

Ruby Essential

Строки. Врем и дата.

Ruby Essential

Автор курса



Юля Гончаренко

Ruby Essential

После урока обязательно



Повторите этот урок в видео формате на ITVDN.com

Доступ можно получить через руководство
вашего учебного центра



Проверьте как Вы усвоили данный материал
на TestProvider.com

Ruby Essential

Тема

Строки. Время и дата

Ruby Essential

План урока

1. Строки. Способы создания строк.
2. Конкатенация строк.
3. Основные методы для работы со строками: получение доступа, сравнение строк, управление регистром и изменение строк .
4. Время . Основные методы для работы.

Ruby Essential

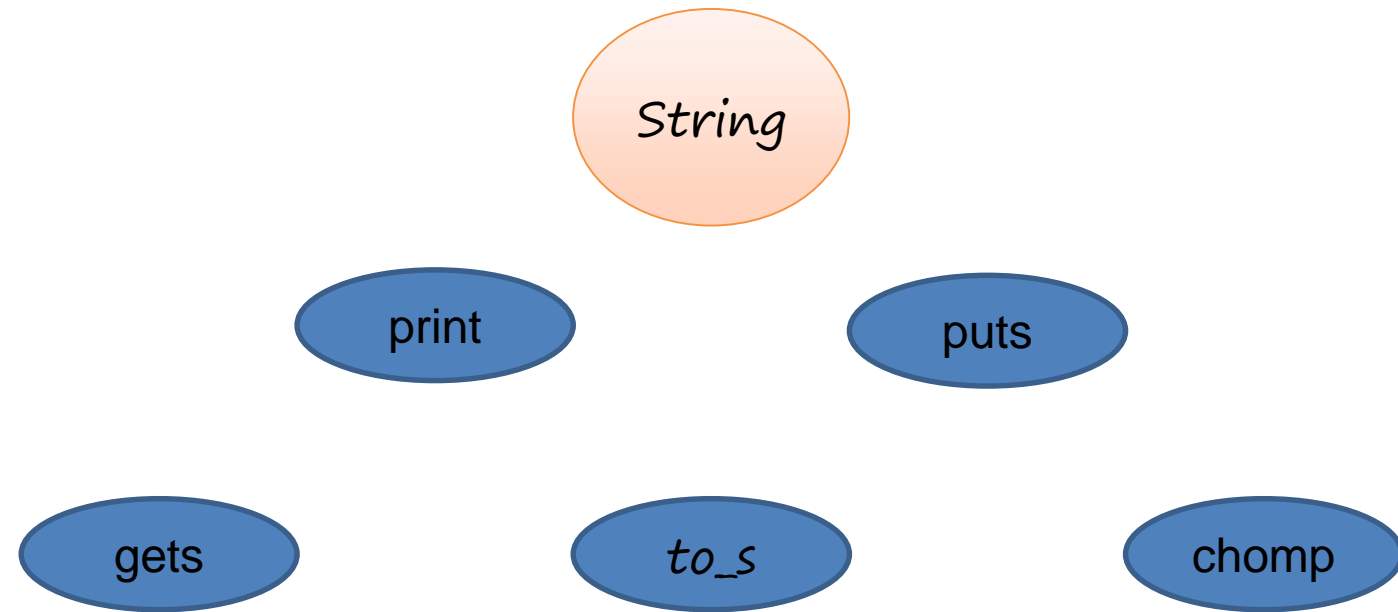
Что мы знаем о строках из предыдущих уроков?

- Существует тип данных String.

- Методы для вывода строк

- Методы для работы со строками

- Интерполяция строк "#{ ... }" - вставка в строку выполняемого фрагмента кода



Ruby Essential

Строки

Строка - это последовательность букв, цифр и других символов.

Строки хранятся как произвольные последовательности байт.

Строки создаются методом `new` или как литералы.

Строки являются изменяемыми объектами, для это рассмотрим множество методов для работы с ними.

Ruby Essential

Создание строк

Строки создаются методом new или как литералы.

'Строка,
заклученная в
одинарные кавычки '

`s1='Значение строки'`

"Строка,
заклученная в
двойные кавычки"

`s2="Значение строки"`

%!Строка в
ограничителе!

`s3=%q[Значение строки]`

`String.new(s4=" ")`

Документ

`s5=<<EOF Значение строки EOF`

Ruby Essential

Понятие конкатенации строк

Конкатенация - это склеивание объектов. В Ruby можно объединять строки, применяя разные способы конкатенации.

`str1 + str2`
`"string1" + "string2"`

`str1 << str2`
`"string1" << "string2"`

`str1 * N`
где N – количество раз, сколько нужно повторить данную строку

`str1.concat(str2)`

Ruby Essential

Получение доступа к строке

Получение доступа к строке используется для доступа к ее отдельным частям, извлечения и изменения частей строки. Для такого доступа используется оператор `[]`, или же как его называют — оператор индекса массива.

В случае строки в квадратных скобках мы указываем индекс символа, находящийся в строке. При указании индекса Ruby возвращает одно символьную строку.

`str1` `[Индекс 1, индекс 2]`

`str1` `[Диапазон]`

`str1` `["Подстрока"]`

`str1` `[Регулярное
выражение]`

Ruby Essential

Получение длины строки

Для получения длины строки используют два идентичных метода: `length` и `size`.

По своей сущности они одинаковы, поэтому можно считать что методы являются разными именами одного метода.

`str.length` `#=> 6`



`str.size` `#=> 6`



Ruby Essential

Управление регистром

`str1.downcase`

- обеспечивает перевод каждого символа строки в нижний регистр.

`str1.upcase`

- обеспечивает перевод каждого символа строки в верхний регистр.

`str1.swapcase`

- переводит регистр каждого символа в строке на противоположный.

`str1.capitalize`

- переводит в верхний регистр первый символ строки, а остальные символы переводит в нижний регистр.

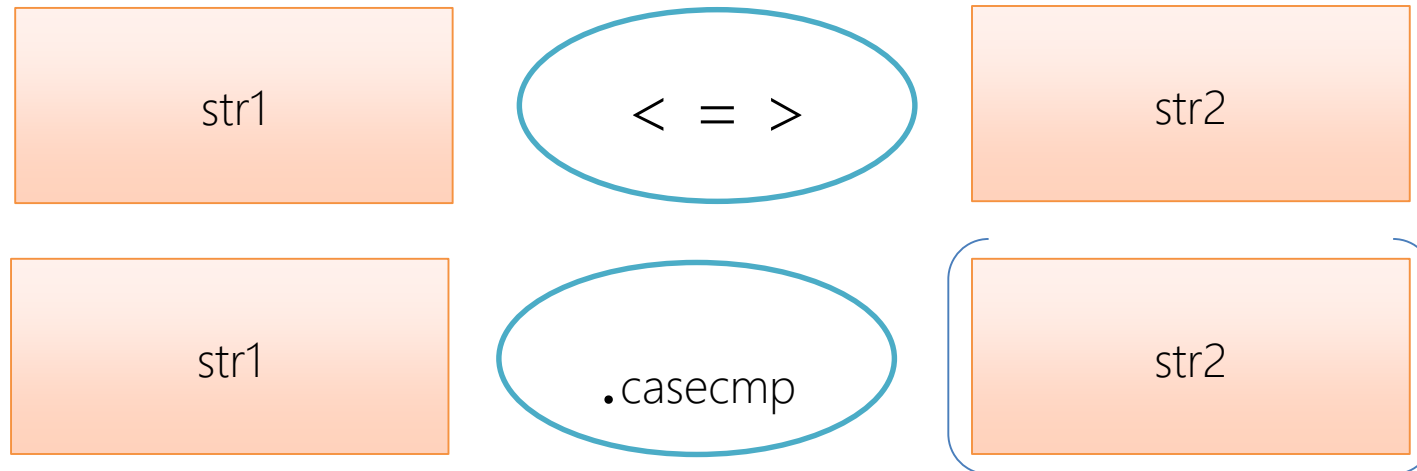
Ruby Essential

Сравнение строк

По умолчанию строки сравниваются в лексикографическом порядке.

Существует 2 способа сравнения строк, результатами которых являются значения 1, 0, -1. Это оператор `< = >` и метод `.casecmp`.

Оба способа сравнивают значения кодов символов в строках. Отличием при сравнении методом `.casecmp` от оператора `< = >` является то, что данный метод не учитывается регистр символов в строке.



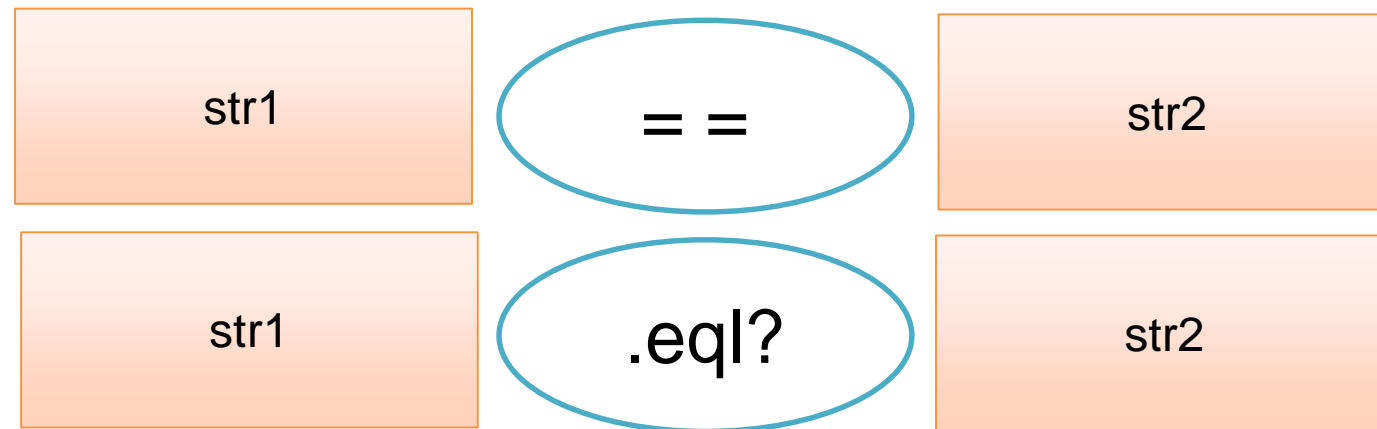
Ruby Essential

Сравнение строк

Простыми способами сравнения строк являются использование оператора `==` и метода `.eq?`.

Эти 2 способа в результате сравнения возвращают результат `True` – если строки равны, и `False` – если строки различны.

Различием этих двух способов является то, что оператор `==` возвращает `True`, если 2 объекта типа `String` равны, а метод `.eq?` возвращает `True`, если две строки равны по своему размеру и содержанию. В противоположных случаях возвращается значение `False`.



Ruby Essential

Методы изменения строк

Методы	Описание
insert	Данный метод принимает в качестве аргументов строку и позицию вставки (индекс) для вставки данной строки в строку для которой он был вызван, при этом символы находящиеся в том месте, куда производится запись не перезаписываются , а сдвигаются вперед.
split	Разбивает строку на части и возвращает массив лексем. В этот метод передаются два параметра -разделитель(разделителем может выступать также другая строка)и лимит(положительное или отрицательное число), которое отвечает за количество возвращаемых подстрок строки. Если параметры не заданы, по умолчанию разделителем является символ пробела.
reverse	Заменяет последовательность символов в обратном порядке.
tr	Позволяет заменять найденные символы по условию на новые.

Ruby Essential

Что мы знаем о строках из предыдущих уроков?

Ruby Essential

Методы удаления

Метод	Описание
delete	Этот метод обеспечивает удаление указанных символов или указанного диапазона символов из строки .
chop	Удаляет последний символ строки.
chomp	Удаляет разделитель записей(то есть символ перевода строки).
strip	Удаляет пробелы вначале и в конце строки.
squeeze	Удаляет вхождение повторяющихся символов из строки, оставляя лишь один экземпляр символа. Есть возможность задания списка символов, на которые будет действовать данный метод.

Ruby Essential

Другие методы

Метод	Описание
empty?	Данный метод проверяет, пустая строка или нет. Знак вопроса обеспечивает возвращаемый результат в виде булевых значений.
include?	Метод принимает фрагмент строки и проверяет строку для которой был вызван на то, существуют ли в ней подстроки соответствующие фрагменту.
index	Позволяет возвращать номер позиции, с которой начинается заданная подстрока начиная отсчет с начала строки.
rindex	Метод аналогичен index, но возвращает индекс начиная отсчет с конца строки.

Ruby Essential

Время

Объекты Time дают возможность получать такие параметры, как время, дата и часовой пояс в различных форматах, а также работать с заданными или полученными значениями, преобразовывать форматы, сравнивать моменты времени и форматировать эти значения.

Создание нового объекта Time осуществляется методом `new`.

Методом получения текущих значений является метод `now`.

Установка текущего местного значения даты, времени и пояса объекта Time осуществляется методом `local`. Метод `Time.gm` по сути устанавливает то же самое, но время считается уже по формату GMT (Гринвич). Синонимом данного метода является метод `utc`.

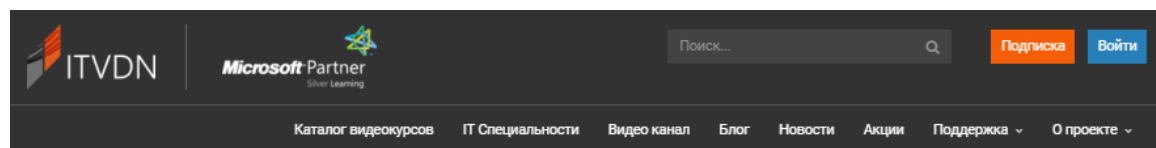
Ruby Essential

Основные методы для работы

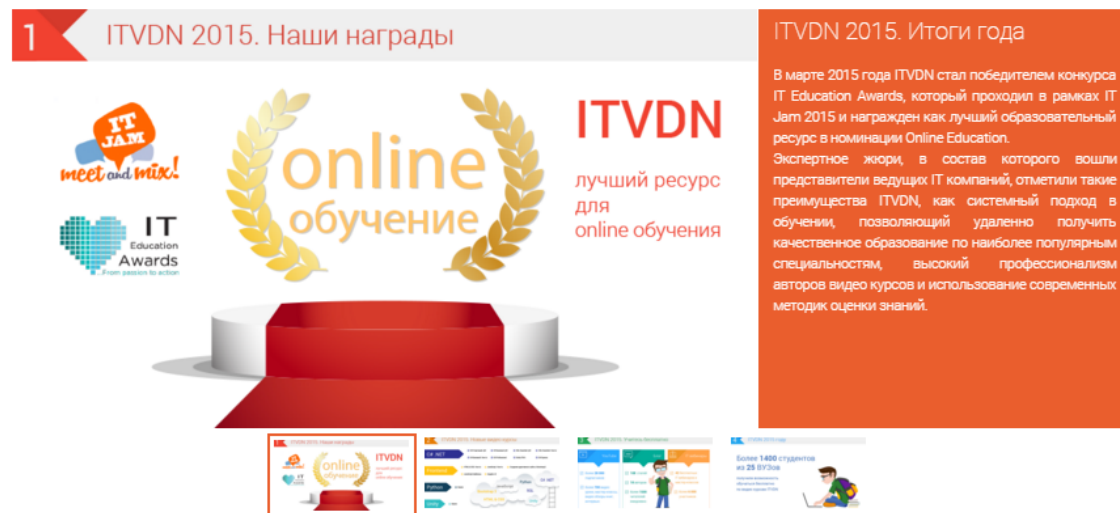
Метод	Описание
year	Год.
month	Месяц.
day	День месяца.
Wday	День недели.
yday	День в году.
hour	Часовой формат времени.
min	Формат времени в минутах.
sec	Формат времени в секундах.
usec	Формат времени в микросекундах.
zone	Часовой пояс.

Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.



Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics

Новые видео

Исключения	0
Итераторы и генераторы	0

Популярные видео курсы

Видео курс C# Стартовый (для начинающих)	9 уроков (16 ч. 3 мин.)
Видео курс по шаблонам проектирования	29 уроков (16 ч. 7 мин.)

Теги

.NET Developer
Frontend Developer



Проверка знаний

TestProvider.com



TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на TestProvider.com

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



Ruby Essential

Q&A

Информационный видеосервис для разработчиков программного обеспечения

