Questions Set

1. Which are the four pillars of oops?Explain each pillar with real life example.
2. Explain the java features.
3. What is class and what is object? Explain it with real life example.
4. Why java is known as truely object oriented language?
5. What is the difference between C++ and Java?
6. Why java is known as platform independant language?
7. Why java is known as robust and dynamic?
8. How java is more secure? Explain
9. Explain memory management in java.
10. What is static memory allocation and dynamic memory allocation? In java, when these types of allocation is done?
11. Descrive the memory areas used in java.
12. What is JVM?
13. What is JIT?
14. What is JRE?
15. What is Class Loader?
16. What is the difference between Class Loader and JIT?
17. What is the Difference between Java Compiler and JIT?
18. In java generally your class name from where execution starts and file name should be same. Why?
19. What is magic behind platform independancy of java?
20. Explain the process of compilation and execution in java.
21. Byte code resides in which file?
22. Which file is generated by java compiler?
23. Whether JVM is platform dependant?
24. Why main in java is declared as public static?
25. What is the use of parameter String args[] of main method?
26. Explain System.out.println().
27. Explain System class properties.
28. Which class is the predefined super class of all the classes?
29. Describe some properties of Object class.
30. What is the use of toString method? Which class' method it is?
31. When you display object of any class, which method will be implicitly called?
32. Why need to override toString method of Object class?
33. What is garbage collection in java?
34. When finalize method is called and why?
35. Why there is no need of explicit destructor in java?
36. In java, at which time static data and literals will get loaded in memory?
37. In java, when objects will get loaded in memory?
38. Explain Java data types with size.
39. Explain primary, derived and user defined data types.
40. Why char data type has size of 2 bytes instead on 1 byte?
41. Explain different types of modifiers in java.
42. What is the difference between access modifiers and non-access modifiers in java?
43. Explain scope of access modifiers.
44. What is the use of static keyword?
45. What is the difference between static and volatile variable?
46. Define the properties of static variables?

47. What are static methods in java?
48. Why there is need to have static variables and methods in java? Elaborate with example.
49. What are static blocks?
50. What do you mean by instance initializer block?
51. What is the difference between static block and instance initializer block?
52. What is the difference between static and non-static variables and methods?
53. When non-static variables get the memory allocated?
54. Why non-static variables defined in class are known as instance variables?
55. What is the use of final keyword?
56. Can we define both nonstatic and static variables as final?
57. What do you mean by local variables?
58. What is the scope of local, non-static and static variables?
59. Can we define local variables as static? Explain yes or no with reason.
60. Can we define local variables as final?
61. What is the default value of all type of instance variables in java?
62. What do you mean by array?
63. How arrays are defined in java?
64. When the memory to array elements is allocated?
65. What do you mean by object array?
66. What will be the default initialization of array?
67. What is class?
68. What is object?
69. What is the difference between class and object?
70. Public classes should be stored in which file?
71. What should be the scope of normal class?
72. In java, how objects are created?
73. What do you mean by constructor?
74. What do you mean by default constructor?
75. When default constructor will get provided?
76. Why we need constructors in java?
77. What constrcutor returns?
78. When constructors are called?
79. Which are types of constructors?
80. How constructors are called implicitly?
81. How constructors are called explicitly?
82. What is object re-initialization? How it is done in java?
83. What is the use if this keyword?
84. Can we define the constructors as private?
85. What should be the accecc modifiers to normal class?
86. Can we have member classes in java?
87. Can we define member classes as private?
88. Why static methods are known as class methods?
89. What are the condiitions on static methods?
90. What is the difference between nonstatic and static methods?
91. What is the use of method?
92. What is difference between method and constructor?
93. If we write the behaviour in constructor then what will happen?
94. Can we overload the constructors? If yes then which type of polymorphism it is?
95. What are the types of polymorphism?
96. Can we override the constructors?

97. Can we overload the methods? If yes then which type of polymorphism it is?
98. What are the conditions of method overloading?
99. Why we need to overload the method? Give example of some inbuilt overloaded methods.
100. If method name and signature is same but return type is different then whether it will be method overloading or not?
101. Which are the types of parameters passing?
102. Can we pass the objects as parameter?
103. How the objects are passed as parameter?
104. What is inheritance?
105. Why we need inheritance in java?
106. What are the types of inheritance?
107. Which type of inheritance is not supported by java? Why?
108. What is the calling sequence of constructors in inheritance?
109. How to call parameterized constructor of immediate super class?
110. What is the use of super keyword?
111. How to explicitly call the immediate super class constructor?
112. If super class and sub class have same instance variable name, in sub class how to differentiate between those?
113. If a subclass want to change the exisitng behaviour of super class what sub class should do?
114. What do you mean by method overriding? Which type of polymorphism it is?
115. Why we need to override the methods?
116. What are the rules of method overriding?
117. What do you mean by static and dynamic binding?
118. Which types of methods are statically bounded?
119. Which types of methods are dynamically bounded?
120. What is upcasting and downcasting?
121. Why downcasting not supported in java?
122. Super class variable can refer subclass object. Is it true? If yes then what type of casting it is? Which are the limitations on such reference variable? Can it access subclass own properties?
123. How to resolve method overriding? Which are the ways to resolve?
124. How to use super keyword to resolve method overriding?
125. How to resolve method overriding at runtime without using super keyword?
126. What is dynamic method dispatch?
127. How to avoid method overriding?
128. How to avoid a class to be inherited? Give example of some inbuilt classes that can not be inherited?
129. What is the use of abstract keyword in java?
130. What is abstract method?
131. Why a class declares abstract method?
132. If a class contains atleast one abstract method, how the class should be declared(means whether it must be abstract or not)?
133. Can we declare the abstract class with no any abstract method?
134. Why absract class is known as partially abstract?
135. What is the difference between abstract method and concrete method?
136. Why their is need to have abstract method in class?
137. Where the abstract method to be implemented?
138. If a class inherits any abstract class then what is the compulsion on subclass?
139. If a subclass does not want to override all the abstract methods of its superclass what is the solution?
140. Can we create the instance of abstract class?

141. What is the syntax of writing abstract method?
142. What are the rules regarding syntax while overriding the abstract method?
143. Can we override private and static method? If no then why?
144. Can we decrease the scope of overriding method in subclass?
145. If we can not create abstract class instance directly then how to access concrete methods of abstract class?
146. Can we declare constructors, instance variables, static variables, final variable in abstract class?
147. How many absract classess a class can extend?
148. If a class can not extend more that one abstract classes, then how to override the methods of other abstract classes in same subclass?
149. Can we declare abstract method as final?
150. Can we declare abstract method as static?
151. Can we declare abstract method as protected in abstract class?
152. What are interfaces?
153. Why interface is known as fully abstract?
154. Whether interface contains only abstract method and not any type of variables?
155. Which types of variables we can declare in interface?
156. What is the default syntax of variables and methods of interface?
157. Can we declare default and protected abstract methods in interface?
158. Interface methods are bydefault public and abstract. True or false?
159. Interface variables are bydefault public static and final?
160. Whether we need to define interface as abstract explicitly like abstract class?
161. Where the interface abstract methods should be overridden?
162. How a class can implement any interface?
163. How many interfaces a class can implement?
164. If a class can implement more that one interfaces then here ambiguity does not occur like abstract classes?
165. Why java does not support multiple inheritance through classes but it can achieve multiple interfaces through interfaces? Elaborate with the example?
166. If a class is implementing two interfaces and if both interfaces contain one same method with same name and type signaures. Can it be possible? If yes then, how many time a class should override that method?
167. What is the syntax of implementing multiple interfaces?
168. Can we create the instance of interface?
169. Can we declare construtor, instance variables in interface?
170. Which type of variables only we can declare in interface?
171. Why interface variables are bydefault public static and final?
172. If interface variables would not made final, What would be happened?
173. If a class implemented any interface, what is the condition on class?
174. If a class does not want to override some abstract methods of interface, what is the solution?
175. Which are the types of interfaces?
176. Can an interface extend another interface?
177. Why need to extend the interfaces?
178. Can interface extend multple interfaces? If yes then why there can not occur ambiguity?
179. Give some examples of inbuilt extended interfaces?
180. Which access modifiers we can give to normal interfaces?
181. Can we define interfaces as static?
182. What are nested interfaces?
183. Why to nest the interfaces?

184. Which type of access modifiers we can give to nested interfaces?
185. Where the private nested interfaces should be implemented?
186. How to implement the nested interface in classes?
187. Give one example in java where nested interfaces are used.
188. Can we store interface in one package and implement it in another package? If yes then what should be the scope of interface?
189. Can we access interface with default scope within the same package?
190. What do you mean by functional interfaces? Give some examples of inbuilt functional interfaces.
191. What are marker interfaces?
192. Does marker interface contain any method signature or variables?
193. What is the use of marker interfaces in java?
194. Give some examples of inbuilt marker interfaces.
195. Can we define a class inside another class or interfaces?
196. String objects are immutable. What do you mean by this?
197. Why string objects are immutable?
198. How to create strings in Java?
199. Which are the ways to create the strings?
200. What is the difference if strings are created using new keyword and if strings are created using literals?
201. What happens if we are creating string objects with same contents in heap and in literal pool?
202. When the literals will get loaded in memory?
203. List some String class properties?
204. What is the difference between equals and compareTo method?
205. How to extract character from particular position of string?
206. How to determine index of particular character or char sequence in string?
207. How to make the string to uppar case or lower case?
208. What is the use of trim method?
209. How to determine the substring of any string?
210. When you do any modifications in string, whether actual string will get modified or not?
211. How create mutable strings?
212. What is the difference between StringBuffer and StringBuilder?
213. What is the difference between String and StringBuffer?
214. How many objects will get created in following statement?
   String str=new String(“HelloWorld”);
215. What is the use of intern method?
216. Can we call intern method on literals?
217. What do you mean by content comparison and reference comparison? How to do both comparisons?
218. Extract the domain name from given email address.
   user123@gmail.com
219. Reverse the string contents.
   hellodad
220. Check whether given string is pallindrome or not?
   DAD, MOM, HELLO
221. Display vowels from given string.
   India is my country
222. Display repeated characters from given string.
   He saw the parrot then He jump out from pool.

223. Replace the string by number from given string.
    There was one farmer. He had two childrens.
224. What do you mean by package?
225. Why packages are needed?
226. List some inbuilt packages of java?
227. What are the advantages of using packages?
228. How to create packages in java?
229. Can we have package inside another package?
230. How to create hierarchical packages?
231. Can we store interfaces in packages?
232. Can we have same name classes in different classes?
233. What do you mean by avoiding name space collision?
234. Where the package is searchable by default?
235. If package is not in current working directory, how to make the package searchable?
236. How to set the class path up to packages?
237. How to import the classes from packages?
238. If the members of particular class are default, can it be accessed outside of its packages?
239. How to control the visibility using packages?
240. What do you mean by static import?
241. What are the advantages of static import over normal import?
242. What do you mean by - import packagename.*;
243. What do you mean by nested class?
244. Why to do the nesting of classes?
245. What should be the scope of normal class?
246. Which access modifiers we can give to nested class?
247. If nested class is private then what is the scope of that class?
248. Which are the types of nested classes?
249. What is the difference between static nested class and inner class(non-static nested class).
250. Can nested class access the private members of outer class?
251. Can outer class access the private members if nested class?
252. Why static nested classes are not frequently used?
253. How to create the object of static nested class?
254. What do you mean by inner class?
255. What are the types of inner classes?
256. How to create the object of inner class?
257. What do you mean by member inner class?
258. What are local inner classes?
259. Why local inner classes are used?
260. What do you mean by annonymous inner classes?
261. Why there is need to use annonymous inner classes?
262. How to create object of annonymous inner class?
263. Where the annonymous inner classes are generally used(For which purpose)?
264. In event handling where the annonymous inner classes are used?
265. Whether annonymous inner class is the type of member inner class or local inner class?
266. What is the compiler's responsibility if we are using annonymous inner class?
267. What is error and exception?
268. What happens if exception occurs in code in java?
269. Give one situation where exception will occur?
270. What is default exception handler?
271. Can user handles the exception which may occur in code?

272. Which class hierarchy is used to handle the exceptions in java?
273. Describe Exception class hierarchy?
274. What is the difference between Exception and Error class?
275. What do you mean by Checked Exceptions and Unchecked Exceptions?
276. Why checked exceptions need to be checked at compile time?
277. Give example of some Checked exception classes.
278. In java, readLine method of BufferedReader class throws which checked exception?
279. In java, join and sleep method of Thread class throws which checked exception?
280. Which are the keywords and clauses used in exception handling?
281. Explain use of try, catch, finally, throw and throws in detail?
282. What is the syntax of catch block?
283. What will happen if exception will occur in try block?
284. What is the use of finally block?
285. Why need to write resource closing statements into finally? What will happend if we write those into try block?
286. What is the execution sequence of try catch finally?
287. What is the execution sequence of try finally?
288. Can we have try with finally only? If possible, who will handle the exception?
289. Why to use throw clause?
290. What is the syntax of using throw clause?
291. How to explicitly throw any exception?
292. What is exception propagation?
293. What is the use of throws clause?
294. How exception is propagated using throws clause?
295. What is the syntax of using throws?
296. How many excpetions can be declared in throws clause?
297. What is the difference between throw and throws?
298. What is implicit exception throwing and explicit exception throwing?
299. How to handle user defined exceptions?
300. Why to propagate exceptions towards caller? If we use check the conditions using if else then why this way is not better that exception propagation?
301. What is the difference between process and thread?
302. List different states of thread.
303. What is the difference between multiprocessing and multithreading?
304. Why multithreading is important?
305. How multithreading is achieved in Java?
306. Explain some Thread class properties.
307. What is the use of currentThread method?
308. What is the description of Thread object?
309. Which method is the entry point of main thread?
310. What is the default name, priority and group name of main thread?
311. How to control the current executing thread?
312. How to create threads in java?
313. Which are the different ways to create and execute the threads?
314. How to check whether the thread is alive or not?
315. How to change name and priority of the thread?
316. List three priority constants defined in Thread class.
317. What is the use of ThreadGroup class?
318. Which Thread constructors are used to create threads by extending Thread class and by implementing Runnable interface?

319. What is the use of join method?
320. How to call join method?
321. How to pause the execution of thread for some period?
322. What is the difference between join and sleep?
323. What is the use of isAlive method?
324. How to start a thread in java?
325. What is the entry point of child thread?
326. What is the syntax of run method?
327. How run method will get called?
328. How execution of child thread starts?
329. If we call the run method explicitly then what will happen?
330. How to create multiple threads?
331. How to create thread by implementing Runnable interface?
332. Which type of interface Runnable is?
333. Why Runnable is known as functional interface?
334. Which method signature is there in Runnable?
335. Which way is better to create the threads?
336. Can we create threads in one class and execute those in other class?
337. What is Thread synchronization? Why it is needed? What will happend if it is not there
338. How to achieve Thread synchronization in java?
339. How to synchronize the methods and blocks in java?
340. Which are the ways in java to synchronize the threads? Which way is better?
341. How to synchronize the method without attaching synchronized keyword?
342. What is static synchronization?
343. What is interthread communication in java?
344. How to communizate among threads in java?
345. Explain wait, notify and notifyAll. Of which class, these methods are?
346. What is the difference between notify and and notifyAll?
347. What is the difference between wait and sleep?
348. What is the difference between wait and join?
349. What is the diffence if thread is waiting by calling wait() method and if thread is waiting by calling wait(long miliseconds)?
350. Can we override wait method?
351. Can we override sleep methods?
352. What is the use of volatile variable?
353. What is the difference between static variable and voltile variable?
354. When deadlock occurs?
355. Can wait() method be responsible for deadlock?
356. What is the work of thread schedular?
357. What is the minimum, maximum and normal priority of threads in java?
358. Explain type of streams?
359. What do you mean by file handling?
360. Why file handling is needed?
361. Explain hierarchy of byte stream classes and character stream classes
362. Which are the methods of InputStream and Reader class?
363. Which are the methods of OutputStream and Writer class?
364. How to open the file in read mode?
365. How to open the file in write mode?
366. How to open the file in append mode?
367. FileInputStream, FileOutputSteam, File Reader and FileWriter throws which type of

exception? Whether it is checked or unchecked?

368. read, write and close methods throw which type of execption? Whether it is checked or unchecked?
369. What is the difference between InputStream and Reader?
370. What is the use of Scanner class?
371. What is the use of PrintWriter class?
372. What is Object Serialization? Why it is needed?
373. How to do serialization in java?
374. Which marker interface need to be implemented by a class which object is to be serialized?
375. Which serialization and deserialization methods are used? Wherer these methods are defined?
376. Whether Serializable interface contains the serialization methods?
377. Which classes are needed in Serialization process?
378. What is the use of transient keyword?
379. Why collection framework is needed?
380. What is the difference between using Object Array and using Collection framework?
381. What are the advantages of using collection over the object array?
382. Explain Collection Hierarchy and Map hierarchy?
383. Why to make the collection generic?
384. List the interfaces and classes used to collect and work on objects?
385. What is the difference between List and Set?
386. What is the difference between List and Queue?
387. What is the difference between Set and Queue?
388. Which interface is extending Set?
389. Which interface is extending Queue?
390. Which are the classes implementing List interface?
391. What is the difference between ArrayList and LinkedList?
392. What is the difference between ArrayList and Vector?
393. Which are the classes implementing Set interface and SortedSet interface?
394. What is the difference between HashSet, LinkedHashSet and TreeSet?
395. Which are the classes implementing Queue interface and Deque interface?
396. What is the property of Deque?
397. Why Map is independant from Collection hierarchy?
398. What is the property of Map?
399. Which are the classses implementing Map interface and SortedMap interface?
400. What is the difference between HashMap, LinkedHashMap and TreeMap?
401. What is the difference between HashMap and HashTable?
402. How to traverse the collection?
403. Which interfaces are used to traverse the collection?
404. What is the difference between ListIterator and Iterator?
405. What is the difference between Iterator and Enumerator?
406. What is the difference between for and enhanced for?
407. How to use enhanced for to traverse the collection?
408. How to traverse collection which is implementing Map?
409. Why Entry interface is used in Map?
410. What type of interface Entry is?
411. What is the difference betwen entrySet and keySet?
412. How to restrict the Map to not to allow duplicate values?
413. In which applications, Map can be used?
414. What is the use of Collections class?

415. What is the difference between Collection and Collections?
416. How to sort the objects in ArrayList?
417. If the LinkedListy contain String objects, how to sort the strings?
418. If TreeSet contain Integer objects, can those be sorted implicitly?
419. How to sort the objects that can not be sorted naturally by just using sort method? For example if a collection contains Book type objects how to sort those?
420. Why Comparable and Comparator interfaces are needed?
421. What is the difference between Comparable and Comparator?
422. If we want to sort the Employees on the basis of employee salary as well as employee experience, Which interface is useful among above?
423. Explain the process of sorting employee objects, if these are stored in LinkedList?
424. Explain the process of sorting employee objects, if these are stored in TreeSet?
425. How to search the particular element in List?
426. How to shuffle the elements in List?
427. sort method implicitly implements which type of sorting algorithm?
428. What is JNI?
429. What do you mean by native methods?
430. What do you mean by JDBC?
431. Which are the types of drivers?
432. What is Class in java?
433. How to establish connections with database using java?
434. What is the difference between Statement, PreparedStatment and CallableStatement?
435. What is the difference between execute, executeQuery and executeUpdate?
436. What is the return type of executeQuery?
437. How to process the results retrieved from query?
438. What is the return type of executeUpdate? What it returns?