**Junior**

**Общие вопросы**

**1. Какие принципы программирования вы знаете и какие из них вы пытаетесь соблюдать чаще всего?**

DRY, SOLID:

S - [Принцип єдиного обов'язку](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF_%D1%94%D0%B4%D0%B8%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%BE%D0%B2%27%D1%8F%D0%B7%D0%BA%D1%83): Кожен [об'єкт](https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)" \o "Об'єкт (програмування)) має виконувати лише один обов'язок.

O - [Принцип відкритості/закритості](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF_%D0%B2%D1%96%D0%B4%D0%BA%D1%80%D0%B8%D1%82%D0%BE%D1%81%D1%82%D1%96/%D0%B7%D0%B0%D0%BA%D1%80%D0%B8%D1%82%D0%BE%D1%81%D1%82%D1%96) Програмні сутності повинні бути відкритими для розширення, але закритими для змін.

L - [Принцип підстановки Лісков](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF_%D0%BF%D1%96%D0%B4%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B8_%D0%9B%D1%96%D1%81%D0%BA%D0%BE%D0%B2): Об'єкти в [програмі](https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%27%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B0" \o "Комп'ютерна програма) можуть бути заміненими їх нащадками без зміни коду програми.

I - [Принцип розділення інтерфейсу](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF_%D1%80%D0%BE%D0%B7%D0%B4%D1%96%D0%BB%D0%B5%D0%BD%D0%BD%D1%8F_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81%D1%83" \o "Принцип розділення інтерфейсу): Багато спеціалізованих [інтерфейсів](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_(%D0%9E%D0%9E%D0%9F)" \o "Інтерфейс (ООП)) краще за один універсальний. Інтерфейс може бути поділений на спеціалізовані ще на стадії проектування, заради майбутньої гнучкості програмних компонентів.

D - [Принцип інверсії залежностей](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF_%D1%96%D0%BD%D0%B2%D0%B5%D1%80%D1%81%D1%96%D1%97_%D0%B7%D0%B0%D0%BB%D0%B5%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9) : Залежності всередині системи будуються на основі [абстракцій](https://uk.wikipedia.org/wiki/%D0%90%D0%B1%D1%81%D1%82%D1%80%D0%B0%D0%BA%D1%86%D1%96%D1%8F" \o "Абстракція), що не повинні залежати від деталей; навпаки, деталі мають залежати від абстракцій. [Модулі](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D1%83%D0%BB%D1%8C_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B8%D0%B9" \o "Модуль програмний) вищих рівнів не мають залежати від модулів нижчих рівнів.

**2. Что такое парадигма ООП? Какие основные составляющие парадигмы?**

[інкапсуляція](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D0%BA%D0%B0%D0%BF%D1%81%D1%83%D0%BB%D1%8F%D1%86%D1%96%D1%8F), [успадкування](https://uk.wikipedia.org/wiki/%D0%A3%D1%81%D0%BF%D0%B0%D0%B4%D0%BA%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)" \o "Успадкування (програмування)), [поліморфізм](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D1%96%D0%BC%D0%BE%D1%80%D1%84%D1%96%D0%B7%D0%BC_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)" \o ") та абстракція

**3. Зачем нужна инкапсуляция? Что может произойти и не произойти, если все методы будут иметь модификатор доступа public?**

**4. Что такое наследование? Назовите пример(ы) плохого наследования (такое наследование, которое не стоило бы реализовывать).**

**5. Что такое полиморфизм? Приведите пример использования полиморфизма.**

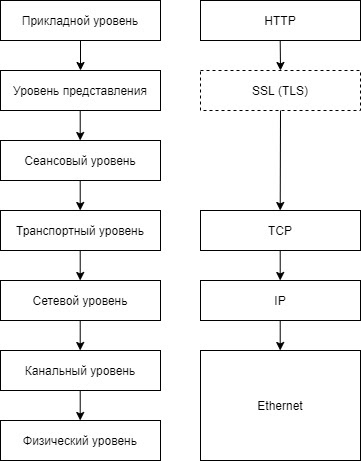
полиморфизм — використання єдиного інтерфейсу для [різнотипних](https://uk.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85" \o "Тип даних) сутностей або у використанні однакового символу для маніпуляцій над даними різного типу.

**6. Что такое паттерн/шаблон проектирования? Какие шаблоны знаете?**

Фабрика, сінгелтон, адаптер, білдер,

**7. Что такое клиент и сервер? Какая механика их взаимодействия?**

**8.Опишіть шлях, який проходить запит після того, як у браузер ввести URL будь-якого ресурсу?**



**9.Що таке протокол HTTP? Яка різниця між HTTP та HTTPS?**  
**10.Що таке API? Чи може вебзастосунок використовувати більше, ніж один API?  
11.Що таке REST API? Які дії найчастіше виконуються над ресурсами?**  
12.Що таке сесія? Що таке cookies? Яка різниця між ними?  
13.Які вебвразливості знаєте? Як від них захиститися? Які є інструменти, щоб автоматично контролювати захист від вебвразливостей?  
14.Якими критеріями користуєтесь для вибору бібліотеки?  
15.Що таке рефакторинг? Які повинні бути умови для виконання рефакторингу?  
16.Чи знаєте ви сімейство UNIX-подібних операційних систем (Linux, MacOS)? Назвіть команди, які найчастіше використовуєте в терміналі.  
17.Що таке аутентифікація?  
18.Що таке авторизація?  
19.Назвіть, з яких частин складається HTTP Request та HTTP Response?  
20.Які HTTP-статуси (HTTP-коди) знаєте? На які групи ви б могли їх розділити?  
21.Які HTTP-методи знаєте? Навіщо використовується кожен з них?

**Git**

22.Чи ознайомлені із системою Git?  
23.Для чого потрібні гілки у Git? Як зробити гілку?  
24.Як скопіювати з Git-хостингу (GitHub / GitLab / Bitbucket / інші) проєкт?  
25.Що таке флаг -f? Для чого він та які мінуси використання?  
26..Що таке Pull Request? Навіщо гілки поєднують через Pull Request?  
27.Які ще команди найчастіше застосовуєте, коли користуєтесь Git?

**Ruby**

28.Як ви контролюєте версії Ruby на своїй локальній машині?  
29.Що таке метапрограмування? Які плюси та мінуси під час використання метапрограмування?  
30.Який менеджер бібліотек для Ruby знаєте? Як додавати бібліотеки? Як контролювати версії бібліотек?

**Style guide**

31.Що таке style guide і навіщо його використовують?  
32.Який популярний лінтер контролює виконання більшої частини пунктів з Ruby Style Guide?

**Основи Ruby**

33.Які типи змінних є в Ruby?  
34.Що повернеться в результаті складання 10.5 та 10?  
35.Чим відрізняються змінні, що починаються з @, @@ та $?  
36.Що таке attr\_reader, attr\_writer, attr\_accessor і навіщо вони?  
37.Що таке метод?  
38.Що означає «?» в кінці назви методів? Що означає «!» в кінці назви методів?  
39.Чим відрізняється енумератор map від each?  
40.Що таке модуль? Які є варіанти під’єднання модулів у клас?  
41.Що таке клас і навіщо він потрібен?  
42.Що таке модуль і навіщо його використовують? Чим модуль відрізняється від класу?  
43.Що таке константа?  
44.Що таке блок? Що таке yield? Чи можна передавати блоки в методи? Яким чином?  
45.Що таке proc?  
46.Що таке lambda?  
47.У чому різниця між proc і lambda?  
48.Що таке eigenclass?  
49.Які види умовних операторів є в Ruby? (if, unless, ternary, case)  
50.Чи є різниця між операторами && і and?  
51.Які види циклів є в Ruby? (loop, while, until, for, each(), times() upto())

**Типи даних**

52.Які типи даних є в Ruby?  
53.Що таке інтерполяція?  
54.Що таке конкатенація?  
55.Що таке масив у Ruby?  
56.Що таке Hash у Ruby? Чим Hash відрізняється від масиву?  
57.Що таке символ і навіщо він потрібен?  
58.Що таке Range і навіщо його використовують?  
59.Що таке Time та Date?  
60.Що таке Struct?  
61.Що таке File?

**Структура класів та об’єктів**

62.Що таке BasicObject?  
63.Що таке Numeric?  
64.Що таке Enumerable?  
65.Що таке Struct?  
66.Що таке OpenStruct?  
67.Що таке Set?  
68.Що таке Queue?  
69.Що таке Object?  
70.Що таке Kernel?

**ООП на Ruby**

71.Чи є інтерфейси в Ruby?  
72.Що таке getter і setter в Ruby? Як їх створювати?  
73.Що таке інстансна змінна?  
74.Що таке константа класу?  
75.Яка різниця між методом класу і методом об’єкта?  
76.Які є області видимості?  
77.Які є модифікатори доступу? Як private відрізняється від protected?  
78.Чи підтримує Ruby множинне спадкування?  
79.Як успадковуються методи? Чи можна перевизначити метод?  
80.Навіщо метод super?  
81.Що таке Singleton-метод?  
82.Що таке міксини? Навіщо вони і чим відрізняються від декоратора?  
83.Що таке include та extend і в чому різниця між ними?  
84.Який шлях пошуку методу в ієрархії класів?  
85.Яка різниця між послідовними та іменованими аргументами методу?  
86.Що таке exception? Коли вони застосовуються? Як викликати exception? Як відловити exception?

**Метапрограмування**

87.Що означає open class?  
88.Що таке monkey patching?  
89.Чи можливо створити клас динамічно? Якщо так, то як саме?  
90.Які є способи динамічного керування методами, змінними та класами (define\_method(), class\_eval(), module\_eval())?  
91.Як відбувається пошук методу в ієрархії? Як можна опрацювати відсутність методу (method\_missing())?  
92.Що таке аліас методу? Для чого потрібні аліаси?  
93.Як отримати значення інстансної змінної? Як задати значення? Як перевірити існування інстансної змінної (instance\_variable\_get(), instance\_variable\_set(), instance\_variable\_defined?)?  
94.Які методи інтроспекції є в Ruby (class(), ancestors(), methods(), private\_methods(), instance\_variables(), singleton\_methods(), local\_variables())?  
95.Що таке хуки й навіщо їх використовують (inherited(), included(), method\_added(), singleton\_method\_added(), method\_missing())?

**RubyGems**

96.Що таке gem? Що таке RubyGems?  
97.З якою метою використовують RVM? Що таке gemset?  
98.Що таке Bundler і навіщо він?  
99.Що таке Gemfile і для чого його використовують?  
100.Назвіть, за що відповідає кожен з gem’ів. Якщо знаєте нюанси в роботі якогось з них — обов’язково озвучуйте.

active\_admin  
active\_model\_serializers  
bundle audit  
bcrypt  
devise  
jwt  
cancancan/pundit  
draper  
kaminari  
sidekiq  
rspec  
shoulda-matchers  
faker  
factory\_bot  
simplecov  
rubocop (with extensions)  
pry

**Ruby on Rails**

101.Що таке MVC? Чи є Rails MVC-фреймворком?  
102.Що таке міграція? Для чого вона потрібна? Які є правила створення міграцій?  
103.Що таке роути? Що таке resource та resources в роутах?  
104.Чим відрізняється has\_many through від has\_and\_belongs\_to\_many?  
105.Навіщо використовують concerns?  
106.Як працювати із sensitive-даними в Rails-застосунках (токени, ключі тощо)? Як працює механізм credentials в Rails?  
107.З якою метою використовують локалі (locals)?

**Style guide**

108.Чи ознайомлені ви з RoR style guide?

**Структура файлів у Rails**

109.З чого складається Rails-застосунок? Яка архітектура директорій? Навіщо потрібна кожна з них?  
110.Чим відрізняється lib від vendor?  
111.Як створити background job в Rails?  
112.Які є способи завантаження асоціацій у Rails?

**Моделі та ActiveRecord**

113.Навіщо потрібні моделі? Чим є модель?  
114.Що таке ORM? Що таке ActiveRecord?  
115.Які є правила найменування моделей і таблиць у базі даних?  
116.Що таке міграції та навіщо потрібен цей механізм? Як створювати міграції?  
117.Які види валідацій є в AR?  
118.Як можна викликати валідацію? Коли валідація викликається автоматично? Як можна вимкнути валідацію під час збереження?  
119.Що таке callback в Rails?  
120.Які є види колбеків в AR?  
121.Які види асоціацій є в AR?  
122.Що таке has\_one і belongs\_to?  
123.Що таке has\_many?  
124.Що таке has\_many: through?  
125.Що таке has\_one: through?  
126.Що таке has\_and\_belongs\_to\_many?  
127.Що таке поліморфна асоціація?  
128.Що таке STI?  
129.Яким чином можна отримати дані з бази даних? Як отримати один об’єкт і колекцію? Як задати умови вибірки?  
130.Що таке скоупи? Яка їхня функція?  
131.Як виконати довільний SQL-запит?  
132.Як подивитися інформацію про те, як виконувався запит, які використовувалися ключі тощо?

**Контролери**

133.Що таке контролер? Яке завдання виконує?  
134.Що таке action? Що таке CRUD action?  
135.Що таке permit params?  
136.Що таке ActiveRecord? Як він реалізований в Rails-фреймворку?  
137.Що таке параметри? Як вони потрапляють у контролер? Що таке дефолтні параметри (в роутах)? Що таке стронг-параметри?  
138.Що таке cookies і навіщо вони? Де зберігаються cookies? Як отримати доступ до cookies?  
139.Що таке сесії та в чому їхнє завдання? Де зберігаються сесії? Як отримати доступ до сесії?  
140.Які види фільтрів є в Rails? (before / after / around actions)  
141.Навіщо потрібен об’єкт Request? Яку інформацію він містить?  
142.Для чого потрібен об’єкт Response? Яку інформацію він містить?  
143.Що таке стримінг? Яким чином можна віддати файл на завантаження (Send\_data, send\_file)?  
144.Що таке Rescue в контролерах і для чого застосовується?  
145.Як примусово увімкнути використання HTTPS-протоколу?

**Маршрутизація**

146.Що таке роути і в чому їхня функція?  
147.Що означає абревіатура CRUD?  
148.Якщо потрібно зробити новий CRUD для певного ресурсу, які класи ви створите та які файли зміните?  
149.Що таке ресурси?  
150.Що таке Singular ресурси і що відрізняє їх від звичайних ресурсів?  
151.Що роблять неймспейси в роутах?  
152.Що таке вкладені ресурси?  
153.Що таке shallow nesting?  
154.Що означає RESTful роути?  
155.Що таке нересурсні роути?  
156.Що таке actions в роутах?

**View**

157.Що таке view і для чого він потрібен?  
158.Що таке layout? Чим відрізняється від view?  
159.Що таке шаблонізатор і з якими шаблонізаторами доводилося працювати?  
160.Навіщо потрібен метод render? Що можна рендерити за його допомогою?  
161.Як передаються змінні у view?  
162.Що таке partial? Коли варто його застосовувати?  
163.Що таке хелпер і навіщо його використовують?  
164.Які є хелпери для генерації роутів?  
165.Які є хелпери для генерації форм?  
166.Як створювати власні хелпери?

**Інтернаціоналізація Rails**

167.Що таке i18n? Навіщо потрібна i18n? Як вона влаштована в RoR? В якому вигляді можна зберігати перекази?

**ActionMailer**

168.Що таке ActionMailer і які завдання він розв’язує?

**AssetsPipeline**

169.Що таке AssetsPipeline і які завдання він виконує?

**Інструменти консолі та rake-таски**

170.Які є консольні команди для роботи з RoR? Що таке rake-таски?

**Безпека**

171.Чи знаєте якісь вразливі місця, захист від яких Rails надає out of the box?  
172.Які атаки можуть бути при неправильному використанні сесій? Чому не можна зберігати важливі дані в сесіях?  
173.Що таке CSRF? Як правильно захиститися від CSRF?  
174.У чому вразливість Open Redirect? Як захиститися від можливих атак, пов’язаних з Open Redirect?  
175.Які вразливості пов’язані із завантаженням файлів? Чому важливо фільтрувати імена та розширення файлів? Як захиститися від небезпеки?  
176.Чому не можна завантажувати файли в каталог, доступний зовні? Що буде, якщо зловмисник завантажить файл із кодом?  
177.Які вразливості пов’язані з управлінням користувачами (посилання активації, паролі в чистому вигляді)? Як захиститись?  
178.Що таке ін’єкції та як відбувається така атака? Як їх уникнути?  
179.Як організувати безпечну генерацію запрошень? Що таке deep\_munge?

**Конфігурація**

180.Де розташовується основна конфігурація програми?  
181.Що таке initializers? Навіщо вони?  
182.Що таке оточення? Яке його призначення?  
183.Як і де налаштовується взаємодія з базою даних?  
184.Як і де налаштовується Assets, генератори, мідлвари, i18n?  
185.Чи можна вказувати довільні налаштування? Якщо так, то де і яким чином?

**Кешування**

186.Що таке кеш і навіщо потрібен цей механізм?  
187.Які види кешування є в Rails?  
188.Де може зберігатися кеш?  
189.Навіщо використовувати кешування сторінок?  
190.Навіщо використовувати кешування екшенів?  
191.Навіщо використовувати кешування фрагментів?  
192.Що таке Memcached?  
193.Навіщо і як реалізується низькорівневе кешування?  
194.Як і коли кешується SQL?  
195.Як тестувати кешування в режимі розробки?

**Реляційні бази даних**

**Теорія реляційних баз даних**

196.Що таке СУБД?  
197.Що таке реляційна база даних?  
198.Що таке нормалізація бази даних?  
199.Що таке нормальна форма?  
200.Скільки є нормальних форм баз даних?  
201.Опишіть перші три нормальні форми баз даних.  
202.Що таке денормалізація баз даних?  
203.У чому різниця між Primary Key, Foreign Key? Навіщо вони?  
204.Яка різниця між LEFT JOIN, INNER JOIN, RIGHT JOIN?  
205.Як додати запис у таблицю без ORM?  
206.Як оновити запис у таблицю без ORM?  
207.Як видалити запис з таблиці без ORM?  
208.Як прочитати записи в таблиці без ORM?  
209.Які правила групування даних у таблиці ви знаєте?

**SQL**

210.Що таке DML (Data Manipulation Language)?  
211.Що таке DDL (Data Definition Language)?  
212.Що таке DCL (Data Control Language)?  
213.Що таке TCL (Transaction Control Language)?  
214.Як вибрати дані з кількох таблиць? Які види JOIN ви знаєте?  
215.Які агреговані показники можна використовувати під час вибірки з таблиці?  
216.Що таке таблиця у базі даних?  
217.Як створити таблицю?  
218.Як вибрати записи з таблиці?  
219.Як вставити запис у таблицю?  
220.Як змінити запис у таблиці?  
221.Як видалити запис з таблиці?  
222.Як об’єднати вибірку з різних таблиць?  
223.Що таке групування або агрегування?  
224.Яким чином можна задавати умови агрегування?

**Індекси баз даних**

225.Що таке індекси в базі даних?  
226.Для чого потрібні індекси?  
227.Які види індексів існують?  
228.Що таке первинний ключ?  
229.Що таке зовнішній ключ?  
230.Що таке складовий ключ?

**Повнотекстовий індекс у базах даних**

231.Що таке повнотекстовий пошук і в чому його завдання?  
232.Що таке повнотекстовий індекс?  
233.Що таке ts\_vector?  
234.Що таке ts\_query?

**PostgreSQL**

235.Що таке select distinct і яка його функція?  
236.Яким чином можна сортувати результати вибірки?  
237.Як обмежити кількість результатів вибірки?  
238.Як проводити пошук за підрядками?  
239.Які види об’єднань (JOIN) є в PostgreSQL?  
240.Що таке підзапити і в чому їхнє завдання?  
241.Які типи даних є в PostgreSQL?  
242.Які є види символьних типів даних?  
243.Які є види типів даних для роботи з датою та часом?  
244.Що таке тип enum? Коли він використовується?  
245.Як можна створювати таблиці у базі даних?  
246.Як змінювати структуру таблиці?  
247.Як видаляти таблиці?  
248.Як очистити таблицю?  
249.Чи можливо задавати значення за замовчуванням для полів? Якщо так, то як саме?  
250.Чи можливо контролювати унікальність значень під час вставлення? Якщо так, то яким чином?  
251.Що таке функції в PostgreSQL?  
252.Які оператори є в PostgreSQL?

**Бази даних NoSQL**

**Redis**

253.Що це таке Redis і навіщо використовують цей інструмент?

**Elasticsearch**

254.Що таке Elasticsearch і навіщо його використовують?

**Статичний аналіз коду**

255.Що таке Overcommit і навіщо потрібен?  
256.Що таке Danger і для чого використовується?  
257.Що таке статичні аналізатори коду? Навіщо їх використовують?  
258.Які аналізатори для Ruby / Ruby on Rails знаєте?  
259.Які статичні аналізатори коду для JavaScript знаєте?

**Тестування**

260.Які види тестування знаєте? Які з них ви практикували під час використання Rails?  
261.Який фреймворк для тестування використовуєте?  
262.Які показники якості для своїх тестів можете назвати?  
263.За допомогою якої бібліотеки контролюєте процент покриття тестами?  
264.Розкажіть про основи тестування в Ruby? Що таке RSpec? Яка різниця між describe, context та it?  
265.Які різниця між юніт-тестами та інтеграційними?  
266.Що перевіряють приймальним тестуванням?  
267.Що перевіряють інтеграційним тестуванням?  
268.Що тестують у моделях?  
269.Що тестують у контролерах?  
270.Що тестують у серіалайзерах?  
271.Як тестувати API?  
272.Як тестувати різні Service Objects?  
273.Що таке RSpec? Які завдання виконує цей інструмент?  
274.У чому різниця між let і let! у RSpec?  
275.Які хуки в RSpec знаєте?  
276.Що виконується раніше let! чи хук before(:all)?  
277.Що таке Capybara? Які завдання вирішує Capybara?  
278.Що таке стаби?  
279.Що таке моки?  
280.Що таке матчер?  
281.Яке завдання виконує гем Shoulda Matchers?  
282.Яке завдання виконує Faker?  
283.Навіщо потрібні Webmock і VCR? Чим вони відрізняються?  
284.Що таке WebKit і PhantomJS? У чому різниця між ними? Які ще є драйвери?  
285.Як працювати з базою даних у тестах? Як працює gem database\_cleaner?

**Додаткові RubyGems та Rails плагіни**

286.Які завдання виконує Devise?  
287.Які завдання виконує Rake?  
288.Які завдання виконує Cancancan і Pundit?  
289.Які завдання виконує Kaminari та Will\_paginate?  
290.Які завдання виконує Carrierwave?  
291.Які завдання виконує MiniMagick?  
292.Які завдання виконує Aasm?  
293.Які завдання виконує ActiveAdmin?  
294.Які завдання виконує Nokogiri?  
295.Які завдання виконує Formtastic і SimpleForm?  
296.Які завдання виконує Omniauth?  
297.Які завдання виконує Draper?  
298.Які завдання виконує Faye?  
299.Які завдання виконує Pusher?  
300.Які завдання виконує Sidekiq, Resque?  
301.Які завдання виконує Mongoid?

**Управління, сервіси та DevOps**

302.Що таке SSH? Як і для чого використовується?  
303.Яким чином можна налаштувати аутентифікацію на хості без пароля?  
304.Які завдання виконує Heroku? У чому його особливість?  
305.Які завдання виконує Digitalocean?  
306.Які завдання виконує Docker?

**Моніторинг**

307.Які завдання виконують Airbrake, DataDog, NewRelic?  
308.З якою метою використовують Monit?

**Ruby Web Tools**

309.Які є вебсервери, написані на Ruby?  
310.Що таке Sinatra?  
311.Що таке API? Для чого використовується? Що таке серіалайзери і навіщо вони потрібні?  
312.Що таке клієнт-серверна модель?  
313.Яка різниця між GET- та POST-запитами?  
314.Що таке HTTParty і які завдання він виконує?  
315.Що таке Faraday і які завдання він виконує?  
316.Що таке Rack? Що таке Rack-application? Що таке Rack middleware?  
317.Що таке push-технологія?

**Інтеграції**

318.Чи є досвід інтеграції з Facebook? Якщо так, то на якому проєкті?  
319.Чи є досвід інтеграції з Twitter? Якщо так, то на якому проєкті?  
320.Чи є досвід інтеграції з LinkedIn? Якщо так, то на якому проєкті?  
321.Чи є досвід інтеграції з Google? Якщо так, то на якому проєкті?

**Практичні завдання**

322.Описати в ООП-стилі: один із предметів інтер’єру в контексті конкретного бізнесу. Наприклад, крісло, на якому сидить кандидат, тепер треба продавати в магазині.  
323.Написати консольну версію гри «хрестики-нулики», де вхідні дані передаються через gets, а виводять через puts. Два користувачі на одній машині повинні могти зіграти партію.  
324.Відсортувати масив.  
325.Написати функцію, яка поверне унікальні елементи масиву, не використовуючи функцію uniq або set, to\_set зі стандартної бібліотеки.

**Middle**

**Загальне**

1.Що таке SOLID? Наведіть приклади на кожну з літер.  
2.Що таке мікросервісна архітектура? Які плюси та мінуси ви бачите під час її використання?  
3.Чи знаєте ви шаблони проєктування Singleton / Factory / Facade / Decorator / Template Method / Strategy?  
4.Чи можна виміряти швидкість виконання алгоритму? Які чинники впливають на швидкість? Які види залежностей ви знаєте?  
5.Що таке методологія Scrum? На яких активностях / зустрічах може бути присутнім / корисним Software Developer?  
6.Які інструменти для логування використовуєте в роботі?  
7.Які інструменти для моніторингу вебзастосунків використовуєте?  
8.За допомогою якого / яких інструментів ви деплоїли застосунки на staging / production? Опишіть процес.  
9.Якщо один з ендпоінтів у застосунку відпрацьовує дуже довго, що робити, аби покращити швидкість?  
10.Чим відрізняється бібліотека від фреймворку?  
11.Що таке триланкова архітектура? Навіщо та коли її використовують? Наведіть приклади.  
12.Що таке кешування? Навіщо та коли його використовують? Наведіть приклади.  
13.Що таке функціональний розподіл? Навіщо та коли його використовують? Наведіть приклади.  
14.Що таке Comet-сервер? Навіщо та коли його використовують? Наведіть приклади.  
15.Що таке денормалізаця? Навіщо та коли її використовують? Наведіть приклади.  
16.Що таке деградація функціональності? Навіщо вона та в яких випадках використовується? Наведіть приклади.  
17.Що таке сервісно-орієнтована архітектура? Навіщо та коли її використовують? Наведіть приклади.  
18.Що таке монолітний застосунок? Навіщо та коли його використовують? Наведіть приклади.  
19.Що таке асинхронна обробка? Навіщо та коли її застосовують? Наведіть приклади.  
20.Що таке ліниві обчислення? Навіщо та коли їх використовують? Наведіть приклади.

**Предметно-орієнтоване проєктування**

21.Що таке DDD? У чому переваги перед класичним MVC?  
22.Що таке домен?  
23.Що таке модель?  
24.Що таке контекст?  
25.Що таке сутність?  
26.Що таке value object?  
27.Що таке агрегування?

**Git та code review**

28.Що роблять команди cherry-pick, log, stash, reset, squash?  
29.Що таке Gitflow? Які правила використання Git на вашому минулому проєкті?  
30.Яких правил іменування комітів / гілок / Pull Request’ів ви дотримуєтеся?  
31.На що звертаєте увагу під час code review?  
32.Що б ви могли назвати погано проведеним code review?

**Бази даних**

33.Для чого потрібні індекси?  
34.Які є види індексів?  
35.Чи прискорюють індекси всі операції з базами?  
36.Що таке DB view?  
37.Чим відрізняються materialized db view від non-materialized db view?  
38.Як можна зберегти дані в різні таблиці та гарантувати, що всі вони або запишуться, або ні?  
39.Чи можна будувати індекси за кількома полями? Чи важливий порядок цих полів в індексу?  
40.Які ви знаєте constraints під час створення стовпців?  
41.У чому різниця між SQL та NoSQL базами даних?  
42.Як би ви імпортували великі масиви даних у БД (1–2 мільйони рядків у CSV-файл)?  
43.Що таке N+1 та як уникати?

**Реляційні бази даних**

44.Які відмінності між джоінами FULL OUTER JOIN, CROSS JOIN, NATURAL JOIN, INTERSECT та EXPECT?  
45.Які специфічні типи даних є в PostgreSQL?  
46.Що таке view? З якою метою використовується?  
47.Що таке materialized view?  
48.Що таке recursive view?  
49.Що таке збережена процедура і навіщо вона потрібна?  
50.Що таке партиціонування та яку проблему воно вирішує?  
51.Чи вмієте працювати з чистими SQL-запитами?  
52.Яким чином можна працювати з геолокацією в PostgreSQL?  
53.Які є способи резервного копіювання даних? Що таке pg\_dump? У якому вигляді можна створювати резервні копії?

**NoSQL бази даних**

**Redis**

54.Що таке Redis? Які завдання він виконує?  
55.Які типи даних є в Redis і для чого потрібен кожен з них?  
56.Що таке pub / sub?  
57.Що таке транзакції?  
58.Що таке mass insertion?  
59.Що таке партиціонування?  
60.Чи вмієте працювати з Redis у консолі?  
61.Чи вмієте налаштовувати Redis?

**Elasticsearch**

62.Що таке Elasticsearch і які завдання він виконує?  
63.Що таке кластер?  
64.Що таке нода?  
65.Що таке індекс?  
66.Що таке тип?  
67.Що таке документ?  
68.Що таке шарди і репліки?  
69.Що таке Query DSL?  
70.Що таке мапинг?

**Статичний аналіз коду**

71.Що таке overcommit і навіщо потрібна ця утиліта?  
72.Що таке danger і для чого використовують цю утиліту?  
73.Які завдання виконує кожен з нижчеперелічених гемів?

* Rubocop;
* Reek;
* Rails Best Practices;
* Brakeman;
* RubyCritic;
* SimpleCov;
* Bundle Audit;
* Bundle Leak;
* Traceroute.

74.Які завдання виконує кожен з нижчеперелічених модулів?

* Eslint;
* Stylelint;
* Prettier.

**Ruby**

75.Які проблеми в Ruby (як мові програмування) ви бачите, які, на вашу думку, потрібно розв’язувати якнайшвидше?  
76.Якими бібліотеками, окрім фреймворку Rails, користувалися для створення структури проєкту?  
77.Як можна в Ruby надіслати запит на сторонній API? Який вбудований клас дає змогу це зробити?  
78.Що таке Thread? Чим він відрізняється від Process?  
79.Чи є в Ruby можливість створити абстрактний клас? Чи є спосіб штучно створити абстрактний клас?  
80.Чи є в Ruby можливість множинного наслідування? Чи є спосіб штучно замінити множинне наслідування?  
81.Назвіть максимально точний шлях пошуку методів у Ruby?  
82.Які мінуси використання method\_missing знаєте?  
83.Чи можна дізнатись ідентифікатор об’єкта в пам’яті, якщо так, то яким чином?  
84.Які види enumerator для масиву ви знаєте?  
85.Що таке мемоізація методів? Навіщо її використовують?  
86.За допомогою якого хука можна порахувати кількість нащадків певного класу?  
87.У чому різниця запуску rspec та bundle exec rspec?  
88.Чи створювали ви гем у Ruby? Яка механіка / послідовність дій для створення гему?  
89.Що таке Rack middleware?  
90.Є такий код:

    users = User.where(id: [1,3,4])

     users.where(name: 'Alex')

     users.inspect

Після якої стрічки коду ActiveRecord виконує запит в базу?

**Ruby on Rails**

91.Що таке ActiveSupport? Навіщо потрібен? Які завдання виконує? Наведіть приклад методів, які додає ActiveSupport.  
92.Що таке Action Cable? Навіщо використовують цей фреймворк?  
93.Як створити функціональність для відсилання імейлу в Rails-застосунку?  
94.Які бібліотеки для background jobs використовували? Яка механіка роботи таких бібліотек?  
95.Які види кешу в Rails знаєте? Де можна зберігати кеш?  
96.Які бібліотеки / підходи для створення API документації знаєте / використовуєте?  
97.Що таке Swagger? Які переваги створення API документації у Swagger-форматі?  
98.Що таке Query Objects? Коли і як їх можна використовувати?  
99.Що таке Decorators? Коли і як їх можна використовувати?  
100.Що таке Form Objects / Contract? Коли і як їх можна використовувати?  
101.Де та як ви рекомендуєте тримати бізнес-логіку в Rails-застосунку?  
102.Що таке JSON API формат? Чи використовували бібліотеки для серіалізації об’єктів в JSON API форматі?

**Тестування**

103.Чи працювали ви з feature-тестами? Що таке capybara?  
104.Що таке TDD? Які переваги / недоліки використання TDD-підходу бачите?  
105.Що таке BDD? У чому відмінність між TDD та BDD?  
106.Що може бути чинником того, що тести на проєкті виконуються дуже довго?  
107.Що таке принцип тестування FIRST?  
108.Як порекомендуєте писати тести на код, в якому надсилається запит на сторонній сервіс (API)?  
109.Які інструменти допомагають контролювати якість тестів?  
110.Які складові response’у ви тестуєте, коли пишете тест на певний ендпоінт?  
111.Чи є 100% code coverage на проєкті запорукою найнижчої вірогідності щось «зламати» під час багфіксингу / імплементації нової функціональності? Поясніть свою відповідь.

**Управління, сервіси та DevOps**

112.Опишіть CI/CD-підхід на своєму останньому проєкті? Як би ви його змінили?  
113.Які загалом Continuous strategies знаєте? Чим відрізняється Continuous Deployment від Continuous Delivery?  
114.Що таке Docker? Чим відрізняється Docker від віртуальної машини?  
115.Для чого використовують docker-compose?  
116.Чи вмієте працювати з Docker?  
117.Що таке image в Docker? Як працювати з ним у Docker?  
118.Що таке container у Docker? Як працювати з контейнерами? Як запускати та зупиняти контейнери? Як переглянути список робочих контейнерів?  
119.Що таке Docker Daemon?  
120.Що таке Docker Client?  
121.Що таке Docker Hub?  
122.Що таке Docker Compose?

**Практичні завдання**

123.Написати клієнт для відкритого API ресурсу з використанням тільки низькорівневих бібліотек: Net::HTTP, HTTP.rb, Faraday, etc.  
124.Як би ви порахували кількість пасажирів, що заходять чи виходять на певній станції метро?  
125.Порахувати кількість автомобілів у заторі.  
126.Організувати автомобільну стоянку.  
127.Написати функцію, яка поверне унікальні елементи масиву, не використовуючи uniq або set, to\_set зі стандартної бібліотеки. Очікують варіант із застосуванням хеш-таблиці, де ключами будуть унікальні елементи з масиву, а значеннями true.