# **Workspace**

**📌 Tosca Workspace – Interview Questions (Final List)**

**🔹 Basic (Concepts & Fundamentals)**

1. What is a workspace in Tosca?
2. Types of workspaces in Tosca (Single-user vs Multi-user)?
   * **Single-user workspace**

* Local workspace stored in a file-based database.
* Used when only one tester is working on the project.
* No central repository → no collaboration.
* Suitable for small projects, POCs, or offline work.
  + **Multi-user workspace**
* Connected to a **common repository (DB like Oracle/SQL Server/DB2)**.
* Multiple testers can work in parallel, with check-in/check-out version control.
* Supports collaboration, traceability, and versioning.
* Used for large enterprise projects with many team members.

1. How do you create a new workspace?
   * The main components of a Tosca workspace are:
2. **Modules** → Store technical information of application objects (scanned via XScan).
3. **TestCases** → Business workflows created by combining modules.
4. **TestStep Blocks / Reusable Assets** → For modularization and reusability.
5. **ExecutionLists** → To execute TestCases and track results.
6. **Test Configuration Parameters (TCPs)** → For environment-specific values (URLs, credentials, etc.).
7. **Requirements & Issues** → To map business requirements and defects.
8. **Buffers & Libraries** → For data storage and reusable functions.
9. **Logs & Reports** → For execution tracking and reporting.
10. What are the main components of a workspace?
    * **Local Workspace**

* A personal copy of the project database stored on the tester’s machine.
* Used for creating, modifying, and executing TestCases.
* Can be repaired, reset, or recreated if corrupted.
* Changes here are only visible to the user until they are checked in.
  + **Common Repository**
* Central database (SQL/Oracle/DB2) shared by all team members.
* Stores all project artifacts (Modules, TestCases, ExecutionLists, etc.) with version control.
* Ensures collaboration, synchronization, and traceability.
* Requires check-in/check-out to manage updates and prevent conflicts.

1. Difference between Local workspace and Common repository?
2. How do you open/close a workspace in Tosca?
3. What is the purpose of Tosca Commander in relation to the workspace?
   * Tosca Commander is the **graphical user interface (GUI)** of Tosca, and it’s the main tool to interact with a workspace. Its purpose is to:

* **Create, manage, and organize** all workspace artifacts (Modules, TestCases, ExecutionLists, Requirements).
* Provide **check-in/check-out control** for multi-user workspaces.
* Enable **test design, execution, and reporting** within the workspace.
* Handle **configuration** (TCPs, environments, test data).
* Act as the central hub where testers, developers, and managers collaborate through the same workspace.

1. Where is the workspace file stored (.tws file)?

* The Tosca workspace file (.tws) is stored **locally on the tester’s machine**. By default, it’s created in the user’s **Documents\Tosca Workspaces** folder (or a custom path defined during workspace setup).
* For a **single-user workspace**, the. tws directly holds the project data.
* For a **multi-user workspace**, the .tws only contains connection details to the **common repository (DB)** and local cache.
* So the. tws is basically the **entry point file** Tosca Commander uses to open the workspace.

1. What is the role of DB in a workspace (SQLite/Oracle/MS SQL)?
   * The **database (DB)** is the backbone of a Tosca workspace.

* In a **single-user workspace**, Tosca uses a **local SQLite DB** to store all Modules, TestCases, ExecutionLists, and logs.
* In a **multi-user workspace**, Tosca connects to a **central DB** (Oracle, MS SQL, or DB2) which acts as the **common repository**. This DB stores all project artifacts, manages **check-in/check-out**, and keeps **version history**.
  + **Role of DB:**
* Stores and organizes all workspace objects.
* Provides **data integrity, version control, and traceability**.
* Enables **collaboration** in large teams by keeping a single source of truth.
* Supports **performance optimization** via indexing and DB maintenance.

1. How do you take a backup of a workspace?
   * I take workspace backups in two ways:

* **Single-user workspace** → Copy the .tws file (SQLite DB) or export important assets (Modules, TestCases, Libraries) as. tsu**/.xml files**.
* **Multi-user workspace** → The main backup is taken at the **database level (SQL/Oracle/DB2)** by the DBA team. For my local workspace, I create a copy of the .tws file. Before upgrades or major changes, I also **export critical TestCases/Modules** as a safety backup.