27	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Pineapple B)Peach C)...	1)Пицца 2)Пицца 3)... 3)Арбуз ...	C	B	A	
19	28	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Octopus B)Shark C)...	1)Черепаха 2)Осьминог ... 3)Кит ...	C	A	B	
20	29	2	StarLesson2...	StarLesson2...	StarLesson2...	A)Sheep B)Tomato C)...	1)Овечка 2)Клубника ... 3)Томат ...	A	C	B	
21	30	3	StarLesson3...	StarLesson3...	StarLesson3...	A)Bull B)Anchor C)...	1)Якорь 2)Лампочка ... 3)Бык ...	B	A	C	



## Вопросы для тестов

- База создавалась вручную с использованием редактора **SQLiteBrowser** и размещалась в каталоге **assets**.
- Класс  **DataBaseHelper** отвечает за копирование базы из assets в системный каталог хранения БД и последующее ее открытие.
- Так как Android требует, чтобы каждая открытая база была впоследствии закрыта, класс был спроектирован в соответствии с паттерном **синглтона**, что позволило иметь иметь доступ к единственному экземпляру класса в приложении.

```
    public static DataBaseHelper getInstance(Context context) {  
        if (sInstance == null) {  
            sInstance = new DataBaseHelper(context);  
        }  
        return sInstance;  
    }
```

```
public void createDataBase() throws IOException{  
    boolean dbExist = checkDataBase();  
    if (!dbExist) {  
        //nothing to do - database is already exists  
    } else {  
        this.getReadableDatabase();  
        try {  
            copyDataBase();  
        } catch (IOException e) {  
            Log.e("AppLog", "Error copying database");  
            throw new Error("Error copying database");  
        }  
    }  
}
```

## Вопросы для тестов

- Для каждого типа вопроса создан свой класс с методами `get()`, `set(...)`, `isAnswerCorrect(...)`, который наследуется от класса `Question`.
- В атрибутах класса хранится вся информация, которая есть в базе данных по вопросу.

```
public class QTypeCOR extends Question {  
  
    private String type;  
    private String base;  
    private String answer;  
    private String hint;  
  
    public QTypeCOR(Cursor cur) {  
        type = "CORQ";  
        int idOfbase = cur.getColumnIndex("base");  
        int idOfanswer = cur.getColumnIndex("answer");  
        int idOfhint = cur.getColumnIndex("hint");  
        base = cur.getString(idOfbase);  
        answer = cur.getString(idOfanswer);  
        hint = cur.getString(idOfhint);  
    }  
}
```

  QTypeCOR
  QTypeMATCH
  QTypePIC
  QTypeYN
  Question



## Тестирующая система

```
/**  
 *  
 * @return List of random Questions in exact lesson  
 */  
public List<Question> getQList() {  
    List<Integer> idOfYN = getListOfId(typeYN);  
    List<Integer> idOfCOR = getListOfId(typeCOR);  
    List<Integer> idOfMATCH = getListOfId(typeMATCH);  
    List<Integer> idofPIC = getListOfId(typePIC);  
    QList = new ArrayList<Question>();  
    Collections.shuffle(idOfYN);  
    for (int i = 0; i < 3; i++) {  
        int curId = idOfYN.get(i);  
        Cursor cursor = myDBHelper.getCursorAtId(curId, typeYN);  
        cursor.moveToFirst();  
        QTypeYN question = new QTypeYN(cursor);  
        QList.add(question);  
        cursor.close();  
    }  
    Collections.shuffle(idOfCOR);  
    for (int i = 0; i < 3; i++) {  
        int curId = idOfCOR.get(i);  
        Cursor cursor = myDBHelper.getCursorAtId(curId, typeCOR);  
        cursor.moveToFirst();  
        QTypeCOR question = new QTypeCOR(cursor);  
        QList.add(question);  
        cursor.close();  
    }  
    Collections.shuffle(idOfMATCH);  
    for (int i = 0; i < 3; i++) {  
        int curId = idOfMATCH.get(i);  
        Cursor cursor = myDBHelper.getCursorAtId(curId, typeMATCH);  
        cursor.moveToFirst();  
        QTypeMATCH question = new QTypeMATCH(cursor);  
        QList.add(question);  
        cursor.close();  
    }  
    Collections.shuffle(idofPIC);  
    for (int i = 0; i < 2; i++) {  
        int curId = idofPIC.get(i);  
        Cursor cursor = myDBHelper.getCursorAtId(curId, typePIC);  
        cursor.moveToFirst();  
        QTypePIC question = new QTypePIC(cursor);  
        QList.add(question);  
        cursor.close();  
    }  
    return QList;  
}
```

- Система тесно связана с классом **Qholder**, который хранит **ArrayList** **<Question>**.
- Чтобы создать **ArrayList** в метод передается номер урока. Создается **ArrayList** с **\_id** элементов, подходящих по значению номера урока.
- Каждый **ArrayList** перемешивается и из него выбираются 3 или 2 значения, которые помещаются в новый **ArrayList**. (4 типа вопроса, всего 10 значений = 10 вопросов в тесте)
- Новый **ArrayList** перемешивается и метод его возвращает.



## Тестирующая система

- В классе активности **TestActivity** создается счетчик **counter**, который считает количество вопросов, открывается БД, создается **ArrayList<Question>** со случайными вопросами по данному уроку.
- Сам интерфейс активности пустой – программа каждый раз получает вопрос из **ArrayList** и по его типу определяет какой метод нужно вызвать чтобы отобразить нужный интерфейс.
- На каждый клик кнопки **NEXT**, **counter++**. По значению **counter==10** управление передается методу, который показывает результаты теста.

```
public void setViewYN(final Question question, final List<Question> listofQ) {...}

/**
 * setting view if question type is "Correct input"
 */
public void setViewCOR(final Question question, final List<Question> listofQ) {...}

/**
 * setting view if question type is "Match words"
 */
public void setViewMATCH(final Question question, final List<Question> listofQ) {...}

/**
 * setting view if question type is "Match pictures"
 */
public void setViewPIC(final Question question, final List<Question> listofQ) {...}

/**
 * setting view if counter == 10. All the questions are answered, so need to show results.
 */
public void setViewRESULT() {...}
```



## Тестирующая система

- По каждому клику кнопки **NEXT**, программа проверяет ответ на корректность и записывает информацию в **ArrayList** в классе **Session**.
- Если ответ правильный, запишется строка «Верно», если не правильный, то в **ArrayList** будет записан ответ пользователя и правильный ответ.
- В **layout**, который показывает результат, размещен **ListView**, который через адаптер получает данные из **ArrayList** в **Session** и отображает результаты по каждому вопросу.

```
public void userIsCorrect(Question question, int number0fQuestion) {...}

/**
 *
 * inc number of incorrect answers and total answers
 * calling method to put the string with info
 */

public void userIsWrong(Question question, int number0fQuestion, String userAnswer) {...}

/**
 *
 * specifying the exact string to be put into List
 * put string into List
 */
private void putCorrectAnswer(Question question, int number0fQuestion) {...}
```



## Хранение статистики

- Для хранения статистики по всем урокам с лучшими результатами был использован системный класс **SharedPreferences**, который хранит данные постоянно(и после закрытия приложения) в формате «ключ-значение».
- Эта статистика загружается в **StatisticsActivity**, куда можно попасть из любого места приложения по нажатию кнопки меню. Статистику можно сбросить – все значения по всем урокам в классе **SharedPreferences** выставляются по умолчанию.

```
final public static int TOTAL_LESSONS_IN_APP = 5;
//
public static final String APP_STATISTICS = "StarEnglishStatistics"; //Filename of full statistics
SharedPreferences preferences; // object to get access to statistics
private String TOTAL_ATTEMPTS = "TOTAL_ATTEMPTS_";
private String BEST_ATTEMPT = "BEST_ATTEMPT_";
private String BEST_ATTEMPT_TIME = "BEST_ATTEMPT_TIME_";
private String DONE_LESSON = "DONE_LESSON_";
private static final String TOTAL_LESSONS_DONE = "TOTAL_LESSONS_DONE";
//
```



# Механизмы программы

## Небольшой дизайн

- Основные цвета приложения – темно-синий и акцентирующий темно-красный.
- Чтобы дизайн был одинаковым на устройствах с Android 2.3.X, 4.X.X, 5.X были сгенерированы элементы интерфейса с единым дизайном.
- Все элементы использованы при генерации стилей и тем, которые использованы в программе.

```
<style name="_StarThemeBar" parent="Theme.Startoolbar">  
  
    <item name="android:editTextStyle">@style/EditTextStarTheme</item>  
  
    <item name="android:textColorHighlight">#99303f9f</item>  
  
    <item name="android:radioButtonStyle">@style/RadioButtonStarTheme</item>  
  
    <item name="android:buttonStyle">@style/ButtonStarTheme</item>  
  
    <item name="android:imageButtonStyle">@style/ImageButtonStarTheme</item>  
  
    <item name="android:spinnerStyle">@style/SpinnerStarTheme</item>  
  
    <item name="android:spinnerDropDownItemStyle">@style/SpinnerDropDownItemStarTheme</item>  
  
    <item name="android:textViewStyle">@style/TextViewStarTheme</item>  
  
</style>
```



ENGLISH



# Механизмы программы

## Возможности для развития

- Все главные методы независимы от значений параметров.
- В базу данных можно добавлять как вопросы для новых уроков, так и вопросы для уже существующих.
- Чтобы добавить урок в программу нужно дописать  $\approx 5$  строк кода.

---

Урок 6. Coming soon.

Урок 7. Coming soon.

Урок 8. Coming soon.



# Механизмы программы

## Итоги

- На данном этапе приложение закончено.
- В перспективе:
  - Добавление новых уроков
  - Добавление новых типов вопросов
  - Переработка системы тестирования под различную степень сложности  
(нужно добавить колонку “Difficulty” в БД)
  - Добавление многопользовательского режима
  - Более глубокая проработка дизайна. Material...?



IT ШКОЛА SAMSUNG

**Спасибо за внимание!**