

**СЪЗДАВАНЕ НА ПРИЛОЖЕНИЯ ЗА МОБИЛНИ  
ТЕЛЕФОНИ И ТАБЛЕТИ С ОПЕРАЦИОННА СИСТЕМА  
АНДРОИД В ЧАСОВЕТЕ ЗА ИЗВЪНКЛАСНА РАБОТА  
ПО ИНФОРМАТИКА**

**Сюзан Феимова**

Настоящата разработка представя материали за обучението на ученици за създаване на мобилни приложения в часовете за извънкласна работа по информатика. Предложено е примерно тематично разпределение за преподаване. Разгледани са методически аспекти. Направен е увод в мобилното програмиране с език Java в работна среда Eclipse. Дадено е съкратено описание за създаване на контекстно меню в мобилни приложения за Android. Описани са основите за създаване на бази от данни със SQLite.

**Увод.** През последните години пазарът на приложения за мобилни телефони и таблети се развива с бурни темпове. Мобилните приложения носят удобство и подобряват организацията ни, като моделират света, в който живеем. Без значение дали сме у дома, на работа или в движение, мобилните приложения осигуряват достъп до нашите данни непрекъснато. В резултат на това се наблюдава и видима промяна в принципите, по които са изградени потребителските интерфейси. Стават популярни интерфейси, оптимизирани за мобилни устройства, с възможност за разпознаване на докосвания и жестове, подобрени технологии за търсене, управление с глас и широки мултимедийни възможности. Мобилните устройства се развиват и осигуряват все по-сложни и разнообразни откъм възможности и функции услуги за потребителите. Много приложения, които първоначално са били предназначени за десктоп или уеб-среда, сега са се пренесли към мобилните устройства. Развива се „мобилно програмиране“, като много актуално съвременно изследователско направление, допринасящо за усъвършенстване на алгоритмичното мислене. То дава възможност на учениците да видят практическата приложимост на своите знания.

Петгодишният опит на автора в извънкласни занимания с ученици за създаване на приложения за мобилни телефони и таблети с операционна система Андроид показва, че реално така се доразвиват уменията на обучаемите да програмират и се надграждат познанията им по информатика. В бъдеще мобилното програмиране би могло да се добави и като част от задължителната подготовка по информатика.

В доклада е предложен метод за плавно навлизане в мобилното програмиране чрез създаване на базова програма. По-специфични, но изключително значими, са темите „Създаване на контекстни менюта“ и „Бази от данни“. Затова се обръща специално внимание на основни моменти от теоретичното им представяне, като се описва общият вид на необходимите за работа основни команди.

Примерно тематично разпределение (36 седмици по 2 часа = 72 часа)

№	Тема	Вид на урока	Брой часове
1	Същност на мобилното програмиране и възможности за създаване на мобилни приложения.	нови знания	1
2	Инсталиране на среда Eclipse и приставка Android SDK.	нови знания	1
3	Инсталиране на среда Eclipse и приставка Android SDK.	упражнение	1
4	Версиите на Android. Запознаване със средата Eclipse. Работа с проекти - създаване на нов проект, запазване на проект, отваряне на съществуващ проект.	нови знания	1
5	Работа с проекти – създаване на нов проект, запазване на проект, отваряне на съществуващ проект.	упражнение	1
6	Основни компоненти етикет, текстово поле, бутон, радиобутон, поле за избор, поле за изображение – добавяне и използване.	нови знания	1
7	Основни компоненти етикет, текстово поле, бутон, радиобутон, поле за избор, поле за изображение – добавяне и използване.	упражнение	1
8	Свойства на основните компонентите – прозорец Properties, предназначение и стойности на компонентите.	нови знания	1
9	Свойства на основните компонентите – предназначение и стойности.	упражнение	1
10	Стандартни изгледи за разположение на обектите (Layouts – linear, relative, table, grid).	нови знания	1
11	Стандартни изгледи за разположение на обектите (Layouts – linear, relative, table, grid).	упражнение	1
12	Работа с цветове – начини за дефиниране на цветове, създаване и използване на системни цветове.	нови знания	1
13	Работа с цветове – начини за дефиниране на цветове, създаване и използване на системни цветове.	упражнение	1
14	Деклариране и задаване на стилове. Промяна на основната тема чрез добавяне на създадени стилове.	нови знания	1
15	Деклариране и задаване на стилове. Промяна на основната тема чрез добавяне на създадени стилове.	упражнение	1
16	Изграждане и оформление на графичния дизайн на приложението.	нови знания	1
17	Изграждане и оформление на графичния интерфейс на приложението.	упражнение	1
18	Създаване на начален екран на приложение на ниво графичен потребителски интерфейс.	обобщение	1
19	Контролна работа.	проверка на знанията	1
20	Работа върху индивидуални задачи.	упражнение	1
21	Представяне на индивидуалните задачи.	упражнение	1
22	Стартиране на приложението на емулатор. Създаване на инсталационен файл. Инсталиране на приложението на мобилно устройство.	нови знания	1

23	Стартиране на приложението на емулатор. Създаване на инсталационен файл. Инсталиране на приложението на мобилно устройство.	упражнение	1
24	Прехвърляне на елементи от XML в Java обекти.	нови знания	1
25	Прехвърляне на елементи от XML в Java обекти.	упражнение	1
26	Регистриране и обработка на събития чрез методите Event Listeners.	нови знания	1
27	Регистриране и обработка на събития чрез методите Event Listeners.	упражнение	1
28	Компоненти форми (Intents) – основни свойства и събития.	нови знания	1
29	Компоненти форми (Intents) – основни свойства и събития.	упражнение	1
30	Екипна работа върху задачи.	упражнение	1
31	Представяне на задачите от всеки екип.	упражнение	1
32	Добавяне на лента за превъртане с компонент ScrollView – вмъкване и настройване.	нови знания	1
33	Добавяне на лента за превъртане с компонент ScrollView – вмъкване и настройване.	упражнение	1
34	Създаване на функциониращо приложение с няколко форми.	обобщение	1
35	Контролна работа.	проверка на знанията	1
36	Проектиране и реализиране на главно и контекстно меню.	нови знания	1
37	Проектиране и реализиране на главно и контекстно меню.	упражнение	1
38	Компонент поле за действие (Action Bar) – предназначение, създаване, използване.	нови знания	1
39	Компонент поле за действие (Action Bar) – предназначение, създаване, използване.	упражнение	1
40	Допълнителни възможности за диалог с потребителя чрез компоненти диалози.	нови знания	1
41	Допълнителни възможности за диалог с потребителя чрез компоненти диалози.	упражнение	1
42	Работа върху индивидуални задачи.	упражнение	1
43	Представяне на индивидуалните задачи.	упражнение	1
44	Запазване на текуща информация от приложението с компоненти Shared Preferences.	нови знания	1
45	Запазване на текуща информация от приложението с компоненти Shared Preferences.	упражнение	1
46	Създаване и използване на бази от данни. Осъществяване на връзка с база от данни. Достъп до информацията в база от данни, добавяне и изтриване на записи.	нови знания	1
47	Създаване и използване на бази от данни. Осъществяване на връзка с база от данни. Достъп до информацията в база от данни, добавяне и изтриване на записи.	упражнение	1
48	Мобилни приложения с меню, диалози и бази от данни.	обобщение	1
49	Контролна работа.	проверка на знанията	1
50	Екипна работа върху задачи.	упражнение	1
51	Представяне на задачите от всеки екип.	упражнение	1

52	Добавяне на звук и видео.	нови знания	1
53	Добавяне на звук и видео.	упражнение	1
54	Записване на информация във вътрешната и външната памет на мобилното устройство.	нови знания	1
55	Записване на информация във вътрешната и външната памет на мобилното устройство.	упражнение	1
56	Задаване на мрежови HTTP GET и HTTP POST заявки.	нови знания	1
57	Задаване на мрежови HTTP GET и HTTP POST заявки.	упражнение	1
58	Съхраняване на системна информация за работата на приложението с компонент Context.	нови знания	1
59	Съхраняване на системна информация за работата на приложението с компонент Context.	упражнение	1
60	Създаване на списък от записи с компонент ListView. Начини за попълване на списъка от записи с данни.	нови знания	1
61	Създаване на списък от записи с компонент ListView. Начини за попълване на списъка от записи с данни.	упражнение	1
62	Създаване на индивидуални проекти.	упражнение	1
63	Представяне на индивидуалните проекти.	упражнение	1
64	Добавяне на допълнителни услуги, подпомагащи работата на приложенията (services).	нови знания	1
65	Добавяне на допълнителни услуги, подпомагащи работата на приложенията (services).	упражнение	1
66	Методи за определяне на местоположението на мобилното устройство.	нови знания	1
67	Методи за определяне на местоположението на мобилното устройство.	упражнение	1
68	Създаване на завършено и функционално приложение. Инсталиране на мобилно устройство.	обобщение	1
69	Тестване и подобряване на приложението. Публикуване в Андроид маркет.	упражнение	1
70	Контролна работа.	проверка на знанията	1
71	Създаване на проекти чрез работа в екипи.	упражнение	1
72	Представяне и защита на проектите.	упражнение	1
	Общ брой часове:		72

**Цели на обучението.** Основните цели са учениците: 1) да се запознаят със средата за програмиране Eclipse; 2) да усвоят работата с обектите, свойствата и събитията; 3) да научат и свободно да използват основните алгоритмични и структурни конструкции на езика Java, които се използват в мобилното програмиране; 4) да развият и усъвършенстват алгоритмичното, логическото и стратегическото си мислене; 5) да проектират и създават функционални мобилни приложения.

**Методически аспекти.** Когато обучението се провежда в блок от два часа, се прилага програмна схема, при която първият час да бъде упражнение върху новия материал от предния час, а вторият да е за въвеждане на нови знания. Така в края на втория час учителят може да зададе домашна работа, която ученикът да реализира на компютър вкъщи и да бъде проверена и обсъдена в началото на следващия

блок часове. Предложената организационна технология ще доведе до по-ефективна работа в часовете и до създаване на обективни предпоставки за повишаване на качеството на обучение.

Успешното постигане на полезно и пълноценно обучение предполага ученикът да работи с компютър поне 70% от учебното време.

Методиката на преподаване включва лекции, демонстрации, дискусии, изследователски практики, общи и индивидуални задачи, писмен, устен и практически контрол на усвоените знания. Новите понятия и дейности могат да бъдат ефективно усвоени, ако се демонстрират и изследват върху готови, подходящо създадени от учителя модули (примери). Работейки с примерните модули, учениците ще се упражняват със системата за проектиране и ще усвоят използването на езиковите конструкции и работата с компонентите и техните свойства и събития.

Препоръчително е периодично да се задават индивидуални и екипни задачи, които да се представят пред всички, а накрая обучението да завърши със самостоятелна и екипна разработка и защита на проект. За целта следва да бъде отделено време за анализ на задачите и надеждни условия за самостоятелна и екипна работа на учениците в учебните часове, а по възможност и извън тях. При работа по индивидуалните и екипни задачи и проекти на учениците трябва да се даде възможност да покажат своя собствен стил на мислене и подбор на начин за реализация. Тъй като добиването на умения за работа в екип е една съществена задача в обучението, следва да се обръща особено внимание върху планирането и разпределянето на дейностите, както и на спазването на съответните норми на поведение при такъв стил на работа.

**Методи и форми за оценяване на ученика.** Тъй като акцентът се поставя върху практическото усвояване на умения, то превес в оценяването трябва да имат тези форми, които измерват преди всичко степента на практическо овладяване. Контролните работи могат да бъдат практически задачи, които учениците трябва да изпълнят в рамките на текущия час.

Всеки обучаем е личност със свой характерен начин на мислене. Предразположен е към специални за него мисловни процеси, форми и начини на учене и определено има индивидуални предпочитания към някои от тях. Създаването на приложения за мобилни устройства предоставя на обучаваните много добри теоретични и практически възможности да представят своята индивидуалност и специфичния си стил на работа, защото изисква от обучаваните да разработят програмен продукт, като им дава свободата сами да изберат начините и средствата за постигане на желани резултат. Така различията в начина на мислене и уникалните и нестандартни подходи за реализиране на задачите трябва да се възприемат от учителя като богатство и да се стимулират, а не да се санкционират. Основни критерии при оценяването са ефективност (степен на резултатност) и ефикасност (постигане на единица резултат при минимизиране на разходите на ресурси – време, усилия и др.). Запознаването на учениците с тези критерии оказва много силно мотивиращо въздействие върху тях и им помага да придобият навици да оценяват критично собствената си работа и начини на мислене, сравнявайки ги с тези, предложени от учителя или използвани от съучениците им. Това на практика им позволява да усвояват и прилагат ефективни стратегии за мислене, учене и работа.

При работа по проекти се оценява както готовият продукт, така и процесът на

работа. При работа в екип и група се оценяват и уменията за ефективно общуване в рамките на групата, конструктивно поведение и принос с идеи.

**Инсталиране на софтуера.** Софтуерът е безплатен и може да се свали от сайта: [developer.android.com/sdk](http://developer.android.com/sdk). Избира се DOWNLOAD FOR OTHER PLATFORMS и се сваля версията за желаната операционна система. Тук всичко е в един пакет – средата Eclipse с необходимата приставка SDK за създаване на приложения за Android.

**Етапи за създаване на мобилно приложение.** 1. Създаване на графичен дизайн на приложението; 2. Програмиране на функционалността; 3. Тестване на приложението; 4. Одобрение и публикуване в Android Market; 5. Промотиране на приложението.

**Създаване на програма, използваща текст, въведен от потребителя.** Разглежда се: запознаване с работата на Eclipse и улесненията, които носи; започване на нов проект; залагане на първите контроли; запознаване с части от XML и елементите EditText, TextView и Button; настройка на изгледа на приложението; първи резултати от действията.

<p>Създава се нов проект чрез избор на File/New/Android Application Project. В прозореца Package Explorer са работните директории на проекта. В папка layout, намираща се в папка res, е файлът activity_main.xml (така се именува по подразбиране, но може и да се промени името му). Отваря се. Първото нещо, което се вижда е графична среда (Graphical Layout).</p>	<p>Възможно е да се наложи малко изчакване на зареждането ѝ. Това зависи от компютъра, на който се работи. В самия activity_main.xml файл се вижда кодът, който създава този графичен екран.</p> <pre>&lt;RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent" android:layout_height="match_parent" android:paddingBottom="@dimen/activity_vertical_margin" android:paddingLeft="@dimen/activity_horizontal_margin" android:paddingRight="@dimen/activity_horizontal_margin" android:paddingTop="@dimen/activity_vertical_margin" tools:context=".MainActivity" &gt; &lt;TextView android:layout_width="wrap_content" android:layout_height="wrap_content" android:text="@string/hello_world" /&gt; &lt;/RelativeLayout&gt;</pre>
---	---

<p>Има два елемента. Първият е Relative layout, който определя как да бъдат разположени един спрямо друг елементите на екрана. Вторият е компонент TextView, изобразяващ текст, който се вмъква по подразбиране при създаване на проекта. Зададена е стойността на свойство text да бъде низът "hello_world".</p>	<p>Низът е взет от файла strings.xml, намиращ се в папка values, която е подпапка на папка res. Файлът strings.xml има следното съдържание:</p> <pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;resources&gt;&lt;string name="app_name"&gt;Name&lt;/string&gt; &lt;string name="action_settings"&gt;Settings&lt;/string&gt; &lt;string name="hello_world"&gt;Hello world! &lt;/string&gt; &lt;/resources&gt;</pre>
---	--

Параметърът name определя името на символния низ. След това е зададена стойността на низа. Това позволява при необходимост да се изпие даденият текст, да се извика с името му, като се посочи, че е дефиниран във файла string.xml, написвайки @string/ пред името му.

Във файла `activity_main.xml` свойствата `layout_width` и `layout_height` определят съответно ширина и височина. В дадения файл те нямат числови стойности, а стойности `match_parent` и `wrap_content`. `Match_parent` или `fill_parent` означават компонентът да заеме размерите на родителското тяло. Ако има `android:layout_width="match_parent"`, то компонентът `TextView` добива ширината на екрана на разработваното приложение. `Wrap_content` означава, че ширината или височината на компонента зависи от големината на текста.

За да се премахне надписът „Hello World!“ от файла `activity_main.xml` се изтрива:  
`android:text="@string/hello_world"`

Вмъкват се четири компонента – `EditText` (текстово поле), два компонента `TextView` (етикет) и `Button` (бутон). Реализира се следното: в единия етикет се слага надпис: “Вашето име е: “ и до него след натискане на бутона се изписва името, което потребителят е въвел в текстовото поле.

Задават се имена (`id`) на елементите. Чрез тези имена се свързват елементите с променливи (обекти) в `java` кода. Записва се `android:id="@+id/име на елемента"`. Например за компонента `EditText` се пише: `android:id="@+id/ET_put"`. Аналогично се задават имена на всички останали елементи.

Дават се фиксирани размери на елементите. На текстовото поле се задава ширина `200dp`, височина `60dp`. Препоръчително е да се определят размерите в `density pixels`, което се означава с `dp`. Така елементите се оразмеряват в зависимост от екрана на устройството, на което се използва приложението. Добавя се свойство `hint` за текстовото поле. Това свойство задава ушътващ текст, който се вижда първоначално в текстовото поле и изчезва при въвеждане на текст от потребителя. За стойност на `hint` се пише: „Въведете името си“. На бутона се задава необходимата ширина, за да се види текстът в него (`wrap_content`), височина `60dp`, текст “ЗАПИШИ“. Стойности на свойствата на контролите могат да се задават както в програмния код във файла `activity_main.xml`, така и в прозореца `Properties`. Това, което се записва във файла `activity_main.xml`, е:

<pre> &lt;EditText   android:id="@+id/ET_put"   android:layout_width="200dp"   android:layout_height="60dp"   android:hint="Въведете името си" &gt; &lt;/EditText&gt; &lt;TextView   android:id="@+id/TV_viewTekst"   android:layout_width="wrap_content"   android:layout_height="wrap_content"   android:text="Вашето име е: "/&gt; </pre>	<pre> &lt;TextView   android:id="@+id/TV_view"   android:layout_width="wrap_content"   android:layout_height="wrap_content"   android:text= /&gt; &lt;Button   android:id="@+id/B_button"   android:layout_width="wrap_content"   android:layout_height="60dp"   android:text="ЗАПИШИ"   android:textSize="20dp"/&gt; </pre>
--	--

<p>Отваря се файлът MainActivity.java:</p> <pre>package com.feimova.name; import android.os.Bundle; import android.app.Activity; import android.view.Menu; public class MainActivity extends Activity{@Override protected void onCreate(Bundle savedInstanceState) {super.onCreate(savedInstanceState); setContentView(R.layout.activity_main); }@Override public boolean onCreateOptionsMenu(Menu menu) //Inflate the menu;this adds items to the action bar getMenuInflater().inflate(R.menu.main, menu); return true;}}</pre> <p>Декларират се няколко публични променливи (обекти) извън методите, след</p> <pre>public class MainActivity extends Activity {</pre> <p>Създават се обекти за всеки един от елементите. Какъвто е елементът, така се задава и обекта – съответно TextView, Button и EditText. <code>public Button but; public TextView view; public EditText put;</code></p> <p>Дава грешка, защото обектите не са добавени към съответните класове. Това може да се направи лесно, като се кликне върху червеното хиксче, което се появява в началото на реда и след това се избере добавяне (<code>import</code>).</p>	<p>Следва да се насочат обектите към елементите. Това става в метода <code>onCreate</code>. Указва се като какъв елемент ще се използва обектът (например <code>but = (Button)..</code>). За да се насочи към даден елемент записваме <code>findViewById(R.id.id</code> на елемента, който искаме).</p> <p>За обекта <code>but</code> се задава:</p> <pre>but=(Button)findViewById(R.id.B_button);</pre> <p>Аналогично и за другите обекти:</p> <pre>view=(TextView) findViewById(R.id.TV_view); put=(EditText) findViewById(R.id.ET_put);</pre> <p>Трябва като се кликне върху бутона да се изпълни определено действие. Затова е необходимо да се напише:</p> <pre>but.setOnClickListener (newView.OnClickListener() {@Override public void onClick(View v) { //тук записваме това, което искаме да се изпълни при натискане на бутона}});</pre> <p>За да се изведе в етикета текстът от текстовото поле при натискане на бутона, се записва:</p> <pre>view.setText(put.getText());</pre>
---	---

Файлът MainActivity.java вече има следния вид:

<pre>package com.feimova.name; import android.os.Bundle; import android.app.Activity; import android.view.Menu; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.TextView; public class MainActivity extends Activity { public Button but; public TextView view; public EditText put; @Override protected void onCreate(Bundle savedInstanceState)</pre>	<pre>{super.onCreate(savedInstanceState); setContentView(R.layout.activity_main); but=(Button) findViewById(R.id.B_button); view=(TextView) findViewById(R.id.TV_view); put=(EditText) findViewById(R.id.ET_put); but.setOnClickListener(new View.OnClickListener(){@Override public void onClick(View v) { view.setText(put.getText());}});} @Override public boolean onCreateOptionsMenu(Menu menu) { getMenuInflater().inflate(R.menu.main, menu); return true;}}</pre>
---	--

Стартира се емулятор. Стартира се програмата. Ако няма грешки и програмата може да бъде изпълнена, в папка `bin` се създава файл с разширение `apk`. Това е инсталационният файл, от който може да се инсталира приложението на мобилно устройство.



**Създаване на контекстно меню.** При мобилните телефони контекстното меню се активира при по-продължителен допир върху съответната част на екрана.

<p>В работната папка res (създава се автоматично при започване на нов проект) се създава нова папка menu. В папка res се създава Android XML файл context_menu.xml.</p> <pre>&lt;?xml version="1.0" encoding="utf_8"&gt; &lt;menu&gt; &lt;item android:id="@+id/item_one" android:title="first item"/&gt; &lt;item android:id="@+id/item_two" android:title="second item"/&gt; &lt;/menu&gt;</pre> <p>Добавя се един компонент TextView и един EditText. Задава се на всеки компонент съответното id.</p> <pre>android:id="@+id/text_view" android:id="@+id/edit_text"</pre>	<p>В активити класа на Java се декларират две private полета mTextView и mEditText.</p> <p>В главния xml файл се записва:</p> <pre>mTextView=(TextView)findViewById(R.id.text_view); mEditText=(EditText)findViewById(R.id.edit_text);</pre> <p>Свързване на компонентите с менюто:</p> <pre>registerForContextMenu(mTextView); registerForContextMenu(mEditText);</pre> <p>Създаване на метода onCreateContextMenu():</p> <pre>@Override public void onCreateContextMenu(ContextMenu, m, View v, ContextMenuInfo i) {super.onCreateContextMenu(m, v, i); m.setHeaderTitle("context menu\"); MenuInflater inflater=getMenuInflater(); Inflater.inflate(R.menu.my_menu, m);}</pre>
--	---

**Работа с базата от данни SQLite.** SQLite е релационна база от данни с отворен код, поддържаща стандарта SQL. Реализирана е като библиотека към приложението, а не като самостоятелно работеща програма. Концепцията на SQLite е цялата база да бъде съсредоточена в един единствен файл. Това я прави база данни без сървърен процес, особено подходяща за използване в мобилни устройства, таблети и софтуер, където е невъзможно поддържането на сървърен процес.

**Създаване на база от данни.** Базата от данни SQLite е вградена в Android и може да се използва без необходимост от допълнителни библиотеки и инсталации.

<p>В главния Java Activity файл в метода onCreate() се декларира и инициализира база от данни на SQLite по следния начин:</p> <pre>SQLiteDatabase db; db=openOrCreateDatabase(fileName.d SQLiteDatabase.CREATE_IF_NECESSARY,null);</pre> <p>Дефиниране на таблица с поле за номер и поле за текстови стойности:</p> <pre>final String; CREATE_NAMEOFTABLE="CREATE TABLE tb_nameoftable("+id INTEGER PRIMARY KEY AUTOINCREMENT, "+Name_Column TEXT);";</pre>	<p>Изпълнение на SQL заявката:</p> <pre>db.execSQL(CREATE_NAMEOFTABLE);</pre> <p>Въвеждане на стойности в таблицата:</p> <pre>ContentValues cv=new ContentValues(); cv.put("Name_Column", "tekst1"); db.insert("tb_nameoftable", null, cv); cv.put("Name_Column "tekst2"); db.insert("tb_nameoftable null, cv);</pre> <p>Затваряне на базата от данни:</p> <pre>db.close();</pre> <p>Заклучване на базата от данни:</p> <pre>db.setLockingEnabled(true);</pre>
---	--

**Практическа задача.** Да се създаде информационна система на дадена тема (темата може да се определи от учителя или да бъде по избор на учениците). Данните да се съхраняват в база от данни. Да се предостави възможност за преглед на информацията, извличане на информация по дадени критерии, добавяне, редактиране, актуализиране и изтриване на данни. Да има меню за достъп до различните функционални възможности и контекстни менюта на необходимите места.



Примерно мобилно приложение, създадено от ученици, участващи в извънкласните обучения. „Животните“ е мобилно приложение, което дава възможност на децата да се запознаят с животните, да научат имената им, да видят как изглеждат те, да чуят какви звуци издават и да се позабавляват със забавни клипчета и песнички за тях.



Програмата съдържа следните основни модули: 1) Имена на животните; 2) Звуци на животните; 3) Песни; 4) За приложението. В първия модул потребителят може да прослуша наименованието на съответното животно, като натисне върху неговата икона. Във втория да изслуша какъв звук издава животното, а в третия да прослуша детски песнички за някои животни.

<pre> <b>AndroidManifest.xml</b> &lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;manifest xmlns:android= http://schemas.android.com/apk/res/android package="com.example.animals" android: versionCode="1" android:versionName="1.0"&gt; &lt;uses-sdk android:minSdkVersion="8" android:targetSdkVersion="18"/&gt; &lt;application android:allowBackup="true" android:icon= "@drawable/ic_launcher" android:label= "@string/app_name" android:theme= "@style/AppTheme"&gt;&lt;activity android:name= "com.example.animals.MainActivity" android:label="@string/app_name" android:screenOrientation="portrait"&gt; </pre>	<pre> &lt;intent-filter&gt; &lt;action android:name= "android.intent.action.MAIN"/&gt; &lt;category android:name= "android.intent.category.LAUNCHER"&gt; &lt;/intent-filter&gt; &lt;/activity&gt; &lt;activity android:screenOrientation="portrait" android:name="imena"&gt;&lt;/activity&gt; &lt;activity android:screenOrientation= "portrait" android:name="zvuci"&gt;&lt;/activity&gt; &lt;activity android:screenOrientation= "portrait" android:name="pesni"&gt;&lt;/activity&gt; &lt;activity android:screenOrientation= "portrait" android:name= "info"&gt;&lt;/activity&gt; &lt;/application&gt;&lt;/manifest&gt; </pre>
---	---

<pre> <b>pesni.java</b> package com.example.animals; import android.app.Activity; import android.os.Bundle; public class pesni extends Activity { protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.pesni);}} </pre>	<pre> <b>zvuci.java</b> package com.example.animals; import android.app.Activity; import android.os.Bundle; public class zvuci extends Activity { protected void onCreate(Bundle savedInstanceState) {super.onCreate(savedInstanceState); setContentView(R.layout.zvuci);}} </pre>
---	--

<pre> <b>imena.java</b> package com.example.animals; import android.app.Activity; import android.os.Bundle; import android.view.View; import android.view.View.OnClickListener; import android.widget.ImageButton; public class imena extends Activity implements OnClickListener{ public ImageButton cat; public ImageButton bear; public ImageButton bird; </pre>	<pre> public ImageButton dog; public ImageButton donkey; public ImageButton elephant; public ImageButton goat; public ImageButton mouse; public ImageButton pig; protected void onCreate(Bundle savedInstanceState) {super.onCreate(savedInstanceState); setContentView(R.layout.imena);} @Override public void onClick(View v) { }} </pre>
---	---

<pre> <b>MainActivity.java</b> package com.example.animals; import com.example.animals.R; import android.os.Bundle; import android.app.Activity; import android.content.Intent; import android.view.Menu;import android.view.View; import android.view.View.OnClickListener; import android.widget.ImageButton; public class MainActivity extends Activity implements OnClickListener{ public ImageButton btnimena; public ImageButton btnzvuci; public ImageButton btnpesni; public ImageButton btninfo; @Override protected void onCreate(Bundle savedInstanceState) {super.onCreate(savedInstanceState); setContentView(R.layout.activity_main); btnimena=(ImageButton) findViewById(R.id.btnimena); btnzvuci=(ImageButton) findViewById(R.id.btnzvuci); </pre>	<pre> btnpesni=(ImageButton) findViewById(R.id.btnpesni); btninfo=(ImageButton) findViewById(R.id.btninfo); btnimena.setOnClickListener(this); btnzvuci.setOnClickListener(this); btnpesni.setOnClickListener(this); btninfo.setOnClickListener(this); } @Override public boolean onCreateOptionsMenu(Menu menu) { getMenuInflater().inflate(R.menu.main, menu); return true; } @Override public void onClick(View arg0){ if(arg0.getId()==btnimena.getId()) {startActivity(new Intent(this,imena.class));} else if(arg0.getId()==btnzvuci.getId()) {startActivity(new Intent(this,zvuci.class));} else if(arg0.getId()==btnpesni.getId()) {startActivity(new Intent(this,pesni.class));} else if(arg0.getId()==btninfo.getId()) {startActivity(new Intent(this,info.class));}} </pre>
--	--

**Заклучение.** Описаният метод за прилагане на мобилното програмиране е успешно използван в часовете за извънкласна работа по информатика през последните пет години в МГ „Баба Тонка“, гр. Русе, като средство за по-ефективното и бързо въвеждане на технологичните новости. Той се разви като естествено допълнение към целия обучителен курс по информатика. За учениците с изявени интереси в областта, създаването на мобилни приложения се оказва важен фактор, подпомагащ тяхното личностно утвърждаване, развитие и придобиване на полезни знания и умения.

#### ЛИТЕРАТУРА

- [1] O. CINAR. Android Apps with Eclipse. Apress, 2012.
- [2] K. MERRIFIELD MEW. Android 3.0 Application Development CookBook. Packt Publishing Ltd, 2011.
- [3] M. GARGENTA. Learning Android. O'Reilly Media, Inc., USA, 2011.
- [4] R. SCHWARZ, P. DUTSON, J. STEELE, N. TO. The Android Developer's Cookbook: Building Applications with the Android SDK. Addison-Wesley, 2013.
- [5] L. DARCEY, S. CONDER. Teach Yourself Android Application Development in 24 Hours. Sams Publishing, 2012.
- [6] J. WEI. Android database programming. Birmingham, UK: Packt Pub., Ltd., 2012.
- [7] <http://developer.android.com>
- [8] <http://www.minedu.government.bg>

Сюзан Феимова  
Математическа гимназия „Баба Тонка“  
ул. „Иван Вазов“ 18  
7000 Русе  
e-mail: sfeimova@gmail.com

**CREATING APPLICATIONS FOR MOBILE PHONES AND TABLETS  
WITH ANDROID OPERATING SYSTEM IN EXTRACURRICULAR  
CLASSES IN INFORMATICS**

**Suzan Feimova**

This paper presents training materials for students on how to create mobile apps in extracurricular classes in informatics. A sample distribution of the teaching material is proposed. Methodological aspects are considered. Introduction to mobile programming in Java and programming environment Eclipse is made. A short description of the creation of a context menu in mobile applications for Android is given. The foundations for creating databases with SQLite are described.