SHRI VILE PARLE KELAVANI MANDAL'S

SHRI BHAGUBHAI MAFATLAL POLYTECHNIC

AND COLLEGE OF ENGINEERING

PT 2 – JAVA QUESTION BANK

CHAPTER NO:04
Exception Handling {10 MARKS}

Q.1	**Describe the working of the following with snippets (4m)** **a) try b) throw c)throws d) catch e)Finally**
Q.2	**Describe the class hierarchy of Exception. Also elaborate on caught and uncaught exceptions (6m)**
Q.3	**Explain the process of creating a user-defined exception with an example program.(6m)**
Q.4	**1. Write a Java program to find out sqrt of a number, generate an exception for a negative number(4m)** **2. WAP in Java to read Employee experience (years)(6m)** **(lower limit = 0 upper limit = 40) and generate an exception if it is 41 or above**
Q.5	**Explain checked and Unchecked exceptions in Java. (6m)**

CHAPTER NO: 06
AWT and SWING Controls { 12 MARKS}

Q.6	**Draw the hierarchy of AWT classes. Describe any 2 in brief. (4m)**
Q.7	**Explain the event delegation model OR event handling mechanism with a diagram (4m)**
Q.8	**Enlist all inbuilt packages (4m)**
Q.9	**Difference Between AWT and Swing. Give the different Swing controls. Also, write a simple program to demonstrate Swing controls. (6m)**
Q.10	**Describe the Adapter class in Java (4m)**
Q.11	**Explain the following Event listener with an example program (4m each)** **a)ActionListener b)KeyListener c)MouseListener** **d)MouseMotionListener e)WindowListener**
Q.12	**Enlist a different layout manager in AWT (6m)**

Q.13	**Develop a Java program where a user types text into a TextField and the program displays the last key pressed in a label. (4m)**
Q.14	**Write a Java AWT application that responds to window minimize and restore actions by updating a status label. (4m)**
Q.15	**Create a program that asks for confirmation (Yes/No dialog) when the user tries to close the application window.(4m)**
Q.16	**Compare AWT and SWING min4 points(4m)**
Q.17	**WAP in Java to implement a simple arithmetic calculator (8m)**
Q.18	**WAP to design a Login form in Java and display if the login is successful or not. (8m)**
Q.19	**Design a Java AWT application that detects mouse clicks and movements inside a frame and displays the coordinates on the screen. (6m)**
Q.20	**Create a Java program where the background color of the frame changes when the mouse enters and resets when the mouse exits the frame. (4m)**

CHAPTER NO:05
Multithreading {10 MARKS}

Q.21	**Describe the Thread Life cycle. (4m)**
Q.23	**Explain the life cycle of a thread in Java with a neat diagram. What are the different thread states? (6m)**
Q.24	**Give the different ways to create a thread in Java? Explain with examples. (6m)**
Q.25	**Explain implementing threads by extending the Thread class with an example.**
Q.26	**Explain implementing threads by the Runnable interface with an example.**
Q.27	**How does thread priority affect thread execution? Does Java guarantee priority-based scheduling (6m)**
Q.28	**Explain the concept of synchronization in Java. How does the synchronized keyword work for methods and blocks?(6m)**
Q.29	**How do wait(), notify(), and notifyAll() work in Java? Explain with an example of the producer-consumer problem (4m)**
Q.30	**Write a Java program where the main thread waits for two child threads to complete execution using the join() method. (6m)**
Q.31	**Explain the following methods of the thread class with an example: i)start() ii)run( ) iii)sleep() iv)yield() v)suspend() vi)resume() vii)join viii)isAlive()**
Q.32	**Compare extending the thread and implementing Runnable, min 4 points (4m)**
Q.33	**Explain inter-thread communication with an example.(4m)**
Q.34	**Design a Java program using multithreading where one thread (Producer) generates a series of numbers and another thread (Consumer) consumes them. The producer should not produce if the buffer is full, and the consumer should not consume if the buffer is empty. Use wait() and notify() methods for inter-thread communication. (8m)**

Q.35 (6m/8m)

Design and implement a Java application using **AWT (Abstract Window Toolkit)** that creates a **Student Registration Form** with the following components:

- **Components to include:**

1. **Labels** – For field names: Name, Gender, Hobbies, Country, Skills, Address
2. **TextFields** – For entering Name and Email
3. **Checkboxes** – For selecting Hobbies (Reading, Sports, Music)
4. **CheckboxGroup (Radio Buttons)** – For selecting Gender (Male, Female, Other)
5. **Choice (Dropdown)** – For selecting Country (e.g., India, USA, UK)
6. **List** – For selecting multiple Skills (e.g., Java, Python, C++)
7. **TextArea** – For entering full Address
8. **Buttons** – Submit and Reset

- **Functional Requirements:**

- On clicking the **Submit** button, display all entered/selected information using a **dialog box or console output**.
- On clicking **Reset**, clear all fields.
- Use proper **layout managers** (e.g., GridLayout, FlowLayout) for neat arrangement.