



IT ШКОЛА SAMSUNG

# Индивидуальный проект

## Белка

Город: Уфа

Площадка: МБОУ Лицей №83

Учащийся: Хайруллин Эмиль Рамилевич

Преподаватель: Дружинская Елена Владимировна

Дата: 16 мая 2016 г.



С ЧЕГО НАЧАТЬ  
ОБУЧЕНИЕ?



## Язык должен быть...

...Легким в понимании

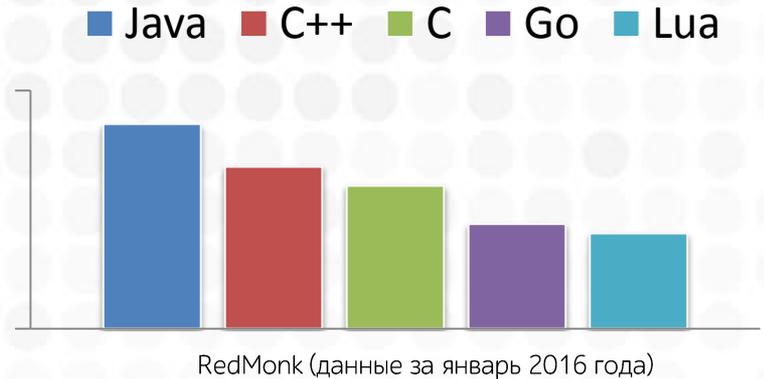
...Быстрым в изучении

Язык **Lua** вобрал в себя лучшие качества других языков программирования и по праву считается хорошей основой для изучения.

**Lua** обладает незамысловатым синтаксисом и семантикой.



Язык программирования **Lua** входит в 20ку востребованных ЯП мира по версии рейтинга **RedMonk**.





**В ЧЕМ  
ЗАКЛЮЧАЕТСЯ  
УСПЕШНОЕ  
ОБУЧЕНИЕ?**





Хорошее средство обучения есть ключ к успешному обучению. И одним из таких средств является приложение **Belka!**



Легкий путь в изучении **Lua!**



**Belka** – это обучающая система, которая поможет Вам изучить язык программирования **Lua** в кратчайшие сроки. Приложение **Belka** совмещает в себе теорию и практику в виде уроков с заданиями.

В приложении имеется два раздела:



И





- Раздел предоставляет краткий и информативный курс.
- Присутствует обновление курса - загрузка новых уроков!
- После каждого урока можно пройти задание и узнать, усвоен ли материал.

Уроки	↻
Стартовый	
Переменные	
Условная конструкция if	
Циклы	
Функции	



- Место для полёта фантазии! – Написали код в «Песочнице»? Его можно запустить и посмотреть в действии!
- Для запуска кода используется самописный интерпретатор **Lua**.
- Написанные программы можно сохранять на устройстве и делиться ими со своими друзьями!

&lt;/&gt; Песочница (RandMatrix)



```
colors = {"red", "green", "blue", "magenta"}
a = 1
while a ~= 6 do
  b = 1
  while b ~= 6 do
    screen.setPos(a*100, b*100)
    screen.setColor(colors[math.rand(0,3)])
    screen.drawRect(100, 100)
    b = b + 1
  end
  a = a + 1
end
```



В редакторе кода «Песочница» есть 4 кнопки, которые отвечают за его главные функции:



**Просмотр сохраненных файлов**



**Создания нового файла**



**Сохранение файла**



**Запуск программы**

</> Песочница (RandMatrix)



```
colors = {"red", "green", "blue", "magenta"}
a = 1
while a ~= 6 do
  b = 1
  while b ~= 6 do
    screen.setPos(a*100, b*100)
    screen.setColor(colors[math.rand(0,3)])
    screen.drawRect(100, 100)
    b = b + 1
  end
  a = a + 1
end
```



Обработка кода интерпретатором проходит в 3 этапа.

## Этап 1

```
a, b = 1, a+math.PI
if a ~= b then
    screen.drawText("YES")
else
    screen.drawText("NO")
end
```

### **LMain**

Генерирует абстрактно-синтаксическое JSON-дерево

На вход интерпретатору подается некоторый код и управление передается **LMain**



## Этап 2

### **LMain**

Генерирует абстрактно-синтаксическое JSON-дерево

```
[
  ["a", "", "b", "=", "1", "", "a", "+", "math", ".", "PI"],
  {
    "type": "if",
    "cond": ["a", "~=", "b"],
    "stmt": [
      ["screen", ".", "drawText", "(" , "b", ")"],
      ["b", "=", "b", "-", "a"]
    ],
    "else": [
      ["screen", ".", "drawText", "(" , "a", ")"]
    ],
    "elseif": [],
  }
]
```

Результатом работы объекта класса LMain является абстрактно-синтаксическое JSON-дерево кода, переданного в этапе 1.



## Этап 3

```
[  
  ...JSON-tree from  
  2nd step...  
]
```

**LExpr**  
Класс обработка JSON-дерева  
методом рекурсивного спуска.

**LType**  
Представление типов  
данных Lua

**LConst**  
Класс постоянных, содержащий  
в себе стандартные функции и  
объект с переменными и  
пользов. функциям.

LFunc

LStdlib



Необходимо доработать/сделать:

1. Перенести все библиотечные функции из стандарта Lua
2. Написать класс оптимизации кода

Вывод по работе:

Я доволен тем, что реализация получилась лучше чем, я ожидал.



**СПАСИБО ЗА ВНИМАНИЕ!**