*Резюме –* **Създаден от японеца Юкихиро Мацумото, по прякор „Мац“, основната цел наезика, според създателя му, е да направи програмистите щастливи, като им спести ненужната работа, следвайки принципите за дизайн на потребителски интерфейс.**

*Ключови думи –* Ruby**,** език за програмиране**.**

I**.** Въведение

Руби е лесен и интуитивен обектно ориентиран **език за програмиране,**  събрал вдъхновение и черти от много други езици като Smalltalk**,**  Perl**,** Lisp и Python**.**

II**.** Принципи и идеали

Поведението на езика трябва да е интуитивно и той трябва да работи така**,** както програмистът би очаквал**.** Възможно е крайният резултат да е „принципът на най - малка изненада за Мац“ **,**  но огромен брой програмисти по света твърдят**,**  че езикът съответства и на техния модел на мислене**.** **Ruby** е перфектно балансиран език**.** Създателят му често казва**,**  че се е опитал да създаде „естествен**,**  а не лесен за писане език“ **.**

Добромир Огнянов Иванов – Технологично училище Електронни системи към Технически Университет – София

[syliconsynapse@gmail**.**com](mailto:syliconsynapse@gmail.com)

Резултатът е лесен и удобен за работене и научаване език**,**  който има широк диапазон от приложения за интернет и други приложения**.**

III**.** Растеж и популярност

Още от създаването си през 1996 година езикът привлича вниманието на програмистката общност**.** Десет години по - късно**,**  през 2006 година **Ruby -** Talk – основниятпощенски списък за **Ruby** получава по над 200 съобщения на ден**.** Езикът е поставен на 9 - то място сред растящите **езици за програмиране,** голям принос**,**  за което има популярният фреймуърк за интернет приложения **Ruby** on Rails или както често бива съкращаван Rails или RoR**.**

Тъй като **Ruby** е проект с отворен код**,** всеки е свободен да го използва и модифицира както намери за добре!

IV**.** Всичко е обект

В езика **Ruby** всичко е обект**.** Това включва операторите**,** примитивните типове ( за разлика от останалите езици за програмиране ) и дори числата**.** По тази тема Mitz споделя: „Исках скриптов език за програмиране**,** по - мощен от Perl и далеч по - обектноориентиран от Python**.**“ **.**

Всеки един бит информация може да притежава свои атрибути и методи**,** което прави езика изключително гъвкав**.**

V**.** Гъвкавост

**Ruby** се счита за изключително гъвкав език**,** тъй като програмистът е свободен да променя всяка една част от кода по свое желание**.**Езикът има за цел да премахне всички ограничения**,** от които страдат останалите **езици за програмиране.**Всеки един от операторите в зеика може да бъде предефиниран по желание на програмиста**,** което дава голяма свобода и гъвкавост в използването на различните оператори**.**

VI**.** Блокове

**Ruby** има още една ключова за гъвкавостта си черта - „затваряния“ **.**Те могат да се прикачат към всеки метод**.**Програмистът прикачва „затваряния“ и описва как ще се изпълняват методите в тях**.**

„Затварянето“ се нарича блок и се е превърнало в една от най - популярните функции за начинаещите програмисти**,** мигрирали от други **езици за програмиране,**  като PHP**,**  Visual Basic и др**..**

Идеята за блоковете е почерпена от процедурните **езици за програмиране.**

VII**.** Множествено унаследяване

За разлика от повечето обектно ориентирани **езици за програмиране,**  **Ruby** не предлага възможността за множествено унаследяване**,**  но това в никакъв случай не е недостатък на езика**.** Вместо това програмистът може да добавя модули ( наричани още Категории в Objective - C ) **.** Модулите представляват**,** най - общо**,** колекции от методи**.** Програмистът може да добавя тези колекции в обектите си и да получава достъп до методите им**,**  реализирайки по този начин унаследяване**.**

IIX**.** Променливи

**Ruby** не се нуждае от декларация на променливи**.**Но за сметка на това притежава изключително лесен и прост за четене и разбиране начин на именуване на различните променливи**.**Например:

var е локална променлива**.**

@var е променлива на инстанцията**.**

$var е глобална променлива**.**

Този начин на именуване на променливите спомага за по - лесното четене и разбиране на кода**,**  като позволява лесна идентификация на ролята на всяка променлива**.** Също така този начин на именуване премахва нуждата от използването на ключовате дума self**,**  като идентификатор на всеки член от инстанцията**,**  и така спомага за по - бързото и лесно писане на код**.**

IX**.** Неконвенционални черти

**Ruby** притежава много и уникални черти някои от които са:

* Прихващане и обработка на ексепшъни ( exceptions ) **,** както в Java**,** Python и други обектно ориентирани езици**,** което улеснява манипулацията и обработка на грешки**.**
* Притежава истински Garbage Collector за всички обекти**,** което премахва нуждата от ръчно заделяне и освобождаване на памет**,** което допълнително улеснява работата на програмиста и му позволява да се съсредоточи върху основната част от приложението**.**
* Писането на C разширения в Ruby е по - лесно от това на Perl или Python**,** с елегантно API за извикване на Ruby от C**.** Това спомага за вграждането на Ruby като скриптов език**.** Налична е поддръжка на SWIG интерфейс**.**
* Позволява динамично зареждане на разширителни външни библиотеки **,** но само ако операционната система го позволява**.**
* Позволява програмиране с нишки**,** независимо от операционната система**.**
* **Ruby** е преносим език**,** което му дава предимнство пред останалите програмни езици**.** Създаден е предимно на GNU/Linux**,** но работи под много UNIX - базирани операционни системи**,** MacOS X**,** Windows 95 / 98 / Me / NT / 2000 / XP**,** DOS**,** BeOS**,** OS / 2 и др**.**

X**.** Мигриране от други езици

Когато някой начинаещ в **Ruby** програмист**,** който има опит в програмирането с други езици**,** започне да пише програми на езика **Ruby,** може да му се стори много познато**.** Това е направено с цел**,** но и разбира се**,** е следствие от факта**,** че синтаксисът на езикът е взаимстван от други такива**,** като напримве Java**,** Python**,** Perl**,** Smalltalk**.**

XI**.** Rails

**Ruby** on Rails или както се съкращава често – Rails е фреймуърк с отворен код**,**  написан изцяло на **Ruby,**  който има за цел да улесни живота на програмистите при създаването на интернет приложения**.**Използвайки Rails**,**  програмистът може да създава сложни и гъвкави интернет приложения**,** каквито биха му отнели седмици и дори месеци**,**  само в рамките на броени дни**.** Това качество на Rails го е превърнало в много широко използван фреймуърк в света на интернет програмирането**.** Някои от най - големите и известни сайтове в интернет използват имено Rails**.** Това са GitHub**,** Basecamp**,** Twitter**,**  Yellow Pages**,** Shopify**,**  Livingsocial**,**  Mingle2 и много други**.**

XII**.** Заключение

**Ruby** е изключително гъвкав и интуитивен език за програмиране**,** което го прави лесен за научаване и удобен за използване**.** Фактът**,** че е проект с отворен код му помага да се развива още по - добре и по - бързо**.** Все пак както всички знаят проектите с отворен код винаги са по – добри от платените**.**

Езикът **Ruby** има висока степен на интеграция**,** лесно се поддържа и позволява бърза и лесна разработка на големи проекти**.** Тези му качества в комбинация с фреймърка за интернет приложения**,** Ruby on Rails или както често се съкращава – просто **Rails** го правят адекватен избор за работа при разработката на нови интернет приложения**.**

Начинаещите програмисти успавят лесно и бързо да го усвоят тъй като синтаксисът му, взаимстван от различни други езици, е създаден специално с цел да бъде интуитивен и изчистен.

Тези които преди това са писали на други езици го намират за лесен за свикване, тъй като си прилича с други езици. Тези и други причини спомагат за бързо нарастващата популярност на езика за програмиране **Ruby**.