Q.No	Part A (5 x 2 = 10 marks) (Answer all the questions)
1	What are the different ways to create string object?
2	What is the default access modifier in Java for class members?
3	Define the term "method signature" in Java.
4	What is the difference between the throw and throws keywords in Java?
5	You have a multi-line string with inconsistent indentation. How would you normalize it?

Q.No	Part B - (2 x 16 = 32 marks), (1 x 8 = 8 marks) (Answer all the questions)
11 A	Explore the use of inner classes in Java. Discuss the advantages of using inner classes, especially in the context of encapsulation and code organization. Provide an example that showcases the application of inner classes in a real-world scenario.
	OR
11 B	In a basic library system, you have a base class LibraryItem representing common attributes like title and itemID. Create two subclasses, Book and DVD, that inherit from LibraryItem and add attributes specific to each, such as author for books and duration for DVDs. Implement a sample scenario where you instantiate objects of both subclasses and demonstrate the useof inherited and subclass-

	specific attributes. Discuss how the concept of inheritance promotes code reuse and organization in this context.
12 A	Create a scenario-based Java program for a simple address book application. The program will allow users to store contacts in a file, view existing contacts, add new contacts, and search for contacts by name.Contacts are stored in a file named contacts.txt, where each line contains the name and phone number separated by a comma.
OR	
12 B	i) Assume that you're working on a file. Describe how you would reads the data from the file and display the data on console window using byte streams. ii) You are implementing a feature in your java application that involves writing data to a file.How would you handle IOExceptionthat may occur during the file-writing process?
13 A	Delve into the concept of protected members in inheritance. Provide a detailed explanation of how protected members facilitate encapsulation and controlled access within the class hierarchy. Offer an example to demonstrate the practical implementation of protected members.