The Ruby Masterclass

Contents:

7. Intro to the gets method.
8. Intro to the tap method on objects.
9, Intro to the then method on objects.
10. Quiz.

SECTION 5: Basic Arithmetic I

1. Intro to Integer class and Float class.
2. +, -, *, /, %, ** and other methods of Integer class.
3. `next` / `succ` methods.
4. `odd?`, `even?` methods.
5. Integer / Float to String, String to Integer / Float conversion.
6. More methods on Float class: truncate, to_i, round, next_float, ceil, floor, zero?
7. The nil? method on objects.
8. The object_id methods on objects.
9. The is_a?, kind_of and === methods.
10. The Spaceship operator (<=>).
11. Intro to syntactical sugars / syntactic shorthand.
12. Binary, Octal, Hexadecimal literals.
13. Quiz: Celsius to Fahrenheit converter.

SECTION 6: Booleans

[We will be using String and Integer / Float class for demonstration.]
1. The true, false, and nil objects.
2. Equality Operator (==) and Inequality Operator (!=)
3. Greater Than Operator (>)
4. Lesser Than Operator (<)
5. Greater Than Equal (>=)
6. Lesser Than Equal (<=)
7. Truthiness and falsiness, not and ! Operators.
8. The Exclusive Or method.
9. The respond_to? Method.
10. Quiz: Guess the outputs!

SECTION 7: Strings I

1. Creating strings: ‘’, “”, %q(), %Q(), %// syntax.
2. String interpolation.
3. Escape Characters.
4. Using the ==, ===, eql? Methods
5. length, size, count methods.
6. Using the upcase, downcase, capitalize and swapcase methods.
7. The reverse method on a string object.
8. String concatenation:
- + and += Operators.
- concat method.
- Shovel operator (<<).
- Efficient practices.

9. The empty? Method on a string object.
10. Extract character(s) from string using Bracket Syntax [].
11. Intro to mutation of string objects.
12. The clear and replace methods.
13. The slice method with various arguments.
14. Introduction to bang methods on string.
15. Intro to chomp, chop, strip, lstrip, rstrip and their complementary bang methods.
16. Intro to index, rindex methods.
17. Intro to squeeze, squeeze! methods.
18. Quiz: Palindrome in least SLOC.

SECTION 8: Methods I
1. Introduction to the def keyword.
2. Introduction to the complementary define_method keyword.
3. Return values.
4. Parameters and arguments.
5. Default return values.
6. More on returning booleans.
7. Calling a method from another method.
8. Challenge: Adding binary numbers in method!
9. Challenge: Creating our custom include? Method.

SECTION 9: Conditions
1. The if keyword with and without then keyword.
2. Intro to inline modifiers.
3. Reviewing truthiness and falsiness.
4. Intro to if, elsif, and nested if conditions.
5. The unless keyword.
6. The &&, &, and Operators.
7. The ||, |, or Operators.
8. Parentheses and Precedence.
9. The ternary operator.
10. Challenge: Grade Calculator in least SLOC.
11. The case statement.
12. The case subsumption operator (===).
13. The ! Operator and not method.

SECTION 10: Loops and Iterations - I
1. Introduction to times loop.
2. Introduction to while loop.
3. Introduction to for loop.

SECTION 11: Challenge FizzBuzz in the least lines of code.

SECTION 12: Range Objects
1. Intro to Range objects.
2. Alphabetical Ranges.

3. Numerical Ranges.
4. to_a method
5. The .size method.
6. min, max, minmax methods.
7. min_by, max_by, minmax_by methods
8. The include? method
9. The cover? Method
10. The Kernel.rand / rand method
11. The exclude_end? method
12. Quiz: Converting a Range of Hexadecimal numbers to decimal.
13. Quiz: Adding a Range of binary numbers!

SECTION 13: Accessing ruby-doc.org, apidock.com for online documentation.
Using ri for accessing offline documentation on Linux, Termux, and Windows.

SECTION 14: Accessing Ruby Interpreters Online.

SECTION 15: Array I

1. Intro to Arrays
2. Creating Array with Bracket Syntax.
3. Creating an Array with Array.new
4. Using %w() or %w{ } or %w[] syntax.
5. The size, length, and count methods.
6. Accessing Array Items with Bracket Syntax.
7. The index method.
8. Finding array elements with the find, and select methods.
9. the sort method.
10. The reverse method.
11. The push, append, concat, += and the Shovel Operator (<<).
12. The pop, delete, delete_at methods.
13. shift and unshift Methods.
14. Accessing array elements with fetch method.
15. Using the slice method.
16. Overwrite values with the bracket syntax.
17. The first, and last methods.
18. Equality and Inequality Operators.
19. The Spaceship (<=>) operator.
20. sort method with <=> operators.
21. Quiz.

SECTION 16: Array II

1. The uniq and uniq! methods.
2. The inject / reduce method for one liners!
3. The sum method.
4. The clone and dup method.
5. The flat_map method.
6. The clear method.

7. The permutation and combination methods.
8. The select!, reject, reject!, drop, drop_while Methods
9. the find, any?, all? Methods.
10. Quiz: Make our custom select method.
11. The shuffle, shuffle! Methods.
12. The sample method.
13. The rotate method.
14. The empty? Method.
15. The cycle method.
16. The each_with_object method
17. The compact, and compact! Methods.
18. Quiz: Make our custom compact! Method.

SECTION 17: Array III

1. the each and for loops.
2. The map / collect, map! / collect! method for one liners!
3. The take method.
4. The freeze method.
5. Nested arrays and transpose method.
6. Array Union and Intersection.
7. Quiz: Create our custom transpose method.
8. Unpacking one and multidimensional arrays.
9. The partition method.
10. The group_by method.

SECTION 18: Symbol

1. Intro to Symbol objects.
2. Usage of Symbols.
3. Converting Symbol objects to String objects and vice versa.
4. object_id of Symbols.

SECTION 19: Hash

1. Intro to Hash Objects.
2. Adding Key and Value pairs with Bracket Syntax: Using various Objects as key-value pair.
3. The fetch method on a Hash.
4. The store method.
5. The merge and merge! methods.
6. The clear method.
7. The length and empty? methods
8. The keys and values methods.
9. The each method for iteration.
10. The flatten method, and conversion of Hash to Array and vice versa.
11. Quiz.
12. The sort and sort_by methods.
13. The default_proc method on a Hash.
14. The select and reject method on a Hash.

SECTION 20: Count the occurrence of words in an Array and return a Hash (Enumerable#tally) with as fewer lines of code as possible.

SECTION 21: Methods II

1. Splat Arguments.
2. Intro to Blocks.
3. The yield keyword.

4, Additional arguments in a method with a block.

5. Calling blocks with .call(), [], .[](), .===(), and .() methods, and &:method syntax.
6. The block_given? method.
7. { } vs. do end block.
8. aliasing methods – the alias keyword.
9. The method() method:
   - calling methods with .call method.
   - knowing the owner of a method.
   - Using the arity method.
10. Using the instance_method method

SECTION: 22: Procs and Lambdas

1. Intro to Procs.
2. Quiz.
3. Intro to lambdas.
4. Lambda vs. Proc.
5. The arity method.

SECTION 23: Loops and Iterations II

1. while loop.
2. until loop.
3. for loop.
4. each loop.
5. times loop.
6. step loop.
7. upto loop.
8. downto loop.
9. loop loop.
10. Intro to redo, break, next keywords..
11. Revising map / inject loop for one-liners.
12. Quiz: Creating our own loops.

SECTION 24: Time Object

1. Introduction to Time class.
2. Instance methods of time objects.
3. Adding and subtracting a Time object from another Time object.
4. Different formats of time with strftime (string from time)
5. The monday?, tuesday?, wednesday?, etc. methods.
6. The String Parse Time, Time gem, and its methods.
7. Using the Date gem.

8. Challenge 1: Time converter.
9. Challenge 2:

SECTION 25: Introduction to File IO

1. Introduction to Reading text files with File and IO classes.
2. Writing to a text file:
 - The open method and its arguments from File class.
 - Using the write method from File class.
3. Reading Linux's /dev/urandom for a custom random generator.
4. Using the zero? method from File class.
5. Using the exists? method from File class.
6. Using readable? , writable?, executable? methods from File class.
7. Intro to blockdev?, chardev?, symlink? methods.
8. Loading gems into Ruby:
   - The load method.
   - The require_relative method.
   - The require method.
   - The Kernel#require method.
9. Introduction to the standard library.
10. Using the prime gem from the standard library.
11. Quiz: Creating an Eratosthenes Generator!
12. The gem command.
13. Intro to www.rubygems.org

SECTION 26: String II: Regular Expressions (Regexp)

1. The include? method
2. The start_with?, end_with? methods.
3. Intro to rubular.com
4. The scan method.
5. The sub, and gsub methods.
6. Class Regexp
7. Challenge: Creating a camelize method.

SECTION 27. STRINGS III

1. Intro to the split method.
2. Intro to the chars method.
3. Intro to the join method.
4. Intro to the each_char method.
5. Intro to the each_line method.
6. Intro to various here documents.
7. The bytesize method.
8. The ord and chr methods.
9. The pack and unpack methods.
10. Challenge: Our custom chars method.

SECTION 28: Using the Internet

1. Using the net/http standard library gem.

2. Using the open-uri standard library gem.
3. Downloading images and music with Ruby.

SECTION 29: Cryptography - String IV

1. Using the crypt method for cryptography.
2. Breaking Linux encryption by brute force.
3. Using Permutation to generate passwords.
4. Using the digest gem from the standard library.
5. Using the openssl gem from the standard library.
6. The Kernel.eval / binding.eval() methods.
7. Have I been pawned?

SECTION 30: Generating Random passwords after confirming if they have been leaked or not.

SECTION 31: Classes I

1. Intro to classes.
2. The class method on every object.
3. The superclass and ancestors method.
4. The methods method.
5. The singleton_methods method.
6. Creating a class.
7. Review of object_id and the equal? methods.
8. Instance methods.
9. Getters.
10. Setters.
11. Shortcuts to getters and setters:
   - attr_reader.
   - attr_writer.
   - attr_accessor.
13. Adding parameters to instance other methods.
14. Intro to self keyword and singleton methods.
15. Quiz.

SECTION 32: Modules and Mixins

1. Intro to Modules.
2. Intro to the Math module.
3. Intro to the Scope Resolution Operator (::).
4. Inheriting classes.
5. Intro to Mixins.
6. The include keyword.
7. The prepend keyword.
8. The extend keyword.
9. Intro to the Enumerable module.
10. Quiz.

SECTION: 33: Classes II

1. Public methods.
2. Private methods.
3. Protected methods.
4. Instance methods and instance variables.
5. Using setter and getter in depth.
6. Using Struct.
7. Revising methods aliasing.
8. Unbound Method.
9. Quiz.

SECTION 34: Monkey Patching: Classes III

1. Introduction to monkey patching.
2. Using Monkey patching
3. Custom class with monkey patching.
4. Using the instance_method, bind, and call methods.
5. Other ways to create methods, and call methods.
6. Using the send, and method keywords.
7. instance_eval method.
8. Challenge: Defining prev and prev! methods on String class.

SECTION 35: Introduction to Error Handling

1. Intro to the begin, rescue, else, ensure keywords.
2. Intro to BEGIN, END, exit, exit!, abort, at_exit.
3. Intro to the retry keyword.

SECTION 36: Introduction to Refinements:

1. Intro to the refine keyword.
2. Intro to the using keyword with a refined class.
3. Refining Array, String and a class.

SECTION 37: Challenge: Enhancing our password generator.

SECTION 38: Challenge: Anagrams from a 100K words open dictionary file.

SECTION 39: Reading and Writing Documentation

1. Intro to class method documentation.
2. Installation of rdoc.
3. Using rdoc to generate own HTML documentation.

SECTION 40: Introduction to benchmarking

1. Use the benchmark gem from the standard library.
2. Performance comparison between loops.
3. Performance comparison: String Concatenation: +=, concat, and << methods.
4. Performance comparison: Array concatenation: +, concat, push, append, <<, unshift methods.
5. Performance comparison: Array pop, delete_at, shift, methods.
6. Benchmarking 3 Array uniq sorts (no algorithm).
7. Benchmarking bubblesort, mergesort, and ruby's quicksort.

8. Benchmarking method call, lambda call, proc call, stabby lambda call.
9. Intro to ObjectSpace.

SECTION 41. Introducing my 'patched-irb' for Students' Convenience.

SECTION 42: Executing Shell commands:
1. Using back-ticks (``).
2. Using %x() or %x{ } or %x[] syntax.
3. Calling the system method.
4. Using the open3 gem from the standard library.
5. Intro to the shell gem from the standard library.

SECTION 43: Standard Input, Standard Output, Standard Error and ANSI Escape sequences
1. Colourize your strings!
2. Gradient RGB colours on strings.
3. Blinking / underlining texts and many more.
4. Replicating the `clear` command with ANSI Escape Sequence!
5. Looking at STDIN, STDOUT, STDERR classes.
6. Looking at 50 ways to print "Hello World" to the standard output!
7. Looking at the Warning module.
8. Kernel.warn() vs. Warning.warn().
9. Looking at the SimpleDelegator class and patching $stdout object.
10. Intro to 'io/console' gem from the standard library:
 - Determining the terminal size.
 - Getting single characters from the user with STDIN.getch
11. Quiz: Creating a colourful output on a user input with gradient RGB colours.

SECTION 44: Introduction to 2D Game Development
1. Intro to Ruby2D.
2. Installation of Ruby2D – Not a Standard Library Gem.
3. Intro to creating Square, Circle, Rectangle, Triangle, Line, Quads – Using gradient RGB colours.
4. Intro to Images by Using copyleft images.
5. Using loops to create thousands of Ruby2D objects and animating them in lesser than 15 lines of code.
6. Intro to Fonts (using downloaded open source fonts for our project from the internet)
7. Intro to Sound and Music (using copyleft music for our project from the internet)
8. Monkey Patching Ruby2D.

SECTION 45: Bonus Challenge: Developing a Colour::Clock.

SECTION 46: Bonus Challenge: Developing a Snake Game in Lesser than 150 LOC!

SECTION 47: Intro to File Compression and decompression with Zlib from the Standard Library.

SECTION 48: A short intro to Various Ruby Interpreters and Performance Comparison. [Demonstration Only]
1. A brief intro to RVM.
2. Using multiple Ruby Versions with RVM in short.

3. Intro to Rubinius and Jruby in short.

SECTION 49: Accessing the Course Resources from GitHub
1. Intro to GitHub Web.
2. Downloading all the resources as a zip file.
3. A sample Ruby code to download the course resources recursively to your local storage.

SECTION 50: Conclusion and Congratulation.