1. Який файл містить опис контейнерів, що будуть запущені?
   * docker-compose.yml;
   * docker-info.json;
   * docker-compose.xml;
   * docker-compose.uml;
   * docker-info.uml.
2. Який формат має файл docker-compose?
   * YAML;
   * XML;
   * JSON;
   * INF;
   * DKR.
3. Яка команда дозволяє запустити всі контейнери, що описані в файлі docker-compose?
   * docker-compose up;
   * docker compose run;
   * docker-compose push;
   * docker compose down;
   * docker-compose start.
4. Яка команда зупиняє та видаляє всі контейнери?
   * docker-compose down;
   * docker compose delete;
   * docker-compose stop;
   * docker-compose exit;
   * docker compose up.
5. Яка команда дозволяє побудувати Docker-образи?
   * docker-compose build;
   * docker-compose run;
   * docker compose build;
   * docker compose up;
   * docker build all.
6. Що являє собою Docker Compose?
   * інструмент, що дозволяє створювати та управляти множиною Docker-контейнерів, що працюють разом як єдине ціле;
   * інструмент, що дозволяє створювати та управляти множиною Docker-контейнерів, що працюють окремо;
   * інструмент, що дозволяє управляти баз даних створених для Docker-контейнерів;
   * утиліта, що дозволяє поєднувати та дебажити поелементно Docker-контейнери;
   * утиліта, що дозволяє створювати та управляти множиною Docker-контейнерів, що працюють разом як єдине ціле;
7. Які основні поняття Docker Compose?
   * services, networks, volumes, containers;
   * services, networks, tests, containers;
   * services, networks, database, containers;
   * services, networks, database, tests;
   * services, tests, database, networks.
8. Щоб запустити одночасно контейнери використовують…
   * Docker Compose;
   * Docker Consul;
   * Docker Bridge;
   * Docker Logger;
   * Docker Stats.
9. Для зупинки додатку за допомогою Docker Compose потрібно використати команду:
   * docker-compose down;
   * docker compose down;
   * docker compose down -force;
   * docker-compose stop;
   * docker-compose restart.
10. Що відносять до переваг Docker Compose?
    * репродуктивність;
    * відсутність вбудованого механізму;
    * обмежена масштабованість;
    * відсутність вбудованого оркестрування;
    * всі перелічені відповіді правильні.
11. Що відносять до недоліків Docker Compose?
    * відсутність вбудованого механізму оркестрування;
    * масштабованість;
    * репродуктивність;
    * моніторинг;
    * керування.
12. За допомогою якої команди можна перевірити стан контейнерів та сервісів?
    * docker-compose ps;
    * docker-compose scan;
    * docker compose ps;
    * docker-compose test;
    * docker-compose up.
13. В файлі docker-compose.yml можна визначити…
    * мережі та об’єми, які використовуються в додатку;
    * список тестів для запуску;
    * список ресурсів, що не використовує додаток;
    * розмір додатку в МБ при релізі;
    * автора та його лог-файл процесів роботи з контейнером.
14. Який готовий образ використовує сервіс “db” ?
    * PostgreSQL;
    * MongoDB;
    * Cassandra;
    * DB2;
    * FireBird.
15. Який засіб використовують для зберігання та обміну даними між контейнерами в Docker Compose?
    * Volumes;
    * Networks;
    * Databases;
    * Containers.
16. Що називають ізольованими середовищами, в яких працюють запущені додатки?
    * Containers;
    * Volumes;
    * Services;
    * Databases;
    * Configurations.
17. При запуску додатку на локальній машині після створення файла конфігурації на якому з перелічених портів буде доступний сайт?
    * 5000;
    * 8080;
    * 5050;
    * 8800;
    * 8000.
18. Які проблеми при розгортанні додатків дозволяє вирішувати Docker Compose?
    * складність розгортання та налаштування середовища для розробки та тестування;
    * проблеми з управлінням та налаштуванням багатьох Docker-контейнерів;
    * синхронне завантаження баз даних;
    * усунення проблеми наявності Docker на кожній машині, на якій запускається додаток;
    * вбудованого механізму оркестрування контейнерів.
19. Який інструмент можна розгорнути для дуже складних додатків з великою кількістю мікросервісів та контейнерів, якщо потужності Docker Compose буде не вистачати?
    * Kubernetes;
    * перезавантажити Docker Compose;
    * MongoCompose;
    * Neo4j;
    * правильна відповідь відсутня.
20. Який параметр відображається першим у файлі docker-compose.yml:
    * version;
    * web;
    * services;
    * volumes;
    * db.
21. В якому році була вперше представлена Apache Kafka?
    * 2011;
    * 2009;
    * 2010;
    * 2012;
    * 2017.
22. На якій мові програмування написана Apache Kafka?
    * Java;
    * C++;
    * PHP;
    * Assembly;
    * Delphi.
23. В якій компанії була вперше представлена Apache Kafka?
    * LinkedIn;
    * Myco;
    * Opportunity;
    * XING;
    * SkillsNet.
24. Що відноситься до переваг Apache Kafka?
    * висока швідкість;
    * гнучкість;
    * висока доступність;
    * вимоги до розробника;
    * відсутність деяких вбудованих функцій.
25. Який компонент Apache Kafka відповідає за обробку та збереження повідомлень?
    * Broker;
    * Topic;
    * Partition;
    * Consumer Group;
    * Client Libraries.
26. Який з перелічених компонентів Apache Kafka відповідає за реплікацію даних для забезпечення високої продуктивності?
    * Broker;
    * Topic;
    * Partition;
    * Consumer Group;
    * Client Libraries.
27. Яку назву має розподілений службовий вузол. який використовується для керування кластером Apache Kafka?
    * Zookeeper;
    * Client Libraries;
    * Consumers;
    * Partition;
    * Topics.
28. Для чого використовується компонент архітектури Client Libraries в Apache Kafka?
    * для підтримки різноманітних клієнтських бібліотек мов програмування;
    * для керування кластером Apache Kafka;
    * для зчитування повідомлень з топіків;
    * для обробки і збереження звернень користувачів;
    * правильна відповідь відсутня.
29. Який компонент архітектури Apache Kafka споживає (читає) повідомлення з топіків?
    * Consumer;
    * Producer;
    * Broker;
    * Reviewer;
    * Data analyst.
30. Який компонент архітектури Apache Kafka публікує повідомлення в топіки?
    * Producer;
    * Consumer;
    * Broker;
    * Reviewer;
    * Data analyst.
31. Яку назву має логічна одиниця до якої розбиваються топіки в Apache Kafka?
    * Partition;
    * Reason;
    * Point;
    * Post;
    * Zookeeper;
32. Який файл відповідає за конфігурацію Kafka?
    * server.properties;
    * zookeeper.propertie;
    * сonfigurations.yml;
    * setup.yml;
    * properties.yml;
33. Який файл відповідає за конфігурацію Zookeeper?
    * zookeeper.propertie;
    * сonfigurations.yml;
    * setup.yml;
    * properties.yml;
    * config.properties.
34. Для використання Produser потрібно…
    * налаштувати його, вказати необхідні конфігураційні параметри, такі як адреса брокера Kafka, серіалізатор повідомлень, роздільник та інші налаштування;
    * прописати необхідні параметри запуску і аналізу топіків на помилки;
    * налаштувати Client Libraries;
    * підготувати вибірку повідомлень (Topics);
    * ніяких налаштувань не потрібно, він чудово працює з вже встановленими параметрами.
35. Яку модель використовує Apache Kafka?
    * виробник-споживач;
    * споживач-виробник;
    * виробник-споживач-аналітик;
    * споживач-аналітик-виробник;
    * жодна з перелічених відповідей не є вірною.
36. Як Consumer може читати повідомлення в Apache Kafka?
    * асинхронно або синхронно;
    * тільки синхронно;
    * тільки асинхронно;
    * Consumer не читає повідомлення, адже цим займається Producer;
    * правильна відповідь відсутня.
37. Які мови підтримує Client Libraries в Apache Kafka?
    * Java, Python, C++, .NET та інші;
    * тільки Java;
    * тільки Python;
    * всі, окрім С++;
    * тільки С++.
38. Хто в Apache Kafka відповідає за відстеження стану брокерів, топіків, споживачів та груп споживачів?
    * Zookeeper;
    * Consumer;
    * Broker;
    * Producer;
    * QA.
39. Хто в pache Kafka має можливість налаштувати додаткові параметри, такі як автоматичне зберігання відстежувачів (offsets), керування затримкою (backoff) при отриманні повідомлень?
    * Consumer;
    * Zookeeper;
    * Broker;
    * Producer;
    * QA.
40. За допомогою чого забезпечується гарантія надійної доставки повідомлень, що дозволяє впевнитися, що повідомлення будуть доставлені до споживачів без втрати?
    * реплікації та затримки (acknowledgement) від споживачів;
    * метрики моніторингу та інструментів управління;
    * динамічного налаштування групи користувачів;
    * автоматичного збереження відстежувачів та перегляду деталей конфігурації брокерів;
    * асинхронной взаємодії мікросервісів та налаштування топіків тільки для перегляду деталей конфігурації.
41. Elasticsearch – це…
    * система пошуку та аналізу даних, яка дозволяє зберігати логи та здійснювати пошук по ним;
    * інструмент візуалізації даних, який дозволяє відображати графіки логів та створювати панелі управління для аналізу логів;
    * агент збору та передачі логів, який дозволяє збирати логи з контейнерів та інших джерел, перевіряти їх на наявність помилок та передавати до централізованого зберігання;
    * система моніторингу та тривоження, яка дозволяє відстежувати стан додатків на Kubernetes;
    * інструмент, який надає API для керування та моніторингу контейнерних додатків.
42. Kubernetes API використовується для..
    * для керування та моніторингу контейнерних додатків;
    * роботи з об'єктами, такими як Pod, Service, Deployment, ReplicaSet та інші;
    * розміщення одного або кількох контейнерів та спільного віртуального мережевого простору;
    * керування масштабуванням та оновленням додатків;
    * правильна відповідь відсутня.
43. ReplicaSet в Kubernetes використовують для:
    * забезпечення керування кількістю копій Pod, що відповідають заданим параметрам;
    * роботи з об'єктами, такими як Pod, Service, Deployment, ReplicaSet та інші;
    * розміщення одного або кількох контейнерів та спільного віртуального мережевого простору;
    * керування масштабуванням та оновленням додатків;
    * правильна відповідь відсутня.
44. Кожен сервіс в Kubernetes має:
    * власну IP-адресу та порти, що дозволяють додаткам звертатися до нього як до єдиного точки доступу;
    * визначену кількість Pod та може автоматично масштабувати їх кількість відповідно до потреб додатку;
    * визначений життєвий цикл та може бути керований за допомогою API.
    * конфігураційні дані та секретні дані, такі як паролі, ключі доступу та інші;
    * правильна відповідь відсутня.
45. Що є найменшою одиницею розгортання в Kubernetes?
    * Pod;
    * Service;
    * Server;
    * Client;
    * Database.
46. Що дозволяє зберігати конфігураційні та секретні дані в Kubernetes?
    * ConfigMaps та Secrets;
    * Deployment;
    * Service;
    * Pod;
    * АРІ.
47. За допомогою чого в Kubernetes можна керувати масшатуванням та оновленням додатків?
    * Deployment;
    * Service;
    * ConfigMaps та Secrets;
    * Server;
    * API.
48. Який компонент Kubernetes містить в собі код додатку та всі його компоненти?
    * Docker Image;
    * Ingress;
    * Deployment;
    * Service;
    * Volumers.
49. Який механізм Kubernetes дозволяє забезпечити зовнішній доступ до додатку, використовуючи правила маршрутизації?
    * Ingress;
    * Deployment;
    * Docker Image;
    * Service;
    * Volumers.
50. Який механізм Kubernetes дозволяє забезпечити доступ до запущених копій додатку, незалежно від того, на якому вузлі вони запущені?
    * Service;
    * Deployment;
    * Docker Image;
    * ConfigMaps та Secrets;
    * Volumers.
51. Який компонент розгортання Kubernetes описує кількість копій додатку, що мають бути запущені, та спосіб їх розгортання?
    * Deployment;
    * Ingress;
    * ReplicaSet;
    * АРІ;
    * Controllers.
52. Що відповідає за керування життєвим циклом об’єктів Kubernetes?
    * Controllers;
    * ReplicaSet;
    * Volumes;
    * Docker Image;
    * Ingress.
53. Що таке Grafana?
    * інструмент візуалізації даних, який дозволяє відображати графіки метрик, створювати панелі управління та дашборди для моніторингу додатків;
    * інструмент візуалізації даних, який дозволяє відображати графіки логів та створювати панелі управління для аналізу логів;
    * процес управління та координації додатків, що працюють у контейнерах, з використанням Kubernetes;
    * об'єкт, який дозволяє розгортати та оновлювати контейнеризовані додатки;
    * правильна відповідь відсутня.
54. Як в Kubernetes можна підготувати Docker Image образ контейнеру додатку?
    * вручну, або автоматично за допомогою CI/CD інструментів;
    * тільки вручну;
    * за допомогою CI/CD інструментів;
    * створити декілька копій додатку;
    * правильна відповідь відсутня.
55. Zabbix та Nagios це –
    * системи моніторингу, які дозволяють відстежувати стан додатків на Kubernetes та сповіщати про проблеми в реальному часі;
    * набір інструментів, який включає Elasticsearch, Logstash та Kibana;
    * інструменти для забезпечення додатків за допомогою різних механізмів;
    * інструменти для оновлення додатків без перерв у роботі;
    * механізми для підготовки Docker Image образу контейнера додатку.
56. Для чого використовують Fluentd в Kubernetes?
    * для збору та передачі логів;
    * для віддаленого оновлення додатків;
    * для розгортання образу контейнера;
    * для візуалізації даних, зображення графіків роботи БД, моніторингу активності користувачів;
    * правильна відповідь відсутня.
57. У вигляді чого ConfigMap дозволяє зберігати конфігураційні дані, які використовують у додатку?
    * пар ключ-значення;
    * шифрованих логінів і паролей;
    * кластерів конфігурацій;
    * зберігаються в окремих мікросервісах;
    * правильна відповідь відсутня.
58. Який перший етап оркестрації сервесів на Kubernetes?
    * створення Docker-образів;
    * створення Kubernetes об'єктів;
    * стеження за станом додатку;
    * оновлення та ролбек додатку;
    * керування конфігурацією.
59. Helm Chart - це:
    * пакет, який містить у собі всі необхідні компоненти та налаштування для розгортання складного додатку на Kubernetes;
    * процес управління та координації додатків, що працюють у контейнерах, з використанням Kubernetes;
    * механізм, який дозволяє забезпечити зовнішній доступ до додатку, використовуючи правила маршрутизації;
    * механізм, який дозволяє забезпечити доступ до запущених копій додатку, незалежно від того, на якому вузлі вони запущені;
    * об'єкт, який визначає кількість копій додатку, що мають бути запущені, та спосіб їх розгортання.
60. Elastic Stack - це:
    * набір інструментів, який включає Elasticsearch, Logstash та Kibana, що дозволяє забезпечити повний стек моніторингу та логування для Kubernetes;
    * агент збору та передачі логів, який дозволяє збирати логи з контейнерів та інших джерел, перевіряти їх на наявність помилок та передавати до централізованого зберігання, наприклад, до Elasticsearch;
    * механізм, який дозволяє забезпечити доступ до запущених копій додатку, незалежно від того, на якому вузлі вони запущені;
    * об'єкт, який визначає кількість копій додатку, що мають бути запущені, та спосіб їх розгортання;
    * інструмент управління та координації додатків, що працюють у контейнерах, з використанням Kubernetes;