

## IT Management & Audits

Practical Lab Manual

# IT Governance Framework Mapper

Practical P02

### Learning Domain

IT Governance & Compliance Frameworks

### Course Learning Outcomes

CLO02: Apply IT governance frameworks (COBIT, ITIL) to organizational processes

### Unit

Unit II: IT Governance & Compliance Frameworks

**Time Allocation:** 3 hours

**Learning Mode:** Hands-on (70%) + Theory (30%)

**Difficulty Level:** Intermediate

## **IT Governance Framework Mapper**

Practical P02

## Quick Reference

---

<b>Practical Code</b>	P02
<b>Practical Name</b>	IT Governance Framework Mapper
<b>Slot</b>	T/P-2
<b>Duration</b>	3 hours
<b>CLO Mapping</b>	CLO02
<b>Unit</b>	Unit II: IT Governance & Compliance Frameworks
<b>Delivery Mode</b>	Hands-on Lab
<b>Target Audience</b>	Intermediate Level
<b>India Integration</b>	HIGH
<b>Screenshot Count</b>	5 Required

## Prerequisites

---

- Basic understanding of IT governance concepts
- Familiarity with command-line interface (CLI)
- Python 3.8+ installed on your system
- Completion of P01 (Lab Setup & Infrastructure) recommended
- Understanding of organizational IT processes

## Tools Required

---

<b>Tool</b>	<b>Version</b>	<b>Free</b>	<b>Notes</b>
Python	3.8+	✓	Required
pip (Python package manager)	Latest	✓	Included with Python
Terminal/CLI	-	✓	bash or PowerShell
Text Editor / IDE	-	✓	VS Code recommended
Web Browser	Latest	✓	For viewing HTML reports

### Learning Objectives

- ✓ Understand COBIT 2019 and ITIL v4 governance framework structures
- ✓ Set up a Python-based governance mapping tool from a Git repository
- ✓ Map organizational IT processes to COBIT 2019 governance objectives
- ✓ Map organizational IT processes to ITIL v4 management practices
- ✓ Perform gap analysis to identify uncovered framework areas
- ✓ Generate compliance scorecards and governance reports
- ✓ Apply governance mapping to India-specific FinTech compliance (RBI, SEBI)

## What You Will Learn

---

By the end of this practical, you will:

1. Understand the structure and purpose of COBIT 2019 (5 domains, 40 objectives)
2. Understand ITIL v4 practices (3 categories, 34 practices)
3. Clone and set up an open-source governance framework mapper tool
4. Define organizational IT processes in structured JSON format
5. Run process-to-framework mapping using a CLI tool
6. Interpret mapping confidence scores and coverage percentages
7. Perform gap analysis to find governance blind spots
8. Generate professional governance reports (terminal and HTML)

## Real-World Application

---

IT governance frameworks are essential for financial institutions in India. The Reserve Bank of India (RBI) mandates that banks and NBFCs implement robust IT governance aligned with international standards like COBIT and ITIL. Companies such as **HDFC Bank**, **Paytm**, and **Bajaj Finance** use these frameworks to ensure their IT processes meet regulatory requirements, manage risk, and deliver reliable digital services.

## Hands-On Procedure

### Part A: Environment Setup

#### Step 1: Clone the Governance Framework Mapper Repository

**Objective:** Download and set up the governance framework mapping tool on your local machine.

**Instructions:**

1. Open your terminal (bash on Linux/Mac, PowerShell on Windows)
2. Navigate to your working directory
3. Clone the repository from GitHub
4. Navigate into the project directory
5. Verify the project structure

**Code/Command:**

```
1 # Clone the repository
2 git clone <repository-url> governance-framework-mapper
3 cd governance-framework-mapper
4
5 # View the project structure
6 ls -la
7 # Expected directories: src/, data/, docs/, templates/, tests/
8
9 # View source code structure
10 ls src/
11 # Expected: cli.py, mapper.py, analyzer.py, reporter.py
12 # Expected: frameworks/ directory with cobit.json, itil.json
```

Clone and Explore the Repository

#### Expected Output

Project directory contains:

**src/** – Source code (CLI, mapper, analyzer, reporter)  
**src/frameworks/** – COBIT and ITIL framework data (JSON)  
**data/** – Sample organizational process data  
**templates/** – HTML report templates  
**docs/** – Reference documentation  
**tests/** – Unit tests

If you do not have Git installed, you can download the repository as a ZIP file from the GitHub page and extract it manually.

**Step 2: Set Up Python Virtual Environment and Install Dependencies**

**Objective:** Create an isolated Python environment and install required packages.

**Instructions:**

1. Create a Python virtual environment
2. Activate the virtual environment
3. Install project dependencies from `requirements.txt`
4. Verify installation by running the CLI help command

**Code/Command:**

```
1 # Create virtual environment
2 python -m venv venv
3
4 # Activate (Linux/Mac)
5 source venv/bin/activate
6
7 # Activate (Windows PowerShell)
8 .\venv\Scripts\Activate.ps1
9
10 # Install dependencies
11 pip install -r requirements.txt
12
13 # Verify installation
14 python src/cli.py --help
```

Python Environment Setup

**Expected Output**

CLI help output shows available commands:  
Usage: cli.py [OPTIONS] COMMAND [ARGS]...

Commands:

map	Map organizational processes to frameworks
analyze	Perform gap analysis on framework coverage
report	Generate governance compliance reports

If pip install fails: (1) Ensure Python 3.8+ is installed: `python -version`, (2) Try pip3 instead of pip, (3) On Windows, ensure Python is in your PATH, (4) Try: `python -m pip install -r requirements.txt`.

**Screenshot 1**

**What to paste:** Terminal showing the cloned repository file structure and the CLI help output after running `python src/cli.py -help`.

*Paste your screenshot here*

## Part B: Understanding the Governance Frameworks

### Step 3: Explore COBIT 2019 Framework Data

**Objective:** Understand the COBIT 2019 framework structure as represented in the tool.

**Instructions:**

1. Open the COBIT framework data file
2. Identify the 5 COBIT domains (EDM, APO, BAI, DSS, MEA)
3. Note the total number of governance objectives (40)
4. Review the keywords associated with each objective
5. Understand how the tool will match processes to objectives using these keywords

**Code/Command:**

```
1 # View the COBIT framework file
2 cat src/frameworks/cobit.json | python -m json.tool | head -60
3
4 # Count total objectives
5 python -c "
6 import json
7 with open('src/frameworks/cobit.json') as f:
8     data=json.load(f)
9     for domain in data['domains']:
10        print(f"{domain['id']}: {domain['name']} - {len(domain['objectives'])} objectives")
11 print(f"Total objectives: {sum(len(d['objectives']) for d in data['domains'])}")
12 "
```

Explore COBIT 2019 Data

#### Expected Output

COBIT 2019 Domains:

EDM: Evaluate, Direct and Monitor - 5 objectives  
APO: Align, Plan and Organize - 14 objectives  
BAI: Build, Acquire and Implement - 10 objectives  
DSS: Deliver, Service and Support - 6 objectives  
MEA: Monitor, Evaluate and Assess - 5 objectives  
Total objectives: 40

### Step 4: Explore ITIL v4 Framework Data

**Objective:** Understand the ITIL v4 framework structure as represented in the tool.

**Instructions:**

1. Open the ITIL framework data file



2. Identify the 3 practice categories (General, Service, Technical)
3. Note the total number of practices (34)
4. Review keywords for several practices
5. Compare ITIL practice structure with COBIT objectives

**Code/Command:**

```
1 # View the ITIL framework file
2 cat src/frameworks/itil.json | python -m json.tool | head -60
3
4 # Count total practices
5 python -c "
6 import json
7 with open('src/frameworks/itil.json') as f:
8     data=json.load(f)
9     for cat in data['categories']:
10         print(f"\t{cat['name']} - {len(cat['practices'])} practices")
11     print(f"\tTotal practices: {sum(len(c['practices']) for c in data['
12         categories'])}")
13 "
```

Explore ITIL v4 Data

**Expected Output**

ITIL v4 Practice Categories:

General Management Practices - 14 practices

Service Management Practices - 17 practices

Technical Management Practices - 3 practices

Total practices: 34

COBIT focuses on *what* IT governance should achieve (objectives), while ITIL focuses on *how* to deliver IT services (practices). Organizations often use both frameworks together for comprehensive governance.

**Screenshot 2**

**What to paste:** Terminal output showing the COBIT 2019 domain summary (5 domains with objective counts) and the ITIL v4 category summary (3 categories with practice counts).

*Paste your screenshot here*

## Part C: Mapping Processes to Frameworks

### Step 5: Review the Sample Organization Processes

**Objective:** Understand the input format and review sample IT processes for a digital payments company.

**Instructions:**

1. Open the sample organizational process file
2. Review the JSON structure: each process has an ID, name, description, and tags
3. Count the number of processes defined
4. Understand how tags and descriptions are used for keyword matching

**Code/Command:**

```
1 # View the sample processes
2 cat data/sample_org_processes.json | python -m json.tool
3
4 # Count processes
5 python -c "
6 import json
7 with open('data/sample_org_processes.json') as f:
8     data=json.load(f)
9     print(f"Total processes: {len(data['processes'])}\n")
10    for p in data['processes'][:5]:
11        print(f"    {p['id']}: {p['name']}\n")
12    print('...')
13 "
```

Review Sample Process Data

#### Expected Output

Sample processes for a digital payments company:

Total processes: 15+

PROC-001: IT Strategy and Planning

PROC-002: Information Security Management

PROC-003: Change Management Process

PROC-004: Incident Response Procedure

PROC-005: Vendor Risk Assessment

...

The quality of mapping results depends heavily on how well processes are described. Use detailed descriptions and relevant tags for accurate matching.

### Step 6: Run COBIT 2019 Mapping

**Objective:** Map the sample organization's IT processes to COBIT 2019 governance objectives.

**Instructions:**

1. Run the mapping command targeting COBIT framework
2. Review the mapping output showing matched objectives
3. Note the confidence scores for each mapping
4. Identify which processes have strong vs. weak matches

**Code/Command:**

```
1 # Run COBIT mapping
2 python src/cli.py map \
3     --input data/sample_org_processes.json \
4     --framework cobit
5
6 # The output will show each process mapped to
7 # its best-matching COBIT objectives with
8 # confidence scores (0.0 to 1.0)
```

Map Processes to COBIT 2019

#### Expected Output

Mapping results table showing:

Process: IT Strategy and Planning

-> AP002 (Managed Strategy) - Score: 0.85

-> EDM01 (Ensured Governance Framework) - Score: 0.72

Process: Information Security Management

-> AP013 (Managed Security) - Score: 0.92

-> DSS05 (Managed Security Services) - Score: 0.78

Score interpretation: 0.70–1.00 = Strong, 0.40–0.69 = Moderate, below 0.40 = Weak

### Step 7: Run ITIL v4 Mapping

**Objective:** Map the same processes to ITIL v4 management practices.

**Instructions:**

1. Run the mapping command targeting ITIL framework
2. Compare ITIL mapping results with COBIT results
3. Note how the same process may map to different practices in each framework

**Code/Command:**

```
1 # Run ITIL mapping
2 python src/cli.py map \
3     --input data/sample_org_processes.json \
4     --framework itil
5
6 # Run mapping against BOTH frameworks simultaneously
7 python src/cli.py map \
8     --input data/sample_org_processes.json \
9     --framework all
```

#### Map Processes to ITIL v4

##### Expected Output

ITIL v4 mapping results showing:

Process: Incident Response Procedure

-> Incident Management - Score: 0.91

-> Problem Management - Score: 0.65

Process: Change Management Process

-> Change Enablement - Score: 0.88

-> Release Management - Score: 0.55

Running with `-framework all` maps processes to both COBIT and ITIL simultaneously, allowing you to see how governance objectives (COBIT) and service practices (ITIL) complement each other.

**Screenshot 3**

**What to paste:** Terminal output showing the COBIT 2019 mapping results table with process names, matched objectives, and confidence scores.

*Paste your screenshot here*

## Part D: Gap Analysis & Report Generation

### Step 8: Perform Gap Analysis

**Objective:** Identify which COBIT and ITIL areas are not covered by the organization's current processes.

**Instructions:**

1. Run the gap analysis command for COBIT
2. Review the coverage percentage per domain
3. Identify uncovered objectives (governance blind spots)
4. Run gap analysis for ITIL
5. Compare coverage across both frameworks

**Code/Command:**

```
1 # COBIT gap analysis
2 python src/cli.py analyze \
3     --input data/sample_org_processes.json \
4     --framework cobit
5
6 # ITIL gap analysis
7 python src/cli.py analyze \
8     --input data/sample_org_processes.json \
9     --framework itil
```

Run Gap Analysis

### Expected Output

COBIT 2019 Gap Analysis:

EDM (Evaluate, Direct, Monitor): 60% covered (3/5 objectives)  
APO (Align, Plan, Organize): 71% covered (10/14 objectives)  
BAI (Build, Acquire, Implement): 50% covered (5/10 objectives)  
DSS (Deliver, Service, Support): 83% covered (5/6 objectives)  
MEA (Monitor, Evaluate, Assess): 40% covered (2/5 objectives)

Overall COBIT Coverage: 62.5% (25/40 objectives)

Uncovered objectives:

EDM03: Ensured Risk Optimization  
EDM05: Ensured Stakeholder Engagement  
...

Gap analysis is critical for compliance. Indian financial regulators (RBI, SEBI) expect institutions to demonstrate coverage of governance frameworks during audits.

## Step 9: Generate Governance Reports

**Objective:** Create professional compliance reports in terminal and HTML formats.

**Instructions:**

1. Generate a terminal-formatted compliance report
2. Generate an HTML report for sharing with stakeholders
3. Open the HTML report in your web browser
4. Review the compliance scorecard, domain-level breakdown, and recommendations

**Code/Command:**

```
1 # Terminal report
2 python src/cli.py report \
3     --input data/sample_org_processes.json \
4     --format terminal
5
6 # HTML report
7 python src/cli.py report \
8     --input data/sample_org_processes.json \
9     --format html \
10    --output governance_report.html
11
12 # Open HTML report in browser (Linux/Mac)
13 open governance_report.html
14
15 # Open HTML report in browser (Windows)
16 start governance_report.html
```

Generate Reports

### Expected Output

Terminal report displays:

```
=====
GOVERNANCE COMPLIANCE SCORECARD
=====
COBIT 2019 Coverage:  62.5%
ITIL v4 Coverage:    58.8%
Overall Score:       60.7%
=====
```

HTML report saved to: `governance_report.html`



**Screenshot 4**

**What to paste:** Terminal output showing the gap analysis results with domain-level coverage percentages and the compliance scorecard summary.

*Paste your screenshot here*

## Part E: Create Your Own Process Mapping

### Step 10: Define Custom IT Processes and Map Them

**Objective:** Create your own set of IT processes and map them to governance frameworks.

**Instructions:**

1. Create a new JSON file with 5–8 IT processes for a hypothetical Indian FinTech company
2. Include processes relevant to RBI/SEBI compliance requirements
3. Run mapping and gap analysis on your custom processes
4. Generate a report for your custom mapping

**Code/Command:**

```
1 # Create a custom process file
2 cat > data/my_processes.json << 'EOF'
3 {
4   "organization": "My_FinTech_Startup",
5   "processes": [
6     {
7       "id": "FT-001",
8       "name": "UPI_Payment_Processing",
9       "description": "End-to-end_UPI_transaction_handling,_
10        settlement,_and_reconciliation_with_NPCI",
11       "tags": ["payment", "UPI", "transaction", "settlement", "NPCI"]
12     },
13     {
14       "id": "FT-002",
15       "name": "KYC_and_Customer_Onboarding",
16       "description": "Digital_KYC_verification_using_Aadhaar,_PAN,_
17        and_video_verification_as_per_RBI_norms",
18       "tags": ["KYC", "identity", "verification", "onboarding", "
19        compliance"]
20     },
21     {
22       "id": "FT-003",
23       "name": "Data_Privacy_and_Protection",
24       "description": "Personal_data_protection,_encryption,_consent
25        _management_aligned_with_DPDP_Act",
26       "tags": ["privacy", "data_protection", "encryption", "consent
27        ", "DPDP"]
28     },
29     {
30       "id": "FT-004",
31       "name": "Fraud_Detection_and_Prevention",
32       "description": "Real-time_transaction_monitoring,_anomaly_
33        detection,_and_fraud_alerting_system",
34       "tags": ["fraud", "monitoring", "detection", "security", "
35        risk"]
36     },
37     {
38       "id": "FT-005",
39       "name": "Compliance_Automation",
40       "description": "Automated_reporting_and_monitoring_for_RBI_SEBI_requirements",
41       "tags": ["compliance", "automation", "reporting", "monitoring"]
42     },
43     {
44       "id": "FT-006",
45       "name": "Risk_Assessment_Framework",
46       "description": "Regular_risk_assessments_and_mitigation_strategies",
47       "tags": ["risk", "assessment", "mitigation", "framework"]
48     },
49     {
50       "id": "FT-007",
51       "name": "Incident_Response_Plan",
52       "description": "Procedures_for_handling_security_incidents_and_data_breaches",
53       "tags": ["incident_response", "security", "data_breach"]
54     },
55     {
56       "id": "FT-008",
57       "name": "Business_Continuity_Plan",
58       "description": "Strategies_to_maintain_operations_during_disasters_or_outages",
59       "tags": ["business_continuity", "disaster_recovery", "outage"]
60     }
61   ]
62 }
```

```
31     "id": "FT-005",
32     "name": "IT_Change_Management",
33     "description": "Controlled_change_process_for_production_
34     systems_including_review,_approval,_and_rollback",
35     "tags": ["change", "release", "deployment", "approval", "
36     rollback"]
37   }
38 ]
39 EOF
40 # Map custom processes to both frameworks
41 python src/cli.py map \
42     --input data/my_processes.json \
43     --framework all
44
45 # Run gap analysis
46 python src/cli.py analyze \
47     --input data/my_processes.json \
48     --framework cobit
49
50 # Generate HTML report
51 python src/cli.py report \
52     --input data/my_processes.json \
53     --format html \
54     --output my_fintech_report.html
```

### Create Custom Process File

#### Expected Output

Custom mapping results showing your 5 FinTech processes mapped to COBIT and ITIL controls, followed by gap analysis showing uncovered areas, and an HTML report generated.

Try adding more processes to increase your coverage score. Focus on areas flagged as gaps — these represent governance blind spots that regulators may question during audits.

**Screenshot 5**

**What to paste:** HTML governance report opened in a web browser, showing the compliance scorecard with domain-level breakdown and coverage percentages.

*Paste your screenshot here*

## Conceptual Background

---

### IT Governance Fundamentals

IT governance is the framework that ensures IT investments support business objectives, risks are managed, and resources are used responsibly. It provides the structure for aligning IT strategy with business strategy.

#### Key Principles:

- **Strategic Alignment:** IT objectives must support business goals
- **Value Delivery:** IT must deliver promised benefits within budget
- **Risk Management:** IT risks must be identified, assessed, and mitigated
- **Resource Management:** IT resources must be used efficiently
- **Performance Measurement:** IT performance must be tracked against targets

### COBIT 2019 Framework

COBIT (Control Objectives for Information and Related Technologies) is developed by ISACA. Version 2019 provides a comprehensive governance framework for enterprise IT.

#### COBIT 2019 Domain Structure

Domain	Full Name	Objectives	Focus Area
EDM	Evaluate, Direct, Monitor	5	Board-level governance, strategic direction
APO	Align, Plan, Organize	14	IT strategy, architecture, resources
BAI	Build, Acquire, Implement	10	Solution development, change management
DSS	Deliver, Service, Support	6	IT operations, security, service delivery
MEA	Monitor, Evaluate, Assess	5	Performance monitoring, compliance

#### Key COBIT 2019 Concepts

- **Governance vs. Management:** Governance (EDM) sets direction; Management (APO, BAI, DSS, MEA) executes

- **Design Factors:** Organization size, threat landscape, compliance requirements, IT role, sourcing model
- **Capability Levels:** 0 (Incomplete) to 5 (Optimizing) — measures process maturity
- **Goals Cascade:** Enterprise Goals → Alignment Goals → Governance/Management Objectives

## ITIL v4 Framework

ITIL (Information Technology Infrastructure Library) is the most widely adopted IT service management (ITSM) framework. Version 4 focuses on co-creation of value through service management.

### ITIL v4 Practice Categories

Category	Practices	Focus Area
General Management	14	Cross-cutting: strategy, security, risk, finance
Service Management	17	Service lifecycle: design, transition, operations
Technical Management	3	Infrastructure, deployment, development

### ITIL v4 Service Value System (SVS)

The SVS describes how all components work together to create value:

1. **Guiding Principles:** Focus on value, start where you are, progress iteratively, collaborate, think holistically, keep it simple, optimize and automate
2. **Governance:** Evaluate, direct, and monitor organizational activities
3. **Service Value Chain:** Plan → Improve → Engage → Design & Transition → Obtain/Build → Deliver & Support
4. **Practices:** The 34 management practices applied across the value chain
5. **Continual Improvement:** Ongoing review and enhancement of services

## COBIT vs. ITIL: Comparison

Aspect	COBIT 2019	ITIL v4
Focus	IT governance (what to achieve)	IT service management (how to deliver)
Developed by	ISACA	Axelos (PeopleCert)
Primary audience	Board, IT governance, auditors	IT operations, service managers
Structure	5 domains, 40 objectives	3 categories, 34 practices
Maturity model	Capability levels (0–5)	Practice maturity (based on SVS)
Best used for	Audit, compliance, governance	Service delivery, operations

## India-Specific Governance Context

### RBI IT Governance Requirements

The RBI mandates IT governance for banks and NBFCs through:

- 1. IT Governance Framework:** Board-level IT strategy committee required
- 2. IS Audit:** Annual information systems audit mandatory
- 3. IT Risk Management:** Documented risk assessment and mitigation
- 4. Business Continuity:** DR plan with regular testing
- 5. Outsourcing Guidelines:** Third-party risk management for cloud/vendor services

### SEBI Cybersecurity Framework

SEBI requires stock brokers and market intermediaries to:

- Designate a Chief Information Security Officer (CISO)
- Implement a cybersecurity policy aligned with standards
- Conduct regular vulnerability assessments and penetration testing
- Report cyber incidents within 6 hours
- Maintain audit trails for a minimum of 5 years

## Digital Personal Data Protection (DPDP) Act

India's DPDP Act (2023) adds governance requirements:

- Consent-based data processing
- Data localization for significant data fiduciaries
- Right to erasure and data portability
- Mandatory breach notification within 72 hours
- Penalties up to INR 250 crore for non-compliance

## Real-World Example: HDFC Bank IT Governance

**Company:** HDFC Bank (India's largest private sector bank)

### Governance Approach:

- ▷ Board-level IT Strategy Committee aligned with COBIT EDM domain
- ▷ ITIL-based service management for 24/7 digital banking operations
- ▷ Annual IS audits covering all COBIT domains as required by RBI
- ▷ Dedicated cybersecurity operations center (SOC) aligned with DSS domain
- ▷ Vendor governance program covering 200+ technology partners
- ▷ BCP/DR testing twice annually with RBI-mandated recovery targets



## Assessment & Deliverables

---

### Deliverables Checklist

Item	Description	Type	Status
Screenshot 1	Repository structure & CLI help	Paste	<input type="checkbox"/>
Screenshot 2	Framework data exploration	Paste	<input type="checkbox"/>
Screenshot 3	COBIT mapping results	Paste	<input type="checkbox"/>
Screenshot 4	Gap analysis & score-card	Paste	<input type="checkbox"/>
Screenshot 5	HTML report in browser	Paste	<input type="checkbox"/>
Custom JSON	Your custom process file	Paste	<input type="checkbox"/>
Coverage %	COBIT & ITIL scores	Text	<input type="checkbox"/>
Gap List	Uncovered objectives	Text	<input type="checkbox"/>

### Verification Checklist

Complete all items below before submitting:

- ☐ Repository cloned and virtual environment set up
- ☐ CLI tool runs successfully with `-help`
- ☐ COBIT 2019 framework data explored (5 domains, 40 objectives)
- ☐ ITIL v4 framework data explored (3 categories, 34 practices)
- ☐ Sample processes mapped to COBIT with confidence scores
- ☐ Sample processes mapped to ITIL with confidence scores
- ☐ Gap analysis completed showing coverage percentages
- ☐ Terminal compliance report generated
- ☐ HTML compliance report generated and opened in browser
- ☐ Custom process file created with 5+ FinTech-relevant processes
- ☐ Custom processes mapped and analyzed
- ☐ All 5 required screenshots captured

## Grading Rubric

Criteria	Description	Points	Score
Setup	Repo cloned, environment ready	10	___/10
Framework Knowledge	COBIT & ITIL structures explored	15	___/15
COBIT Mapping	Sample processes mapped correctly	15	___/15
ITIL Mapping	Sample processes mapped correctly	15	___/15
Gap Analysis	Coverage % and gaps identified	15	___/15
Report Generation	Terminal and HTML reports created	10	___/10
Custom Mapping	Own processes defined and mapped	15	___/15
Documentation	Answers & explanations complete	5	___/5
	<b>TOTAL</b>	<b>100</b>	___/100

## Assessment Questions

Answer the following questions in your submission:

- Q1.** What are the 5 domains of COBIT 2019? Briefly describe each domain's purpose.
- Q2.** How does COBIT differ from ITIL? When would you use each framework?
- Q3.** What is the difference between IT governance and IT management? Give an example.
- Q4.** Why is gap analysis important for compliance? How would you prioritize closing gaps?
- Q5.** What governance frameworks does the RBI require Indian banks to implement?
- Q6.** Your gap analysis showed MEA (Monitor, Evaluate, Assess) at only 40% coverage. What processes would you add to improve this?
- Q7.** How would you use this mapping tool during an actual IT audit engagement?
- Q8.** Describe how COBIT's goals cascade works (Enterprise Goals → Alignment Goals → Objectives).

## Appendix A: COBIT 2019 Quick Reference

### Complete Objective List

ID	Domain	Objective
EDM01	EDM	Ensured Governance Framework Setting and Maintenance
EDM02	EDM	Ensured Benefits Delivery
EDM03	EDM	Ensured Risk Optimization
EDM04	EDM	Ensured Resource Optimization
EDM05	EDM	Ensured Stakeholder Engagement
APO01	APO	Managed IT Management Framework
APO02	APO	Managed Strategy
APO03	APO	Managed Enterprise Architecture
APO04	APO	Managed Innovation
APO05	APO	Managed Portfolio
APO06	APO	Managed Budget and Costs
APO07	APO	Managed Human Resources
APO08	APO	Managed Relationships
APO09	APO	Managed Service Agreements
APO10	APO	Managed Vendors
APO11	APO	Managed Quality
APO12	APO	Managed Risk
APO13	APO	Managed Security
APO14	APO	Managed Data
BAI01	BAI	Managed Programs
BAI02	BAI	Managed Requirements Definition
BAI03	BAI	Managed Solutions Identification and Build
BAI04	BAI	Managed Availability and Capacity
BAI05	BAI	Managed Organizational Change
BAI06	BAI	Managed IT Changes
BAI07	BAI	Managed IT Change Acceptance and Transitioning
BAI08	BAI	Managed Knowledge
BAI09	BAI	Managed Assets
BAI10	BAI	Managed Configuration
DSS01	DSS	Managed Operations
DSS02	DSS	Managed Service Requests and Incidents
DSS03	DSS	Managed Problems
DSS04	DSS	Managed Continuity
DSS05	DSS	Managed Security Services
DSS06	DSS	Managed Business Process Controls
MEA01	MEA	Managed Performance and Conformance Monitoring
MEA02	MEA	Managed System of Internal Control
MEA03	MEA	Managed Compliance with External Requirements

ID	Domain	Objective
MEA04	MEA	Managed Assurance

## Appendix B: ITIL v4 Practice Quick Reference

---

### General Management Practices

1. Architecture Management
2. Continual Improvement
3. Information Security Management
4. Knowledge Management
5. Measurement and Reporting
6. Organizational Change Management
7. Portfolio Management
8. Project Management
9. Relationship Management
10. Risk Management
11. Service Financial Management
12. Strategy Management
13. Supplier Management
14. Workforce and Talent Management

### Service Management Practices

1. Availability Management
2. Business Analysis
3. Capacity and Performance Management
4. Change Enablement
5. Incident Management
6. IT Asset Management
7. Monitoring and Event Management
8. Problem Management

9. Release Management
10. Service Catalog Management
11. Service Configuration Management
12. Service Continuity Management
13. Service Design
14. Service Desk
15. Service Level Management
16. Service Request Management
17. Service Validation and Testing

## Technical Management Practices

1. Deployment Management
2. Infrastructure and Platform Management
3. Software Development and Management

## Appendix C: JSON Process File Format

### Input File Structure

```
1 {  
2   "organization": "Organization_Name",  
3   "processes": [  
4     {  
5       "id": "PROC-001",  
6       "name": "Process_Name",  
7       "description": "Detailed description of the process  
8       including activities and scope",  
9       "tags": ["keyword1", "keyword2", "keyword3"]  
10    }  
11  ]  
12 }
```

Process JSON Format Reference

### Best Practices for Process Definitions

- **Name:** Use clear, descriptive names (e.g., “Change Management Process”)

- **Description:** Include 2–3 sentences covering what the process does, its scope, and key activities
- **Tags:** Use 3–6 relevant keywords that match framework terminology
- **Coverage:** Define at least 10–15 processes for meaningful gap analysis

### Match Score Interpretation

Score Range	Rating	Interpretation
0.70 – 1.00	Strong	Process clearly covers the framework control
0.40 – 0.69	Moderate	Process partially covers the control
0.15 – 0.39	Weak	Some overlap but insufficient coverage
Below 0.15	No Match	Process does not cover this control

## Appendix D: Troubleshooting Guide

### Common Issues and Solutions

**Problem:** `SyntaxError` or module import failures

**Solutions:**

1. Verify Python version: `python -version` (must be 3.8+)
2. Try `python3` instead of `python`
3. Ensure virtual environment is activated
4. Reinstall dependencies: `pip install -r requirements.txt`

**Problem:** Mapping returns empty or zero-score results

**Solutions:**

1. Check JSON file format is valid: `python -m json.tool < file.json`
2. Ensure process descriptions contain relevant keywords
3. Add more descriptive tags to processes
4. Lower the matching threshold if needed

**Problem:** Browser shows blank page or file not found

**Solutions:**

1. Check report file was created: `ls -la governance_report.html`
2. Open manually via browser File → Open
3. Check terminal for error messages during report generation
4. Try generating in a different output directory

## Appendix E: Additional Resources

### Official Documentation

- ISACA COBIT Resources: <https://www.isaca.org/resources/cobit>
- ITIL Official Site: <https://www.axelos.com/best-practice-solutions/itil>
- RBI IT Framework: <https://www.rbi.org.in>
- SEBI Cybersecurity Circular: <https://www.sebi.gov.in>
- DPDP Act: <https://www.meity.gov.in>

### Learning Resources

- “COBIT 2019 Framework: Introduction and Methodology” – ISACA
- “ITIL 4 Foundation” – Axelos Official Guide
- “IT Governance: Policies and Procedures” – Wallace and Webber
- “Information Security Management” – ISO/IEC 27001 standard

### Tools Used in This Practical

Tool	Purpose	Cost
Python 3.8+	Programming language run-time	Free
Click (Python library)	CLI interface framework	Free
Rich (Python library)	Terminal formatting and tables	Free
Jinja2 (Python library)	HTML report template engine	Free
Git	Version control and repository clone	Free

**—END OF LAB MANUAL—**

Document Version: 1.0

IT Management & Audits – Practical Lab Series