1. Что подразумевает понятие Управление данными?
2. Набор компьютерных технологий, связанных с защитой данных
3. Набор компьютерных технологий, связанных с передачей данных
4. Набор компьютерных технологий, связанных с надежным хранением, извлечением и обработкой данных
5. Набор компьютерных технологий, связанных с созданием данных
6. Что такое плоский файл?
7. Это файл, имеющий размер не более 1Мб
8. Это именованный набор данных на внутреннем носителе
9. Это файл прошедший процедуру сжатия
10. Это именованный набор данных на внешнем носителе
11. Для ОС плоский файл это?
12. Набор инструкций
13. Набор словесно-числовой информации
14. Набор байт
15. Набор команд
16. Сериализация это?
17. Процесс сохранения состояния объекта в последовательность бит
18. Процесс сохранения состояния объекта в последовательность байт
19. Процесс сохранения состояния объекта в последовательность команд
20. Процесс сохранения состояния объекта в последовательность словесно-числовой информации
21. Десериализация это?
22. Процесс восстановления объекта из последовательности байт
23. Процесс восстановления объекта из последовательности бит
24. Процесс восстановления объекта из последовательности команд
25. Процесс восстановления объекта из словесно-числовой последовательности
26. Как расшифровывается XML?
27. Extensible Markup Leisure
28. Extensible Markup Language
29. Extensible Meantime Language
30. Experiment Markup Language
31. Что такое SQLite?
32. Библиотека, позволяющая осуществлять взаимодействие с SOAP службами
33. Компактная встраиваемая реляционная база данных
34. Библиотека, содержащая команды для обработки ошибок в приложении
35. Механизм работы с уведомлениями в Android
36. Что такое SOA?
37. Подход к разработке программного обеспечения, предполагающий разработку приложения как единое целое
38. Модульный подход к разработке программного обеспечения, основанный на использовании распределённых, сильно связанных незаменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам
39. Модульный подход к разработке программного обеспечения, основанный на использовании распределённых, сильно связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам
40. Модульный подход к разработке программного обеспечения, основанный на использовании распределённых, слабо связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам
41. Что такое SOAP?
42. Протокол для обмена данными в виде потока бит в распределенной среде
43. Протокол для обмена структурированными данными в виде последовательности команд в распределенной среде
44. Протокол для обмена структурированными сообщениями в распределённой среде
45. Протокол для обмена структурированными сообщениями в формате JSON в распределенной среде
46. Выберите неверное утверждение об REST
47. Как правило используется для построения веб служб
48. Каждая единица информации определяется глобальным идентификатором (URI)
49. REST представляет собой не согласованный набор ограничений, учитываемых при проектировании распределённой гипермедиа-системы
50. URI имеет строго заданный формат
51. Какой класс находится наверху иерархии всех классов?
52. Object
53. View
54. Activity
55. Service
56. Выберите верное определение View
57. Базовый класс для всех классов в Android
58. Базовый класс для всех виджетов пользовательского интерфейса
59. Базовый класс для работы с потоками
60. Базовый класс для работы с файлами
61. Выберите неверное утверждение про Service
62. Service выполняется в фоновом режиме
63. Service нельзя остановить принудительно
64. Service имеет жизненный цикл
65. Может быть запущен приложением по условию
66. В каком состоянии находится Activity, если интерфейс пользователя невидим
67. Активно и выполняется
68. Приостановлено
69. Возобновлено
70. Завершено
71. Какой формат файла имеет Манифест Android
72. JSON
73. TXT
74. XML
75. JAVA
76. Выберите неверное утверждение про Манифест
77. Определяет имя Java-пакета приложения - уникальный идентификатор для приложения
78. Объявляет разрешения
79. Объявляет минимальный уровень Android API
80. Содержит список всех классов в приложении
81. Зачем используется класс R?
82. Содержит ссылки на ресурсы в приложении
83. Содержит ссылки на классы в приложении
84. Содержит ссылки на интерфейсы в приложении
85. Содержит ссылки на методы в приложении
86. Сколько основных типов памяти есть в Android
87. 1
88. 2
89. 3
90. 4
91. Что такое RAM?
92. Постоянная память, ПЗУ
93. Оперативная память, ОЗУ
94. Внутренняя память
95. Внешняя память
96. Какая память является энергозависимой в Android?
97. Internal
98. RAM
99. ROM
100. External
101. Как проверить возможность чтения/записи в External память
102. MEDIA\_WRITED
103. MEDIA\_READED
104. MEDIA\_MOUNTED
105. MEDIA\_SHARED
106. Что такое публичные файлы (Public files)?
107. Файлы доступные в рамках только одного приложения
108. Файлы доступные только системе Android
109. Файлы, которые должны быть доступны для других приложений и пользователя
110. Файлы, которые должны быть доступны только приложениям
111. Выберите верное утверждение про частные файлы (Private files)
112. Файлы, доступ к которым может получить только пользователь
113. Файлы, доступ к которым может получить только система Android
114. Файлы, которые удаляются автоматически при закрытии приложения пользователем
115. Файлы, которые удаляются автоматически при удалении приложения с телефона
116. Выберите класс который не используется для работы с файлами
117. android.content.Context
118. java.io.File
119. java.lang.Object
120. java.lang.Buffer
121. Какой метод позволяет получить ссылку на подкаталог в каталоге приложения
122. getExternalFilesDir()
123. fileList()
124. getDir()
125. getFileInput()
126. Какой метод позволяет вернуть File, представляющий собой каталог для размещения файлов определенного типа
127. getFileStreamPath()
128. getExternalFilesDir()
129. Environment.getExternalStorageState()
130. Environment.getExternalStoragePublicDirectory()
131. Выберите неверное утверждение
132. Операции ввода/вывода по сравнению с операциями в оперативной памяти выполняются медленно
133. Для хранения часто используемых в приложении данных, рекомендуется использовать постоянную память
134. В оперативной памяти выделяется промежуточная область - буфер, в которой постепенно накапливается информация
135. Когда буфер заполнен, его содержимое однократно переносится на диск и буфер очищается
136. Какой метод следует использовать для перемещения к определенной позиции и изменения хранящегося там значения в файле?
137. seek()
138. move()
139. moveToPosition()
140. transfer()
141. Выберите верное утверждение об java.io.FileReader
142. Имеет несколько конструкторов
143. Используется для чтения и записи текстовых данных в файл
144. Производный от абстрактного класса File
145. Производный от абстрактного класса Reader
146. Какой метод следует использовать для создания пустого файла в каталоге, используя данные префикса и суффикса строки для его имени?
147. createNewFile()
148. createFile()
149. mkdir()
150. createTempFile()
151. Как получить внешний каталог для private файлов?
152. super.getExternalFilesDir(Enviroment.DIRECTORY\_DOCUMENTS)
153. super.getFilesDir(Enviroment.DIRECTORY\_FILES)
154. super.getDir(Enviroment.DIRECTORY\_PRIVATE)
155. super.getExternalPrivateDir(Enviroment.PRIVATE\_FILES)
156. При недостатке internal-памяти, кэш-файлы могут быть
157. Удалены с уведомлением пользователя
158. Удалены без предупреждения
159. Удалены с сохранением резервной копии
160. Не будут удалены
161. Выберите неверное определение
162. getName() – возвращает имя файла
163. getParent() – возвращает имя родительского каталога
164. getAbsolutePath() – возвращает абсолютный путь файла, начиная с корня
165. getTreeParent() – возвращает имя верхнего элемента в дереве родительского каталога
166. Каким знаком отделяются пары ключ значение в JSON
167. .
168. ;
169. ,
170. :
171. Как называется библиотека для работы с JSON в Android
172. Gson
173. JsonS
174. Gcon
175. Gjson
176. Из предложенных вариантов выберите строку с основными методами в библиотеке Google для работы с JSON
177. parseJson()
178. parseFromJson() и parseToJson()
179. toJson() и fromJson()
180. readJson() и writeJson()
181. Что следует использовать для разработки собственного сериализатора в GSON
182. GsonMaker
183. GsonBuilder
184. GsonCompressor
185. GsonPuller
186. Что такое XML Schema?
187. Язык программирования
188. Язык для описания структуры XML-программа
189. Язык для описания структуры XML-документа
190. Язык описания структуры JSON-документа
191. Для проверки правильности XML-документа используется
192. XSLT
193. XQuery
194. XML-Schema
195. Xpath
196. На каком языке пишутся правила выбора и преобразования данных при использовании XSLT?
197. Язык запросов XML
198. Язык запросов XQuery
199. Язык запросов XData
200. Язык запросов XPath
201. Чем занимается организация OASIS?
202. Управляет процессом добавления стандартов RFC
203. Управляет разработкой и внедрением программного обеспечения для протокола HTTP
204. Управляет разработкой и принятием промышленных стандартов электронной коммерции
205. Управляет разработкой стандартов и последующим внедрением их в SQL
206. Как называется правильно построенный документ, соответствующий синтаксическим правилам XML?
207. Валидный
208. Коллоидный
209. Верифицированный
210. Стандартизированный
211. Какое из приведенных синтаксических правил XML является необязательным?
212. Наличие корневого элемента
213. Наличие атрибута
214. Древовидная структура
215. Каждый открывающий тег имеет соответствующий закрывающий тег
216. За что отвечает ancestor в XPath?
217. Содержит родительский узел и все выше стоящие родительские узлы вплоть до корневого элемента
218. Содержит атрибуты контекстного узла, если контекстный узел – узел элемента
219. Содержит родительский узел контекстного узла. Корневой элемент не имеет родителя.
220. Содержит пространство имен контекстного узла, до тех пор, пока контекстный узел является элементом
221. За что отвечает ancestor-or-self в XPath?
222. Содержит родительский узел контекстного узла. Корневой элемент не имеет родителя
223. Содержит пространство имен контекстного узла, до тех пор, пока контекстный узел является элементом
224. Содержит узлы-предки вместе с самим контекстным узлом, вплоть до корневого узла
225. Содержит сам контекстный узел и все его дочерние и т.д. узлы.
226. Какие классы нужно добавить для работы с SQLite в Android?
227. SQLite и SQLiteDatabase
228. SQLiteOpenHelper и SQLiteDatabase
229. SQLiteOpenHelper и SQLite
230. SQL и SQLiteDatabase
231. Какие обязательные методы необходимо реализовать в SQLiteOpenHelper?
232. onCreate() и onStart()
233. onStart() и onStop()
234. onStart() и onUpgrade()
235. onCreate и onUpgrade()
236. Какой метод в Android используется для чтения данных из SQLite?
237. select()
238. execSQL()
239. query()
240. exec()
241. Какое количество параметров передается в query при запросе?
242. 2
243. 4
244. 5
245. 7
246. Сколько классов хранения в SQLite?
247. 1
248. 3
249. 5
250. 7
251. В каком виде представляются значения в классе хранения BLOB?
252. Значение представляет собой блок данных, который хранится точно так же, как он был введен.
253. Значение представляет собой текстовую строку, хранящуюся с использованием кодировки базы данных (UTF-8, UTF-16BE или UTF-16LE)
254. Значение представляет собой значение с плавающей запятой, которое хранится как 8-байтовое число с плавающей точкой IEEE.
255. Значение представляет собой целое число со знаком, сохраненное в 1, 2, 3, 4, 6 или 8 байтах в зависимости от величины значения.
256. Что такое аффинированные типы?
257. Тип, который является слиянием нескольких типов данных и позволяет хранить их значение
258. Тип, который используется для хранения больших текстовых данных
259. Тип, который является рекомендуемым для сохраняемых в столбце значений
260. Тип, который является рекомендуемым для больших числовых данных
261. Когда и как определяется аффинированность столбца?
262. Явно после создания таблицы
263. Явно во время создания таблицы
264. Неявно после создания таблицы
265. Неявно во время создания таблицы
266. Какой тип в SQLite используется для хранения даты/времени?
267. Date
268. DateTime
269. Time
270. В SQLite нет отдельного типа данных для хранения даты/времени
271. Значения с классом NULL
272. Считаются меньше любого другого значения, даже другого значения NULL
273. Считаются больше любого другого значения, даже другого значения NULL
274. Считаются равными любому другому значению, даже другому значению NULL
275. Считаются не равными любому другому значению, даже другому значению NULL
276. Значение INTEGER или REAL
277. Считается меньше, чем значение TEXT или BLOB
278. считается больше, чем значение TEXT или BLOB
279. считается равным значению TEXT или BLOB
280. считается не равным значению TEXT или BLOB
281. Значение TEXT
282. Равно значению BLOB
283. Не равно значению BLOB
284. Больше значения BLOB
285. Меньше значения BLOB
286. Какую функцию следует использовать для сравнения значений BLOB
287. compare()
288. cmp()
289. memcmp()
290. compareTo()
291. Для чего используется ContentValues?
292. Для удаления строк из таблицы
293. Для получения значений строк из таблицы
294. Для добавления новых строк в таблицу
295. Для обновления строк в таблице
296. Что возвращает метод insert в случае успешного добавления записи?
297. 1
298. 0
299. Идентификатор \_id вставленной строки
300. Идентификатор \_id предыдущей строки
301. Выберите правильный порядок выполнения операторов SQL
302. FROM WHERE GROUP BY HAVING SELECT DISTINCT ORDER BY LIMIT
303. FROM WHERE HAVING GROUP BY SELECT DISTINCT ORDER BY LIMIT
304. SELECT DISTINCT FROM WHERE HAVING GROUP BY OREDER BY LIMIT
305. SELECT DISTINCT FROM WHERE GROUP BY HAVING ORDER BY LIMIT
306. Зачем нужен Cursor?
307. Обеспечивает перебор строк в файле
308. Обеспечивает доступ к набору записей
309. Обеспечивает навигацию в коллекциях
310. Обеспечивает работу с перечислением
311. В чем отличие query() от rawQuery() ?
312. rawQuery принимает в качестве параметра коллекции
313. rawQuery позволяет при помощи параметров составить SQL запрос
314. rawQuery позволяет сформировать SQL запрос из строки
315. rawQuery позволяет удалить все записи из таблицы
316. Выберите неверную внутреннюю таблицу из приведенных
317. SQLITE\_MASTER
318. SQLITE\_SEQUENCE
319. SQLITE\_STAT
320. SQLITE\_DATABASE
321. Что такое синтаксическая диаграмма (Railroad diagram)?
322. Граф с одним входным ребром и одним выходным ребром и помеченными вершинами
323. Граф с несколькими входными ребрами и одним выходным ребром и помеченными вершинами
324. Граф с несколькими входными ребрами и несколькими выходными ребрами и помеченными вершинами
325. Граф с одним входным ребром и несколькими выходными ребрами и помеченными вершинами
326. Какое имя зарезервировано в SQLite?
327. sql\_
328. sqlite\_
329. master\_
330. system\_
331. При помощи какого оператора удаляются таблицы?
332. delete
333. drop
334. downgrade
335. clear
336. С одинаковыми именами можно создать
337. Таблицу
338. Представление
339. Индекс
340. Триггер
341. Что такое ROWID?
342. Уникальный номер записи в каждой таблице
343. Уникальный номер базы данных
344. Уникальный номер таблицы в базе данных
345. Уникальный номер версии базы данных
346. Выберите неверное утверждение о PRIMARY KEY?
347. UNIQUE + NOT NULL
348. По умолчанию автоинкримент на единицу для целочисленного ключа
349. Значения ключа не совпадают с ROWID
350. Может быть составным
351. Выберите неверное утверждение об WITHOUT ROWID?
352. Можно создать таблицу с опцией WITHOUT ROWID
353. Каждая таблица WITHOUT ROWID не обязана иметь PRIMARY KEY
354. Нет механизма автоинкримента
355. Используется для оптимизации хранения и обработки
356. Выберите неверное утверждение об InsertOrThrow
357. 0 – если нет ошибки и WITHOUT ROWID
358. ROWID – если нет ошибки
359. Исключение – если есть ошибка
360. 1 – успешное выполнение
361. Зачем нужен ON CONFLICT?
362. Метод разрешения конфликтов при вставке
363. Метод генерации конфликта для проверки правильности работы
364. Метод разрешения конфликтов при удалении
365. Метод разрешения конфликтов при переименовании столбцов таблицы
366. Выберите верные варианты алгоритма ON CONFLICT
367. ROLLBACK ABORT FAIL CHECK REPLACE
368. ROLLBACK COMMIT FAIL REPLACE IGNORE
369. ROLLBACK ABORT FAIL IGNORE REPLACE
370. ROLLBACK DROP FAIL REPLACE IGNORE
371. Зачем нужен ROLLBACK?
372. Генерирует ошибку и выполняет ROLLBACK для всей транзакции
373. Генерирует ошибку и выполняет ROLLBACK только для текущего оператора
374. Генерирует ошибку и выполняет ROLLBACK только для последней операции в рамках оператора
375. Выполняет сброс текущей операции, оператор продолжает свою работу
376. Зачем нужен FAIL?
377. Генерирует ошибку и выполняет ROLLBACK для всей транзакции
378. Генерирует ошибку и выполняет ROLLBACK только для текущего оператора
379. Выполняет сброс текущей операции, оператор продолжает свою работу
380. Генерирует ошибку и выполняет ROLLBACK только для последней операции в рамках оператора
381. Что позволяет COLLATE?
382. Позволяет задать способ обработки данных
383. Позволяет задать способ удаления данных
384. Позволяет задать способ сравнения данных
385. Позволяет задать способ добавления данных
386. Как производится сравнение COLLATE при использовании BINARY?
387. Побайтовое сопоставление строк данных, кроме 26-ти прописных букв ASCII, которые перед сортировкой переводятся в свои эквиваленты в нижнем регистре
388. Побайтовое сопоставление строк данных независимо от кодировки текста
389. Побайтовое сопоставление строк данных, но игнорируются пробелы в конце строки
390. Побайтовое сопоставление строк данных, но игнорируются символы запятой и точки запятой
391. Выберите верное утверждение о COLLATE
392. Каждый столбец каждой таблицы имеет связанную с ним функцию сортировки
393. Если функция определена явно, то по умолчанию используется BINARY
394. Нельзя сравнивать строки в разных collation
395. Рекомендуется в большинстве случаев использовать NOCASE
396. При помощи какого ключевого слова указывается таблица и столбец, на который ссылается внешний ключ?
397. REFERENCE
398. CONSTRAINT
399. DEPENDENCE
400. LINK
401. Что такое PRAGMA в SQLite?
402. Специфичный для SQLite компонент используемый для отправки запросов к библиотеке SQLite для получения табличных
403. Специфичный для SQLite компонент используемый для блокирования операций библиотеки SQLite и отправки запросов к библиотеке SQLite для получения внутренних (не табличных) данных
404. Специфичный для SQLite компонент используемый для изменения операций библиотеки SQLite и отправки запросов к библиотеке SQLite для получения внутренних (не табличных) данных
405. Специфичный для SQLite компонент используемый для ускорения отправки запросов к библиотеке SQLite для получения внутренних табличных данных
406. Выберите неверное утверждение. PRAGMA-команды по типу выполняемых действий делятся на следующие:
407. Получающие данные о схеме базы данных (информацию о таблицах, индексах, внешних ключах и т.д.).
408. Изменяющие различные параметры (кодировка, синхронизация и т.д.).
409. Изменяющие внутреннюю версию схем (schema\_version и user\_version)
410. Редактирующие внутренние таблицы (SQLITE\_STAT, SQLITE\_MASTER и т.д.).
411. Что выполняет RESTRICT?
412. явное указание на запрет изменять или удалять PRIMARY KEY значение, если на него ссылаются FOREIGN KEY – действует по умолчанию
413. явное указание на то, что никакие не предпринимать при изменении или удалении PRIMARY KEY значения (фактически отсутствует ограничение)
414. явное указание на то, что удалить PRIMARY KEY значения при их изменении
415. явное указание на то, что обновить PRIMARY KEY значения при их изменении (фактически отсутствует ограничение)
416. Что такое DEFERABLE?
417. Отложенная проверка соответствия
418. Отложенная проверка ограничения
419. Отложенная операция модификации таблицы
420. Установление ограничения на таблицу после операции COMMIT
421. Когда используется MATCH FULL?
422. Если одно из значений составного ключа NULL, то проверка на ограничение не проверяется
423. Если допускаются (проверка не осуществляется) составные ключи, состоящие полностью из NULL
424. Если в составном FOREGN KEY есть NULL-значение, то должен существовать хотя бы один родительский ключ у которого есть значение для NULL-позиции FOREGN KEY – ключа
425. Если несколько значений составного ключа NULL
426. Какой из приведенных операторов сравнения лишний?
427. LIKE
428. GLOB
429. BETWEEN
430. REGEXP
431. Выберите верное утверждение при работе с датами SQLite
432. Временные зоны поддерживаются
433. Смещение на летнее/зимнее время не поддерживается
434. Поддерживается формат UTC – Universal Coordinated Time – практически идентичен времени по Цельсию
435. Поддерживается формат UCTZ – Universal Coordinated Time Zone – практически идентичен времени по Гринвичу
436. Выберите верное утверждение
437. unixepoch – показывает, что время в UNIX формате
438. unixepoch – показывает, что время в POSIX формате
439. unixepoch – показывает, что время в UTC формате
440. unixepoch – показывает, что время в UCTZ формате
441. При работе с датами что означает параметр %m
442. month (*MM*), 01-12
443. minute (*MM*), 00-59
444. millisecond(MM), 0-1000
445. monday
446. Что такое подзапрос?
447. Запрос, вызываемый после выполнения родительского запроса
448. Запрос, вызываемый перед выполнением родительского запроса
449. Запрос, вложенный в другой запрос
450. Запрос, выполняющийся в случае возникновения ошибки
451. Из приведенных вариантов, выберите не существующий JOIN
452. INNER JOIN
453. CROSS JOIN
454. SELF JOIN
455. UNION JOIN
456. Что означает INTERSECT объединение таблиц?
457. Выбираются только общие данные из обоих таблиц
458. Выбираются только уникальные данные первой таблицы, а общие данные и данные второй таблицы отбрасываются
459. Выбираются данные обоих таблиц, кроме повторяющихся значений
460. Выбираются данные обоих таблиц, включая повторяющиеся значения
461. Что такое обобщенные табличные выражения (CTE)?
462. временные результирующие наборы, определенные в области выполнения единичных инструкций FROM
463. временные результирующие наборы, определенные в области выполнения единичных инструкций SELECT
464. временные результирующие наборы, определенные в области выполнения единичных инструкций DROP
465. временные результирующие наборы, определенные в области выполнения единичных инструкций INSERT
466. Выберите верное утверждение о Common Table Expressions
467. Cохраняются в базе данных в виде объектов
468. Время жизни ограничено продолжительностью запроса
469. Не могут ссылаться сами на себя
470. Один и тот же запрос не может ссылаться на CTE несколько раз
471. Выберите не верное утверждение о Common Table Expressions
472. Создание рекурсивных запросов
473. Не содержат многократных ссылок на результирующую таблицу из одной и той же инструкции
474. Заменяют представления
475. Происходит группирование по столбцу, производного от скалярного подзапроса выборки или функции, которая не детерминирована
476. Какую функцию следует использовать в SQLite для получения данных в виде даты?
477. datetime()
478. date()
479. strftime()
480. ftime()
481. Что такое представление?
482. Поименованный INSERT запрос
483. Поименованный UPDATE запрос
484. Поименованный SELECT запрос
485. Поименованный DELETE запрос
486. Что такое виртуальная таблица?
487. Таблица, которая удаляется после выхода пользователя из системы
488. Структурированный набор данных для которого реализованы все основные методы взаимодействия с таблицей
489. Механизм, позволяющий создать таблицу, взаимодействие с которой будет осуществляться при помощи стандартных команд Windows
490. Таблица, которая используется для тестирования скорости операций удаления данных
491. Выберите неверное утверждение
492. Можно создать триггер на виртуальной таблице
493. Нельзя создавать дополнительные индексы на виртуальной таблице, используя инструкции CREATE INDEX
494. Виртуальные таблицы могут иметь индексы, но они должны быть встроены в реализацию виртуальной таблицы
495. Нельзя выполнить ALTER TABLE ... ADD COLUMN
496. Выберите верное утверждение. Применять виртуальные таблицы можно для:
497. Полноцифрового поиска
498. Небесных индексов
499. Просмотра статистики в файле базы данных
500. Доступ к оперативной памяти компьютера, как если бы это была таблица базы данных
501. Выберите верный порядок столбцов во внутренней таблице SQLITE\_MASTER
502. type name tbl\_name rootpage sql
503. type tbl\_name name rootpage sql
504. type rootpage name tbl\_name sql
505. type sql tbl\_name name rootpage
506. Для чего используется внутренняя таблица SQLITE\_SEQUENCE
507. Реализация PK
508. Реализация NOT NULL
509. Реализация AUTOINCREMENT
510. Реализация PRAGMA
511. Для чего используется внутренняя таблица SQLITE\_STAT?
512. Таблица для реализации алгоритмов шифрования
513. Таблица для сбора статистики
514. Таблица для хранения часто используемых объектов
515. Таблица хранящая данные последней удаленной таблицы
516. На каких характеристиках запроса оценивается эффективность запроса?
517. Селективность Плотность Многопоточность
518. Селективность Трансгрессия Многопоточность
519. Селективность Распределение значений Неделимость
520. Селективность Плотность Распределение значений
521. Что такое селективность?
522. Отношение общего числа записей к числу выбираемых записей
523. Отношение числа выбираемых синтаксически верных записей к общему числу записей
524. Отношение числа выбираемых записей к общему числу записей
525. Отношение числа выбираемых некорректных записей к общему числу записей
526. Что означает плотность?
527. Отношение общего числа записей к числу выбираемых записей
528. Отношение числа дубликатов значений к общему числу значений
529. Отношение числа выбираемых записей к общему числу записей
530. Отношение числа выбираемых корректных записей к общему числу записей
531. Выберите неверное утверждение о поиске и сортировке в запросах SQLite?
532. Поиск по RowId
533. Поиск по индексу (поиск RowId, переход по RowId)
534. Поиск по триггеру (поиск RowId, переход по RowId)
535. Поиск по нескольким условия (И)
536. Выберите верное именование индекса
537. Стандартный префикс index\_ + имя таблицы+ имя столбца
538. Стандартный префикс indx\_ + имя таблицы+ имя столбца
539. Стандартный префикс idx\_ + имя таблицы+ имя столбца
540. Имя таблицы+ имя столбца
541. Выберите неверное утверждение
542. Индекс может строится на столбцах нескольких таблиц
543. На представлении индекс построить нельзя
544. На виртуальной таблице индекс построить нельзя
545. Количество столбцов в индексе ограничено
546. Какие из приведенных компонентов, использующихся только для оперативного анализа неверен?
547. ANALYZE
548. EXPLAIN
549. EXPLAIN QUERY PLAN
550. ANALYZE PLAN
551. Выберите неверное утверждение об ANALYZE
552. Сбор статистики о таблицах и индексах
553. Хранится в sqlite\_stat
554. Оптимизатор запросов может получить доступ к информации и использовать ее, чтобы помочь улучшить выбор планирования запросов
555. Хранится в sqlite\_master
556. Что такое триггер?
557. Это особая разновидность хранимых процедур, исполняемых в ответ на какое-то событие
558. Это особая разновидность функций, исполняемых в ответ на какое-то событие
559. Это особая разновидность представлений, исполняемых в ответ на какое-то событие
560. Это особая разновидность индексов, исполняемых в ответ на какое-то событие
561. Для чего используется триггер? Выберите неверное утверждение
562. Обеспечение целостности данных
563. Каскадной вставки/обновления данных
564. Аудита изменений
565. Реализации сложной бизнес-логики
566. По каким группам событий классифицируется триггер?
567. DDL DCL
568. DML TCL
569. DML DDL
570. TCL DDL
571. Как классифицируются триггеры DML по моменту выполнения относительно события? Выберите неверное определение
572. BEFORE
573. AFTER
574. INSTEAD
575. INSTEAD OF
576. Выберите верную классификацию триггера DML по отношению к объекту выполнения
577. Операторный и строчный
578. Многооператорный и строчный
579. Многооператорный и многострочный
580. Операторный и многострочный
581. Выберите не верное утверждения про вложенные и рекурсивные триггеры
582. Вложенный триггер возникает при выполнении триггером действия, вызывающим другой триггер
583. 32 уровня вложенности
584. Прямая и косвенная рекурсии
585. Рекурсивный триггер вызывает сам себя только после выполнения триггером действия, вызывающем другой триггер
586. Какая инструкция используется для возврата произвольного сообщения об ошибке?
587. ERROR
588. RAISERROR
589. SQLERROR
590. THROW\_ERROR
591. Выберите не верное утверждение про фразу WHEN в триггерах SQLite?
592. Можно использовать фразу WHEN для уточнения условия, при котором триггер выполнится
593. Условие WHEN интерпретируется как булево
594. При WHEN операторы выполняются только для строк, для которых предложение WHEN истинно
595. Могут применяться OLD и NEW префиксы
596. Укажите верное ограничение триггера в SQLite
597. При выполнении операции INSERT можно добавлять DEFAULT значения
598. Можно использовать INDEX BY и NOT INDEXED в командах UPDATE и DELETE
599. Нельзя использовать LIMIT и ORDER BY с командами UPDATE и DELETE в теле триггера
600. Можно использовать CTE в теле триггера
601. Как называется специальная функция обработки ошибок в триггере?
602. ERROR
603. RAISERROR
604. THROW
605. RAISE
606. В каком порядке по умолчанию выполняются триггеры?
607. По возрастанию
608. По убыванию
609. Неопределенный порядок
610. По приоритету
611. При удалении таблицы все связанные триггеры
612. Удаляются
613. Обновляются
614. Не изменяются
615. Будет выдано сообщение об ошибке, произойдет откат операции
616. Если удалены таблица, на которую есть ссылка в теле триггера то
617. Триггер также удалится
618. Отобразится ошибка
619. Ничего не произойдет
620. Установится значение NULL
621. Какое количество аргументов у RAISE()?
622. 1
623. 2
624. 3
625. 4
626. Сколько существует агрегатных функций?
627. 3
628. 4
629. 5
630. 6
631. Из приведенных ниже функций ранжирования выберите лишнее
632. RANK
633. DENSE\_RANK
634. ROW\_NUMBER
635. COLUMN\_NUMBER
636. Из приведенных ниже функций сдвига выберите верную
637. LOG
638. LEAD
639. ROW\_VALUE
640. CUME\_DIST
641. Из приведенных аналитических функций выберите лишнюю
642. PERCENT\_RANK
643. DENSE\_DIST
644. PERCENTILE\_CONT
645. PERSENTILE\_DISC
646. Что такое процентиль?
647. Мера, в которой процентное значение общих значений больше этой мере или меньше ее
648. Мера, в которой процентное значение общих значений равно этой мере или меньше ее
649. Мера, в которой процентное значение общих значений больше этой меры
650. Мера, в которой процентное значение общих значений меньше этой меры
651. Что такое JDBC?
652. JDBC – платформенно независимый промышленный стандарт взаимодействия Java-приложений с различными СУБД, реализованный в виде пакета java.sql, входящего в состав Java SE.
653. JDBC – библиотека взаимодействия Java-приложений с различными СУБД, реализованный в виде пакета java.sql, входящего в состав Java SE.
654. JDBC – платформенно независимый промышленный стандарт взаимодействия Java-приложений с MS SQL Server и SQLite, реализованный в виде пакета java.sql, входящего в состав Java SE.
655. JDBC – платформенно независимый промышленный стандарт взаимодействия любых приложений с различными СУБД, реализованный в виде пакета java.sql, входящего в состав Java SE.
656. Какой объект в JDBC предоставляет результат выполнения запроса?
657. Result
658. ResultSet
659. OutPut
660. OutCome
661. Как нумеруются столбцы в запросе к БД при использовании JDBC?
662. С 0
663. Неизвестно
664. Пользователь сам указывает нумерацию, по умолчанию с 0
665. С 1
666. Что в JDBC используется для вызова процедур?
667. CallProcedure
668. CallableProcedure
669. CallableStatement
670. PrepareCall
671. Какие из приведенных облачных услуг предоставляет Azure?
672. PaaS IaaS SaaS DBaaS HWaaS
673. PaaS MWaaS IaaS SaaS DBaaS
674. PaaS DBaaS MWaaS IaaS CaaS
675. PaaS DBaaS DWaaS IaaS CaaS
676. Выберите неверное утверждение об Windows Azure SQL Databases
677. Использует T-SQL в качестве языка запросов
678. В качестве протокола для доступа к сервису через Интернет используется Tabular Data Stream(TDS)
679. По протоколу HTTP SOAP доступ не предоставляется
680. Microsoft рекомендует использовать ADO.NET Data Services для передачи данных и создания сервисов
681. Что такое оконные функции?
682. Функции, которые позволяют осуществлять вычисления в заданном диапазоне строк внутри одного предложения SELECT
683. Функции, которые используются вместо SELECT
684. Функции, которые позволяют осуществлять вычисления в заданном диапазоне строк внутри одного предложения INSERT
685. Функции, которые позволяют осуществлять вычисления в заданном диапазоне строк внутри одного предложения UPDATE
686. Допускается ли применение триггеров для представлений SQLite?
687. Да, допускается. Триггер INSTEAD OF применятся к представлениям
688. Да, допускается. Триггер BEFORE применятся к представлениям
689. Да, допускается. Триггер AFTER применятся к представлениям
690. Нет, не допускается
691. Что подразумевает математическая модель в оптимизации запросов?
692. На вход запросы и статистика, на выход транзакции
693. На вход план запроса, на выход запросы и статистика
694. На вход запросы и статистика, на выход планы запросов
695. На вход стоимость выполнения запроса, на выход план выполнения запроса
696. Что такое кардинальность в оптимизации запросов?
697. Значение, представляющее долю строк таблицы, удовлетворяющих определенному условию выбора
698. Количество строк, возвращаемых после каждой операции плана выполнения запроса
699. значение, представляющее отношение количества уникальных значений индексируемых столбцов к общему числу строк таблицы
700. наилучшая оценка времени, необходимого для выполнения оператора, полученная оптимизатором
701. Что такое транзакция?
702. Виртуальная таблица, представляющая собой поименованный запрос, который будет подставлен как подзапрос при использовании представления.
703. Группа операций, которая может быть выполнена или не выполнена вместе
704. Объект базы данных, представляющий собой набор SQL-инструкций, который компилируется один раз и хранится на сервере.
705. Хранимая процедура особого типа, которую пользователь не вызывает непосредственно, а исполнение которой обусловлено действием по модификации данных
706. К какой группе операторов относятся COMMIT,ROLLBACK?
707. DML
708. TCL
709. DDL
710. CCL
711. Как расшифровывается ACID?
712. Atomicity, Consistency, Isolation, Durability
713. Atomicity, Callback, Isolation, Durability
714. Atomicity, Consistency, Informative, Durability
715. Atomicity, Callback, Informative, Durability
716. Как называется следующее свойство: Независимость результата выполнения транзакций от параллельно работающих транзакций
717. Согласованность
718. Атомарность
719. Долговечность
720. Изолированность
721. От каких проблем защищает уровень изолированности Repeatable Read?
722. “Грязное” чтение
723. “Грязное” чтение, Неповторяющееся чтение
724. “Грязное” чтение, Фантомное чтение
725. “Грязное” чтение, Неповторяющееся чтение, Фантомное чтение
726. Какой вид блокировки резервирует ресурс только для чтения?
727. Монопольная
728. Олигопольная
729. Разделяемая
730. Обновления
731. На каком языке написан SQLite?
732. Delphi
733. C
734. Python
735. Java
736. Что характеризует режим блокировки EXCLUSIVE?
737. Выполнение блокировки БД после ключевого слова BEGIN, БД будет доступна другим пользователям только для чтения
738. БД блокируется на запись при достижении первого оператора чтения или записи, который находится внутри тела транзакции, БД будет
739. БД будет доступна другим пользователям только для чтения, является режимом по умолчанию
740. Блокировка БД не только на запись, но и на чтение, блокировка происходит после ключевого слова BEGIN
741. Какой оператор фиксирует транзакцию?
742. Rollback
743. Begin
744. Commit
745. Savepoint
746. Что такое ContentProvider?
747. Механизм, обеспечивающий доступ к контактам
748. Механизм, служащий для хранения данных.
749. Механизм, позволяющий обеспечить доступ нескольких приложений к общим данным
750. Механизм. обеспечивающий передачу данных.
751. В каком пакете расположен ContentProvider?
752. android.content
753. android.util
754. java.util
755. android.widget
756. Какие методы должны быть реализованы при наследовании от ContentProvider
757. query, insert, delete, update
758. select, add, remove, update
759. query, insert, delete, update, getType
760. select, add, remove, update, getType
761. Сколько параметров в методе query класса ContentProvider
762. 5
763. 8
764. 3
765. 7
766. Функция метода uriMatcher.Match()
767. Добавляет uri в uriMatcher
768. Делит uri на части
769. Сравнивает URI cо значением, заданным в UriMatcher()
770. Проверяет синтаксическую валидность uri
771. Какие атрибуты есть у тега provider?
772. action, data, category
773. authorities, data, exported
774. action, category, name
775. authorities, name, exported
776. Функция класса ContentResolver
777. Обеспечение доступа к контактам
778. Обеспечение доступа к данным провайдеров
779. Обеспечение доступа приложений к общим данным
780. Обеспечение передачи данных
781. Какие методы есть у класса ContentResolver
782. query, insert, delete, update
783. select, add, remove, update
784. query, insert, delete, update, getType
785. select, add, remove, update, getType
786. Что содержит ContactsContract.Contacts
787. записи о контакте, аккаунт, тип и пр.
788. историю изменения контактов
789. агрегированные данные о контакте
790. данные контакта: адреса электронной почты, номера телефонов и др.
791. Что содержит ContactsContract.RawContacts
792. записи о контакте, аккаунт, тип и пр.
793. историю изменения контактов
794. агрегированные данные о контакте
795. данные контакта: адреса электронной почты, номера телефонов и др.
796. Что содержит ContactsContract.Data
797. записи о контакте, аккаунт, тип и пр.
798. историю изменения контактов
799. агрегированные данные о контакте
800. данные контакта: адреса электронной почты, номера телефонов и др.
801. Какие поля содержит ContactsContract.Contacts?
802. \_ID, NAME\_RAW\_CONTACT\_ID, PHOTO\_ID, HAS\_PHONE\_NUMBER, AGREGATION\_MODE, ACCOUNT\_NAME, DATA\_SET
803. \_ID, LOOKUP\_KEY, NAME\_RAW\_CONTACT\_ID, DISPLAY\_NAME\_PRIMARY, PHOTO\_ID, IN\_VISIBLE\_GROUP,HAS\_PHONE\_NUMBER, CUSTOM\_RINGTONE
804. \_ID, CONTACT\_ID, AGREGATION\_MODE, DELETE, ACCOUNT\_NAME, ACCOUNT\_TYPE, DATA\_SET
805. \_ID, AGREGATION\_MODE, DISPLAY\_NAME\_PRIMARY ACCOUNT\_NAME, IN\_VISIBLE\_GROUP , DATA\_SET, CUSTOM\_RINGTONE
806. Какие поля содержит ContactsContract.RawContacts?
807. \_ ID, NAME\_RAW\_CONTACT\_ID, PHOTO\_ID, HAS\_PHONE\_NUMBER, AGREGATION\_MODE, ACCOUNT\_NAME, DATA\_SET
808. \_ID, LOOKUP\_KEY, NAME\_RAW\_CONTACT\_ID, DISPLAY\_NAME\_PRIMARY, PHOTO\_ID, IN\_VISIBLE\_GROUP,HAS\_PHONE\_NUMBER, CUSTOM\_RINGTONE
809. \_ID, CONTACT\_ID, AGREGATION\_MODE, DELETE, ACCOUNT\_NAME, ACCOUNT\_TYPE, DATA\_SET
810. \_ID, AGREGATION\_MODE, DISPLAY\_NAME\_PRIMARY ACCOUNT\_NAME, IN\_VISIBLE\_GROUP , DATA\_SET, CUSTOM\_RINGTONE
811. На что указывает AGGREGATION\_MODE\_DEFAULT?
812. агрегация контактов не допускается
813. значение по умолчанию, агрегация контактов ведется
814. значение по умолчанию, агрегация контактов не допускается
815. агрегация контактов прекращена, однако если ранее агрегировался, то продолжает
816. На что указывает AGGREGATION\_MODE\_DISABLED?
817. агрегация контактов не допускается
818. значение по умолчанию, агрегация контактов ведется
819. значение по умолчанию, агрегация контактов не допускается
820. агрегация контактов прекращена, однако если ранее агрегировался, то продолжает
821. На что указывает AGGREGATION\_MODE\_SUSPENDED?
822. агрегация контактов не допускается
823. значение по умолчанию, агрегация контактов ведется
824. значение по умолчанию, агрегация контактов не допускается
825. агрегация контактов прекращена, однако если ранее агрегировался, то продолжает
826. Что такое Shared Preferences?
827. Объект базы данных, представляющий собой набор SQL-инструкций, который компилируется один раз и хранится на сервере.
828. постоянное хранилище на платформе Android, используемое приложениями для хранения своих настроек.
829. Механизм, обеспечивающий доступ к контактам.
830. Механизм, обеспечивающий передачу данных.
831. Какой алгоритм шифрования используется в SQLite?
832. 128-bit DES
833. 512-bit Serpent
834. 256-bit AES
835. 128-bit AES
836. Что такое авторизация?
837. право выполнения определенной операции
838. определение, кто такой конкретный пользователь
839. разрешение заимствовать права у другого пользователя для выполнения определенного блока кода
840. определение того, какие действия можно производить аутентифицированному пользователю
841. Что такое привилегия?
842. право выполнения определённой операции
843. определение, кто такой конкретный пользователь
844. разрешение заимствовать права у другого пользователя для выполнения определенного блока кода
845. определение того, какие действия можно производить аутентифицированному пользователю
846. Какие есть привилегии в SQLite?
847. query, add, set, remove
848. select, insert, update, delete, references
849. query, add, set, remove, references
850. select, insert, update, delete
851. Какие права даёт привилегия IMPERSONATE?
852. пользователь получает возможность становиться владельцем объекта
853. пользователь получает возможности владельца, имеет возможность предоставлять разрешения, неявно включает разрешение CONTROL для всех объектов в этой области видимости
854. пользователь может выполнить запрос от имени другой учетной записи
855. получает возможность просматривать метаданные объекта
856. Что такое имперсонификация?
857. право выполнения определённой операции
858. определение, кто такой конкретный пользователь
859. разрешение заимствовать права у другого пользователя для выполнения определенного блока кода
860. определение того, какие действия можно производить аутентифицированному пользователю
861. Что такое номенклатура?
862. представление систем пространственных координат пространственных структур и преобразований между системами пространственных координат
863. набор правил целостности, определяющих поведение пространственно связанных географических объектов
864. грамматика языка разметки XML
865. способ идентификации листов географической карты
866. Что такое SRID?
867. Набор правил целостности, определяющих поведение пространственно связанных географических объектов.
868. грамматика языка разметки XML
869. пространственная система привязки координат
870. представление систем пространственных координат пространственных структур и преобразований между системами пространственных координат
871. Что такое GML
872. Грамматика языка разметки XML
873. пространственная система привязки координат
874. Набор правил целостности, определяющих поведение пространственно связанных географических объектов.
875. представление систем пространственных координат пространственных структур и преобразований между системами пространственных координат
876. Когда не требуется репликация?
877. Синхронизация изменений удаленных баз данных с центральной БД
878. Однократно создать копию БД – копирование БД
879. Перемещение определенных наборов данных с центрального сервера распределение их на другие
880. Преобразование данных и распределение их среди подписчиков
881. Что такое издатели?
882. серверы, предоставляющие данные для репликации на другие серверы
883. совокупность статей, предназначенных для подписчиков.
884. Серверы назначения, которые хранят реплицированные данные и получают обновления
885. Серверы, распространяющие реплицируемые данные
886. Какого вида репликации не существует?
887. Репликация сведением
888. Транзакционная репликация
889. Репликация соединением
890. Репликация моментальных снимков
891. В чём особенность топологии с центральным издателем
892. Организует репликацию между равными участниками
893. БД издателя и дистрибьютора находятся на одном и том же сервере, подписчики — на других серверах.
894. В этом случае в центральную БД подписчика собираются данные от нескольких издателей.
895. Базы данных издателя и дистрибьютора находятся на разных серверах, подписчики на своих серверах
896. В чём особенность топологии с центральным подписчиком?
897. В распространении данных эта топология полагается на других подписчиков
898. Организует репликацию между равными участниками
899. В этом случае в центральную БД подписчика собираются данные от нескольких издателей.
900. Базы данных издателя и дистрибьютора находятся на разных серверах, подписчики на своих серверах
901. В чём особенность Merge Agent
902. Синхронизирует изменения, сделанные на подписчике после копирования с издателя первоначального моментального снимка
903. Перемещает транзакции, отмеченные для репликации, из журнала транзакций на издателе в базу дистрибьютора
904. Сохраняет изменения БД в очередь, при помощи которой обновления могут быть распространены издателю асинхронно
905. Создает моментальные снимки, включающие структуру данных и сами данные
906. Функции Replication Agents Checkup
907. обновляет кэшированные запросы, используемые Replication Monitor, по умолчанию запускается автоматически при запуске SQL Server Agent и работает непрерывно
908. проверка агентов репликации, по умолчанию запускается каждые 10 минут.
909. отмечает все ошибочные подписки, по умолчанию выключено
910. отмечает все ошибочные подписки, по умолчанию включено
911. Что такое экземпляр?
912. сервер базы данных, система и пользовательские базы данных которого не доступны для совместного использования другим экземплярам, выполняющимся на том же компьютере
913. группа операций, которая может быть выполнена или не выполнена вместе
914. таблицы, описывающие структуру объектов базы данных
915. механизм, позволяющий обеспечить доступ нескольких приложений к общим данным
916. Каков размер страницы в SQLServer?
917. 8 Мб
918. 16 Кб
919. 8 Кб
920. 16 Мб
921. Что такое информационная схема?
922. группа операций, которая может быть выполнена или не выполнена вместе
923. таблицы, описывающие структуру объектов базы данных
924. состоит из доступных для чтения представлений, которые предоставляют информацию обо всех таблицах, представлениях и столбцах, к которым имеется доступ
925. механизм, позволяющий обеспечить доступ нескольких приложений к общим данным
926. Какое из утверждений о плане выполнения запроса неверно
927. Если два или более операторов имеют одинаковый отступ, они выполняются в порядке сверху вниз
928. Оператор с самым большим отступом выполняется первым
929. выполняет поиск строк по некластеризованным индексам
930. обрабатываются все листья страницы дерева индексов
931. Что не входит в состав оптимизации запроса?
932. анализ запроса
933. проверка синтаксической валидности
934. выбор индекса
935. выбор метода выполнения операций соединения
936. Что такое SYSTEM в Oracle
937. привилегированный предопределенный пользователь Oracle ранга администратора базы данных, который является владельцем ключевых ресурсов БД Oracle: модули Oracle, таблицы словаря БД, V$-представления словаря
938. привилегированный предопределенный пользователь Oracle, которому принадлежат ключевые ресурсы БД Oracle: представления словаря БД, репозиторий инструментов
939. привилегированный предопределенный пользователь Oracle ранга администратора базы данных, которому принадлежат ключевые ресурсы БД Oracle: представления словаря БД, репозиторий инструментов
940. привилегированный предопределенный пользователь Oracle, который является владельцем ключевых ресурсов БД Oracle: модули Oracle, таблицы словаря БД, V$-представления словаря
941. Для чего используется TEMPORARY?
942. хранение сегментов отката, используется всегда один
943. хранение постоянных объектов БД
944. хранение временных данных
945. хранение постоянных объектов БД, используется всегда один
946. Какой командой назначается привилегия?
947. REVOKE
948. SET
949. DENY
950. GRANT
951. Что такое CLOB в Oracle?
952. Символьный тип большой объект до 4GB
953. Большой двоичный объект до 4GB
954. Бинарные данные до 2GB
955. Указатель на двоичный файл операционной системы
956. Какие уровни изоляции явно поддерживает Oracle?
957. Repeatable Read, Serializable
958. Repeatable Read, Read Uncommited
959. Read Commited, Serializable
960. Read Commited, Read Uncommited
961. Выберите неверное утверждение об идентификаторах Oracle
962. начинается с буквы
963. не содержит пробелов
964. компилятор приводит идентификаторы к верхнему регистру
965. “идентификатор” регистронезависим
966. Какие объекты содержит категория словаря ALL?
967. Все объекты базы данных
968. Объекты, к которым пользователь имеет доступ
969. Объекты сервера
970. Объекты, принадлежащие пользователю
971. Какой размер курсора в SQLite
972. 1 Мб
973. 2 Мб
974. 4 Мб
975. 7 Мб
976. Выберите правильный порядок оптимизации запроса
977. Анализ запроса, выбор индекса, выбор порядка выполнения операций соединения, выбор метода выполнения операций соединения
978. Анализ запроса, выбор индекса, выбор метода выполнения операций соединения, выбор порядка выполнения операций соединения
979. выбор индекса, анализ запроса, выбор порядка выполнения операций соединения, выбор метода выполнения операций соединения
980. выбор индекса, анализ запроса, выбор метода выполнения операций соединения, выбор порядка выполнения операций соединения
981. Каких подсказок оптимизатора не существует?
982. табличные подсказки
983. подсказки соединения
984. подсказки строк
985. структуры планов
986. Выберите неверное утверждение об индексах
987. Предназначен для индексирования столбцов с низкой селективностью
988. Битовый индекс создает битовые карты для каждого возможного значения столбца, где каждому биту соответствует строка, а значение бита 1 (0) означает, что соответствующая строка содержит (не содержит) индексируемое значение
989. Хорошо подходят для хранилищ данных
990. Подходит для таблиц с частым обновлением
991. Выберите неверное утверждение о плане выполнения
992. Оператор с самым большим отступом выполняется последним
993. Compute Scalar вычисляет выражение, выдавая в результате скалярное значение
994. Clustered Index Seek выполняет поиск строк по кластеризованным индексам
995. Index Scan - обрабатываются все листья страницы дерева индексов
996. Для чего предназначен AlterDialog в Android?
997. Это диалоговое окно, чаще всего использующееся в тех случаях, когда нам спросить пользователя о чем-то (разрешение/подтверждение выполнения какой-то операции)
998. Это диалоговое окно, чаще всего использующееся для отображения каких-то списков, а также различных компонентов, к примеру GridView, CardView и т.п
999. Это диалоговое окно, чаще всего использующееся для отображения видео/фото компонентов
1000. Это компонент предназначенный для проигрывания различных медиа файлов, имеющий ряд параметров, позволяющих управлять этим воспроизведением
1001. Выберите путь, по которому находятся файлы SQLite базы данных по умолчанию?
1002. /data/data/Application/name\_database
1003. /data/Application/name\_database
1004. /data/data/Application/databases/name\_database
1005. /data/data/Application/database/name\_database
1006. Для чего используется AUTOINCRIMENT?
1007. Это ключевое слово, используемое для автоматического увеличения значения поля в таблице.
1008. Это ключевое слово, указывающий на то, что при удалении данной таблицы будут удалены все связанные с ней другие таблицы
1009. Это ключевое слово, указывающий на то, что данный столбец является внешним ключом
1010. Это ключевое слово, чаще всего использующийся в тех случаях, когда нам требуется спросить пользователя о чем-то (разрешение/подтверждение выполнения какой-то операции)
1011. Как называется компонент в Android, предназначенный для ввода текста?
1012. TextView
1013. EditText
1014. MultipleText
1015. Text
1016. Что такое DDMS?
1017. Сервер мониторинга отладки Dalvik, который поставляется с Android?
1018. Технология мониторинга использования системной памяти устройства (Memory System)
1019. Технология, позволяющая приложению открывать внутри себя системные приложения, без прямого обращения к ним
1020. Технология, позволяющая осуществлять мониторинг за часто вызываемыми компонентами приложения с дальнейшим предоставлением статистики разработчику
1021. Что предпочтительнее использовать Serializable или Parcelable в Android?
1022. Parcelable нету в Android
1023. Предпочтительнее Serializable
1024. Предпочтительнее Parcelable
1025. Рекомендуется использовать сторонние библиотеки, такие как SPL, которые требуют меньше ресурсов системы для выполнения операций.
1026. Для чего предназначено файловое расширение. apk?
1027. Application Package
1028. Application Program Kit
1029. Android Proprietary Kit
1030. Android Package
1031. Какой метод не является методом жизненного цикла Activity?
1032. onPause()
1033. onSaveInstanceState()
1034. onDestroy()
1035. onRestart()
1036. Какую операцию необходимо выполнить для выключения диалога?
1037. Связать on\_dismiss listener с диалогом
1038. Связать setDismissListener с диалогом
1039. Выполнить dismiss()
1040. Вызвать setOnCancelListener()
1041. Что происходит в первую очередь при отрисовке макета (layout)?
1042. Layout – разметка
1043. Measure – измерение
1044. Validation – утверждение
1045. Computation – расчет
1046. Что объект WebView предоставляет пользователю?
1047. Показывать содержимое веб-страницы
1048. Открывать веб-браузер
1049. Создавать html-файлы
1050. Просматривать веб-страницы в виде исходного кода
1051. Какая утилита командной строки используется для просмотра системной отладочной информации?
1052. Tailer
1053. Logcat
1054. Traceview
1055. Outview
1056. Чтобы включить файлы со статическими данными приложения и получить к ним доступ во время выполнения, необходимо:
1057. Поместить файлы в res/data и открыть их с помощью метода openResource(R.raw("<имя-файла>"))
1058. Cохранить файлы где-нибудь в проекте и открыть их с помощью метода openStream("<имя\_файла\_с\_путем>")
1059. Поместить файлы в res/raw и открыть их с помощью метода openRawResource(R.raw.<имя-файла>)
1060. Использовать инструмент datafile для кодирования файлов в исходные файлы Java
1061. Укажите класс, который используется для создания окна с прогрессом (progress bar)?
1062. ProgressBar
1063. ProgressDialog
1064. ProgressBarDialog
1065. ProgressBarItem
1066. Какую библиотеку нужно использовать для работы с БД через контент-провайдер?
1067. android.db.sqlite
1068. android.db
1069. android.database.sqlite
1070. android.database
1071. Как влияет количество отображаемых процессом UI-элементов, видимых пользователю, на важность процесса с точки зрения Android?
1072. Чем больше видимых UI-элементов - тем больше важность
1073. Чем меньше видимых UI-элементов - тем меньше важность
1074. Процесс отображающий видимые UI-элементы важнее, чем процесс не отображающий видимый UI. Точное количество роли не играет
1075. Никак не влияет
1076. Что нужно добавить к фрагменту кода, чтобы намерение (Intent) могло быть передано следующей деятельности (activity)?
1077. i.addExtra(b);
1078. i.putExtras(b);
1079. i.putBundle(b);
1080. i.addData(b);
1081. fun getInt(): Int = null ?: 0 ; print(getInt()). Какой результат выполнения функции?
1082. MAX\_INT
1083. Exception
1084. 0
1085. Null
1086. Каким способом можно локализовать приложение Android?
1087. приложения Android нельзя локализовать
1088. хранить внутри директории res xml-файлы в папках с полным названием языков english, russian и т.д.
1089. хранить внутри директории res xml-файлы в папках с названием values-english, values-russian и т.д.
1090. хранить внутри директории res xml-файлы в папках с названием values-en, values-ru и т.д.
1091. В ОС Андроид каждый уникальный Linux userID принадлежит?
1092. разным пользователям устройства, так же как и в Linux
1093. разным учетным записям, которые зарегистрированы в телефоне google, facebook, skype и тп
1094. каждому установленному приложению
1095. каждому контакту из телефонной книги телефона
1096. Какой из этих атрибутов используется в файле разметки (.xml) для определения веса элемента среди других элементов?
1097. android:layout\_gravity
1098. android:layout\_weight
1099. android:layout\_width
1100. android:orientation
1101. Какой класс является непосредственным родителем класcов Activity и Service?
1102. ApplicationContext
1103. BaseContext
1104. ContextApp
1105. Context
1106. Явным (explicit) называется Intent, который
1107. Создан при помощи метода
1108. Содержит флаг Intent.FLAG\_EXPLICIT
1109. Intent.createExplicit()
1110. Содержит класс получателя
1111. В каком методе происходит освобождение ресурсов?
1112. onDestroy()
1113. onStop()
1114. onResume()
1115. onStart()
1116. Какие классы не наследуют AdapterView?
1117. ListView
1118. GridView
1119. ViewGroup
1120. Gallery
1121. Свойства архитектуры разделения одновременного доступа Reserved?
1122. Процесс планирует записать в БД. Может быть наложена только одна, при этом может сосуществовать в несколькими SHARED блокировками.
1123. Можно читать данные, но нельзя записать. Может быть несколько одновременных блокировок этого вида.
1124. Процесс ждет освобождения блокировок чтения и готов записать в базу. Новые SHARED блокировки не могут добавляться.
1125. Требуется для записи в базу. Может накладываться только одна, и не может сосуществовать одновременно с блокировками других типов.
1126. Какого метода позиционирования в курсоре не существует?
1127. moveToLast
1128. moveToNext
1129. getPosition
1130. setPosition
1131. Как называется тег, содержащий разрешения?
1132. user-permission
1133. uses-permission
1134. permission
1135. use-permission
1136. Как называется криптографическая библиотека в SQLite
1137. SQLCipher
1138. SQLCriptography
1139. SQLiteCipher
1140. SQLiteCriptography
1141. Когда репликация требуется
1142. Однократно создать копию БД – копирование БД
1143. Скопировать данные с одного сервера на другой с преобразованием – процедура импорта-экспорта
1144. Перемещение определенных наборов данных с центрального сервера распределение их на другие
1145. Скопировать данные с одного клиента на другой с преобразованием – процедура импорта-экспорта
1146. Какого взаимоположения объектов не существует?
1147. Пересечение
1148. Объединение
1149. Слияние
1150. Расстояние
1151. Что такое топология?
1152. представление систем пространственных координат пространственных структур и преобразований между системами пространственных координат
1153. набор правил целостности, определяющих поведение пространственно связанных географических объектов
1154. грамматика языка разметки XML
1155. способ идентификации листов географической карты
1156. Из скольких страниц состоит экстент?
1157. 8
1158. 16
1159. 4
1160. 2
1161. Для чего не применяются системные хранимые процедуры
1162. переименование объектов
1163. идентификация пользователей
1164. мониторинг
1165. создание таблиц
1166. Какой редакции СУБД Oracle не существует?
1167. Standard Edition One
1168. Ultimate Edition
1169. Standard Edition
1170. Enterprise Edition
1171. Укажите неверное утверждение о мультиарендной архитектуре Oracle?
1172. Можно создавать несколько CDB – для разных версий программного обеспечения СУБД
1173. Одну и ту же PDB можно переносить между CDB
1174. В CDB создается главный контейнер Root
1175. CDB содержит метаданные Root
1176. Что произойдет, если элемент <application> в AndroidManifest.xml имеет icon, label атрибуты как у дочерней <activity>?
1177. Никакое значение не будет использоваться, так как Android не сможет определить приоритетное
1178. Значения для <activity> переопределяют значения из <application>
1179. Сгенерируется ошибка компиляции из-за противоречивых значений
1180. Сгенерируется ошибка выполнения из-за противоречивых значений
1181. Явным (explicit) называется Intent, который
1182. Содержит категорию
1183. Создан при помощи метода
1184. Содержит класс получателя
1185. Содержит флаг Intent.FLAG\_EXPLICIT
1186. Категория intent (намерения):
1187. Описывает тип компонентов, который должен обрабатывать намерение
1188. Назначает фильтры полномочий для намерений в группах, а не одиночно
1189. Упорядочивает опции, доступные в контекстном меню селектора действий
1190. Называет структуру пакета, которому принадлежит целевая деятельность
1191. К элементам ввода относят:
1192. Ограничивающие элементы ввода
1193. Ползунки
1194. Счетчики
1195. Все вышеперечисленное
1196. Какая графическая библиотека входит в набор библиотек ОС Android?
1197. OpenCV
1198. DirectX
1199. Open GL
1200. OpenCL