**RUBY:**

- Tudo é objeto;

- Variáveis não presisam de tipo, apenas atribuição de valores;

Exemplo:

* num: 25
* boolean: true / false (sem aspas)
* string: "com aspas"

**COMANDOS BÁSICOS:**

print - printa tudo na mesma linha

puts - printa com quebra de linha

“#{var}” -> chama a variável para printar

# - comenta a linha

=begin

=end - comenta um bloco de código

var.class – mostra a classe da variável

var.object\_id – mostra o identificador único do objeto

* **Para Strings:**

var.lengt - retorna o tamanho da string

var.reverse - retorna a string invertida

var.upcase - transforma a string toda maiúscula

var.downcase - transforma a string toda em minúscula

var.capitalize - transforma a primeira letra em maiúsculo e as demais minúsculas

var.include? - compara a string

* If var.include? “string”

var.gsub! - substitui letra ou palavra da string

* var.gsub! (/letra\_ou\_paravra/, “nova\_letra\_ou\_palavra”)

var.split(“\*\*\*\*”) - pega todo o conteúdo digitado pelo o usuário e separa pelo conteúdo dentro das aspas e coloca em uma array, ignorando o conteúdo dentro das aspas, caso nada seja digitado dentro da mesma separa por espaço em branco.

* var.gets.chomp.split(“ ”)
* var.gets.chom.split

1. var.gets.chomp
2. array = var.split(“ “)

* **Pegar dados do usuário:**

var = gets.chomp – para pegar string

var = gets.to\_i – para pegar inteiro

var = gets.to\_r – para pegar real

gets - pega a informação

chomp -remove os espaços em branco

**CONDIÇÃO:**

**if** condição

Código

**elsif** outra condição

Código

**else**

Código

**end**

A diferença entre o “if” e o “unless” é que para entrar na condição “if” ela tem que ser verdadeira, se não passa para o “else’s” em diante, já o “unless” a opção tem que ser falsa, caso contrário ela continuará a executar os blocos de código. O mesmo funciona para o “while” e “until”.

**unless** condição

Código

**else**

Código

**end**

**LOOP**

**while** condição verdadeira

Código

**end**

**until** condição falsa

Código

**end**

**for** contador in **inicio .. fim** ou **inicio ... fim**

Código

**end**

0 ... 3 -> significa que 0 <= var < 3

0 .. 3 -> significa que 0 <= var <=3

**GEMS:** As “GEMS” não as bibliotecas ou conjuntos de arquivos Ruby reutilizáveis, etiquetados com um nome e uma versão.