

Roll No.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Total No. of Pages: 02

Total No. of Questions: 09

M.Sc. (IT) (2015 Onwards) (Sem. – 4)
ADVANCED JAVA PROGRAMMING
M Code: 74113
Subject Code: MSIT-401
Paper ID: [74113]

Time: 3 Hrs.

Max. Marks: 60

INSTRUCTIONS TO CANDIDATES:

1. **SECTIONS-A, B, C & D** contains **TWO** questions each carrying **TEN** marks and students has to attempt any **ONE** question from each **SECTION**.
2. **SECTION-E** is **COMPULSORY** consisting of **TEN** questions carrying **TWENTY** marks in all.

SECTION A

1. a) Describe different states in lifecycle of Thread? Write steps to create a Thread in Java with help of example.
b) What are the benefits of multi-threaded programming? (6, 4)
2. a) What is Lock interface in Java Concurrency API? What are its benefits over synchronization?
b) What is Synchronization? For what purposes is the keyword synchronized used? (5, 5)

SECTION B

3. What is Serialization in Java? What is the need of Serialization? Write steps to make a Java class Serializable? (10)
4. What is a IO stream? What are the types of Streams and classes of the Streams? Explain the necessity of two types of streams – byte streams and character streams. (10)

SECTION C

5. What is JDBC Connection? Explain steps to get Database connection in a simple java program with help of example. (10)
6. a) What is JDBC PreparedStatement? What are the benefits of PreparedStatement over Statement?
b) What is JDBC Transaction Management and why do we need it? (5, 5)

SECTION D

7. Explain the working of Generics in Java with help of suitable example? What is type erasure? (10)
8. What is Annotation? Describe how Annotations Work and Explain how to create user defined Annotations with suitable example. (10)

SECTION E

9. Write short notes on the following with help of example/diagram if needed:
 - a) Semaphores
 - b) Serialization
 - c) Daemon Threads
 - d) Callable
 - e) Importance of Generics
 - f) JDBC
 - g) Locks
 - h) Inheritance
 - i) Atomic Variables
 - j) Input Stream Hierarchy