OVERVIEW

Each block contains the task of developing a same concept in pure Ruby, then in Postgres and RoR
Understand utilization of Ruby and Postgres principles in RoR web apps.
Master thinking patterns that speed up development process.
Take advantage of one-to-one QA sessions with your teacher.
Work by yourself with the concise description of specific feature/functionality/concept, exercises and handy library of most relevant and up-to-date online resources.
Start solving tasks right away if you are already familiar with some Ruby and RoR code.
For a newbie - you can start your journey here with the basic knowledge and exercises with solutions

BLOCK 1	single model
BLOCK 2	second model + has_many belongs_to relationships
BLOCK 3	many_to_many relationship
BLOCK 4	so on

TERMS & CONDITIONS

Join now
(sign up)

BLOCK 1

RUBY

Detailed description of the task
Follow the path == link

RoR

Detailed description of the task
Follow the path == link

Postgres

Detailed description of the task
Follow the path == link

Ruby instructions

STEPS:

CREATE SINGLE CLASS

OBJECTS/@VARIABLES

INITIALIZE, GETTER, SETTER

CUSTOM METHODS

IRB

GITHUB

per step:
WHAT TO DO
link to **Explore TOPIC**
link to step-by-step **INSTRUCTION HOW TO DO IT** for those who are not able to complete the task
list of resources
link to additional exercises
SHOW / HIDE THE SOLUTION

communicator
for student - teacher exterchage

LINK to **Explore the FEATURE**

LINK to additional **EXERCISES**

LIST of **RESOURCES**

RoR instructions

- || -

Postgres instructions

- || -

TOPIC

- short but meaningful description + illustration if needed

- LINK to relevant exercises with solutions

- list of online resources

TASKS **(separate tables for Ruby, RoR, PG)**
title

STEPS
title | description | solution

TOPICS
title | description | images

EXERCISES
feature | description

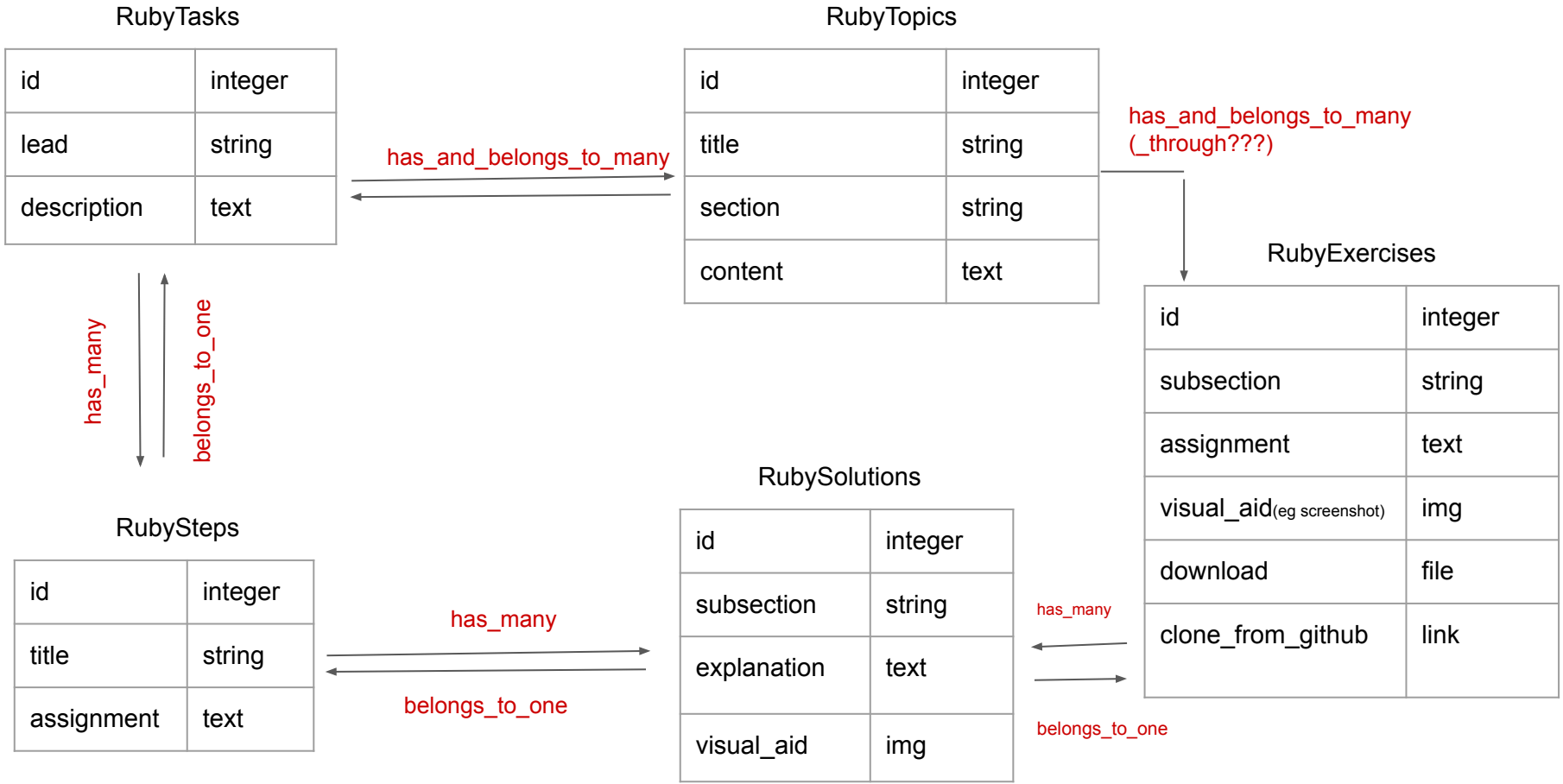
SOLUTIONS
subject | solution

Possible relations:

TASK has_many STEPS
STEP has_many EXERCISES
STEP has_many TOPICS
TOPIC has_many EXERCISES
EXERCISE has_many SOLUTIONS

TABLE_NAME
columns / attributes
do we need a table for that??

?????
- **STEP-BY-STEP EXECUTION FOR TOTAL BEGINNERS**
- **RESOURCES**


RubyTasks
id
integer
lead
string
description
text
has_and_belongs_to_many
RubyTopics
id
integer
title
string
section
string
content
text
has_and_belongs_to_many
(_through???)
RubyExercises
id
integer
subsection
string
assignment
text
visual_aid(eg screenshot)
img
download
file
clone_from_github
link
has_many
belongs_to_one
RubySteps
id
integer
title
string
assignment
text
has_many
belongs_to_one
RubySolutions
id
integer
subsection
string
explanation
text
visual_aid
img
has_many
belongs_to_one