काठमाडौं महानगरपालिका

Project Manager, System Analyst, Front-End Developer, Back-End Developer र Database Engineer को लागि प्रतियोगात्मक परीक्षाको पाठ्यक्रम

पाठ्यक्रमको रुपरेखाः यस पाठ्यक्रमको आधारमा निम्नानुसार ३ चरणमा परीक्षा लिइने छ :

प्रथम चरणः लिखित परिक्षण (Written Examination) पूर्णाङ्कः- ५०

द्वितीय चरणः (क) प्रयोगात्मक परिक्षण (Practical Test) पूर्णाङ्कः- ३०

(ख) अन्तर्वार्ता (Interview) पूर्णाङ्कः- २०

परिक्षा योजना (Examination Scheme)

प्रथम चरणः लिखित परिक्षण (Written Examination)

पत्र/विषय	पूर्णाङ्क	उत्तीर्णाङ्क	परिक्षा प्रणाली	प्रश्नसंख्या X अङ्क	समय
लिखित (Written Examination – As per Syllabus)	५०	२०	बहुबैकल्पिक प्रश्न (MCQs)	५० प्रश्न X १ अङ्क	१ घण्टा

द्वितीय चरणः प्रयोगात्मक परिक्षण (Practical Test) र अन्तर्वार्ता (Interview)

पत्र/विषय	पूर्णाङ्क	उत्तीर्णाङ्क	परिक्षा प्रणाली	समय
प्रयोगात्मक परिक्षण (Practical Test)	३०	१०	प्रयोगात्मक परिक्षण (Practical Test)	१ घण्टा
अन्तर्वार्ता (Interview)	२०		अन्तर्वार्ता (Interview)	

द्रष्टव्यः

१. परिक्षाः काठमाडौं महानगरपालिकाले तोके बमोजिमको स्थानमा सूचना प्रविधिको प्रयोग गरि संचालन गरिनेछ।

२. परिक्षामा कुनै प्रकारको उपकरण लगायत अन्य third party application हरु प्रयोग गर्न पाइने छैन ।

३. यस पाठ्यक्रम योजना अन्तर्गतका पत्र/विषयका विषयवस्तुमा जेसुकै लेखिएको भए तापनि पाठ्यक्रममा परेका कानून, ऐन, नियम तथा नीतिहरु परिक्षाको मिति भन्दा ३ महिना अगाडि (संशोधन भएका वा संशोधन भई हटाईएका वा थप गरी संशोधन भई) कायम रहेकालाई यस पाठ्यक्रममा परेको सम्झनु पर्दछ ।

४. प्रथम चरणको परिक्षाबाट छनौट भएका उम्मेदवारहरुलाई मात्र द्वितीय चरणको परिक्षामा सम्मिलित गराइनेछ ।

Syllabus for Project Manager

1. Software Development Fundamentals (10)

- Software Development Life Cycle (SDLC): Waterfall, Agile, Scrum, Kanban
- Software Engineering Principles: Modularity, cohesion, coupling, design patterns
- Version Control Systems: Git (branching, merging, pull requests, conflict resolution)
- Software Testing: Unit testing, integration testing, system testing, test-driven development (TDD)
- Documentation: Writing technical specifications, user manuals, and API documentation

2. Programming and Problem Solving (10)

- Basic Data Structures: Arrays, linked lists, stacks, queues, trees, graphs
- Algorithms: Sorting (e.g., quicksort, mergesort), searching (binary search), recursion
- Problem-Solving: Writing pseudocode, solving algorithmic problems (e.g., time complexity, space complexity)
- Coding Standards: Clean code principles, naming conventions, code refactoring

3. General IT Knowledge (10)

- Operating Systems: Basics of Windows, Linux, and macOS (file systems, processes, permissions)
- Networking: TCP/IP, HTTP/HTTPS, REST APIs, DNS, basic network security
- Databases: Relational vs. non-relational databases, basic SQL queries (SELECT, JOIN, GROUP BY)
- Cloud Computing: Basics of cloud services (AWS, Azure, GCP), IaaS, PaaS, SaaS

4. Soft Skills and Project Management Basics (10)

- Team Collaboration: Effective communication, conflict resolution, working in cross-functional teams
- Time Management: Prioritization, meeting deadlines, managing workloads
- Problem Analysis: Root cause analysis, critical thinking, decision-making
- Basic Project Management: Task estimation, risk management, stakeholder communication

5. Project Management Fundamentals (10)

- Introduction to Project Management
- Project Life Cycle & SDLC Models (Waterfall, Agile, Spiral, V-Model)
- Triple Constraints (Scope, Time, Cost)
- Project Management Standard Practices

6. Project Planning & Scheduling (10)

- Work Breakdown Structure (WBS)
- Gantt Charts, PERT & CPM
- Estimation Techniques (Function Point, COCOMO, Use Case Estimation)
- Requirements Gathering and Stakeholder analysis

7. Project Execution & Monitoring (15)

- Agile Frameworks (Scrum roles/events, Kanban boards)
- Change Management
- Budget and Resource Management
- Key Performance Indicators (KPIs), Earned Value Analysis (EVA)

8. Risk, Quality & Procurement Management (15)

- Risk Identification & Mitigation Strategies
- Quality Assurance vs Quality Control
- Vendor and Contract Management
- Issue Tracking, logging, prioritizing and resolving issues

9. Tools & Governance (10)

- Project Management Tools (MS Project, JIRA, Trello)
- IT Governance: COBIT, ITIL basics
- Standards related to Software Development
- Managing stakeholder conflicts & scope creep

Syllabus for System Analyst

1. Software Development Fundamentals (10)

- Software Development Life Cycle (SDLC): Waterfall, Agile, Scrum, Kanban
- Software Engineering Principles: Modularity, cohesion, coupling, design patterns
- Version Control Systems: Git (branching, merging, pull requests, conflict resolution)
- Software Testing: Unit testing, integration testing, system testing, test-driven development (TDD)
- Documentation: Writing technical specifications, user manuals, and API documentation

2. Programming and Problem Solving (10)

- Basic Data Structures: Arrays, linked lists, stacks, queues, trees, graphs
- Algorithms: Sorting (e.g., quicksort, mergesort), searching (binary search), recursion
- Problem-Solving: Writing pseudocode, solving algorithmic problems (e.g., time complexity, space complexity)
- Coding Standards: Clean code principles, naming conventions, code refactoring

3. General IT Knowledge (10)

- Operating Systems: Basics of Windows, Linux, and macOS (file systems, processes, permissions)
- Networking: TCP/IP, HTTP/HTTPS, REST APIs, DNS, basic network security
- Databases: Relational vs. non-relational databases, basic SQL queries (SELECT, JOIN, GROUP BY)
- Cloud Computing: Basics of cloud services (AWS, Azure, GCP), IaaS, PaaS, SaaS

4. Soft Skills and Project Management Basics (10)

- Team Collaboration: Effective communication, conflict resolution, working in cross-functional teams
- Time Management: Prioritization, meeting deadlines, managing workloads
- Problem Analysis: Root cause analysis, critical thinking, decision-making
- Basic Project Management: Task estimation, risk management, stakeholder communication

5. Foundations of Systems Analysis (10)

- Role of System Analyst
- Software Development Life Cycle (SDLC) Models
- Requirement Engineering Overview
- System Modeling and feasibility Analysis

6. Requirement Gathering & Modeling (15)

- Techniques: Interviews, Questionnaires, Observation, Workshops
- Use Case Modeling
- Data Flow Diagrams (DFD) and Entity Relationship Diagrams (ERD)
- UML Diagrams (Class, Sequence, Activity, State)
- Requirement Analysis

7. System Design (15)

- Logical vs Physical Design
- Input/Output Design Principles
- Interface Design Principles
- Prototyping
- Tools and Techniques for System Design

8. Feasibility & Documentation (10)

- Technical, Economic, Operational Feasibility
- Cost-Benefit Analysis
- System Requirement Specification (SRS) Writing
- Use of Case tools

9. Testing & Implementation (10)

- Testing Approaches: Unit, Integration, UAT
- System Conversion Methods (Parallel, Phased, Pilot, Direct)
- Post-Implementation Review
- Change Request Handling
- Maintenance Models

Syllabus for Front End Developer

1. Software Development Fundamentals (10)

- Software Development Life Cycle (SDLC): Waterfall, Agile, Scrum, Kanban
- Software Engineering Principles: Modularity, cohesion, coupling, design patterns
- Version Control Systems: Git (branching, merging, pull requests, conflict resolution)
- Software Testing: Unit testing, integration testing, system testing, test-driven development (TDD)
- Documentation: Writing technical specifications, user manuals, and API documentation

2. Programming and Problem Solving (10)

- Basic Data Structures: Arrays, linked lists, stacks, queues, trees, graphs
- Algorithms: Sorting (e.g., quicksort, mergesort), searching (binary search), recursion
- Problem-Solving: Writing pseudocode, solving algorithmic problems (e.g., time complexity, space complexity)
- Coding Standards: Clean code principles, naming conventions, code refactoring

3. General IT Knowledge (10)

- Operating Systems: Basics of Windows, Linux, and macOS (file systems, processes, permissions)
- Networking: TCP/IP, HTTP/HTTPS, REST APIs, DNS, basic network security
- Databases: Relational vs. non-relational databases, basic SQL queries (SELECT, JOIN, GROUP BY)
- Cloud Computing: Basics of cloud services (AWS, Azure, GCP), IaaS, PaaS, SaaS

4. Soft Skills and Project Management Basics (10)

- Team Collaboration: Effective communication, conflict resolution, working in cross-functional teams
- Time Management: Prioritization, meeting deadlines, managing workloads
- Problem Analysis: Root cause analysis, critical thinking, decision-making
- Basic Project Management: Task estimation, risk management, stakeholder communication

5. Web Foundations (10)

- HTML5 (semantic tags, forms, multimedia)
- CSS3 (selectors, box model, flexbox, grid)
- JavaScript ES6+ Basics (variables, functions, objects, DOM manipulation)
- Events in Javascript

6. Advanced JavaScript & Frontend Programming (15)

- Event Handling & Async Programming (Promises, async/await)
- Event Delegation and Bubbling
- ES6 Features (arrow functions, classes, modules)
- Error Handling & Debugging
- Data Handling with JSON

7. Frameworks & Libraries (15)

- Popular UI Frameworks and Libraries
- Hooks and Lifecycle Methods
- Introduction to React (components, props, state)
- State Management (Redux / Context API basics)
- Routing in Single Page Applications (React Router / Angular Router)
- UI libraries (Bootstrap, Tailwind, Material UI)

8. UI/UX & Web Standards (10)

- Responsive Web Design (media queries, mobile-first design)
- Accessibility Standards (WCAG, ARIA roles)
- CSS Preprocessors (SASS/LESS basics)
- Design Systems
- Design principles for usability
- Performance Testing for UI

9. Integration & Optimization (10)

- Fetch API, Axios & API integration (REST/GraphQL)
- Caching and Local Storage
- Webpack & Build Tools Basics
- Performance Optimization (lazy loading, caching, minimizing requests)
- Browser Developer Tools for Debugging and Optimization

Syllabus for Back End Developer

1. Software Development Fundamentals (10)

- Software Development Life Cycle (SDLC): Waterfall, Agile, Scrum, Kanban
- Software Engineering Principles: Modularity, cohesion, coupling, design patterns
- Version Control Systems: Git (branching, merging, pull requests, conflict resolution)
- Software Testing: Unit testing, integration testing, system testing, test-driven development (TDD)
- Documentation: Writing technical specifications, user manuals, and API documentation

2. Programming and Problem Solving (10)

- Basic Data Structures: Arrays, linked lists, stacks, queues, trees, graphs
- Algorithms: Sorting (e.g., quicksort, mergesort), searching (binary search), recursion
- Problem-Solving: Writing pseudocode, solving algorithmic problems (e.g., time complexity, space complexity)
- Coding Standards: Clean code principles, naming conventions, code refactoring

3. General IT Knowledge (10)

- Operating Systems: Basics of Windows, Linux, and macOS (file systems, processes, permissions)
- Networking: TCP/IP, HTTP/HTTPS, REST APIs, DNS, basic network security
- Databases: Relational vs. non-relational databases, basic SQL queries (SELECT, JOIN, GROUP BY)
- Cloud Computing: Basics of cloud services (AWS, Azure, GCP), IaaS, PaaS, SaaS

4. Soft Skills and Project Management Basics (10)

- Team Collaboration: Effective communication, conflict resolution, working in cross-functional teams
- Time Management: Prioritization, meeting deadlines, managing workloads
- Problem Analysis: Root cause analysis, critical thinking, decision-making
- Basic Project Management: Task estimation, risk management, stakeholder communication

5. Core Programming & Algorithms (15)

- OOP Concepts (encapsulation, inheritance, polymorphism, abstraction)
- Data Structures (arrays, linked lists, stacks, queues, hashmaps, trees)
- Algorithm Basics (searching, sorting, recursion)
- Recursion and Dynamic Programming Basics
- Hashing and Trees
- Problem solving with coding

6. Server-Side Development (13)

- Backend Languages (Java/Python/Node.js basics)
- RESTful APIs – design principles
- GraphQL Basics
- Microservices Architecture
- MVC Architecture
- Middlewares in Backend FrameworksE
- Exception Handling and Logging

7. Database & Data Handling (10)

- SQL CRUD Operations, Joins and Subqueries
- ORM frameworks (Hibernate, JPA, Sequelize, etc.)
- Transactions & ACID properties
- NoSQL Databases (MongoDB, Redis basics)
- Database connectivity (JDBC, Node.js drivers)

8. Security & Authentication (12)

- JWT, OAuth2, Session Management
- RBAC methods and Techniques
- Input Validation & Secure Coding Practices
- Common Vulnerabilities (OWASP Top 10)
- SQL Injection and XSS prevention

9. Deployment & Scalability (10)

- Version Control (Git basics)
- CI/CD pipelines (Jenkins, GitHub Actions basics)
- Containerization (Docker fundamentals)
- Cloud Basics (AWS/Azure/GCP overview)
- Load Balancing and Caching

Syllabus for Database Engineer

1. Software Development Fundamentals (10)

- Software Development Life Cycle (SDLC): Waterfall, Agile, Scrum, Kanban
- Software Engineering Principles: Modularity, cohesion, coupling, design patterns
- Version Control Systems: Git (branching, merging, pull requests, conflict resolution)
- Software Testing: Unit testing, integration testing, system testing, test-driven development (TDD)
- Documentation: Writing technical specifications, user manuals, and API documentation

2. Programming and Problem Solving (10)

- Basic Data Structures: Arrays, linked lists, stacks, queues, trees, graphs
- Algorithms: Sorting (e.g., quicksort, mergesort), searching (binary search), recursion
- Problem-Solving: Writing pseudocode, solving algorithmic problems (e.g., time complexity, space complexity)
- Coding Standards: Clean code principles, naming conventions, code refactoring

3. General IT Knowledge (10)

- Operating Systems: Basics of Windows, Linux, and macOS (file systems, processes, permissions)
- Networking: TCP/IP, HTTP/HTTPS, REST APIs, DNS, basic network security
- Databases: Relational vs. non-relational databases, basic SQL queries (SELECT, JOIN, GROUP BY)
- Cloud Computing: Basics of cloud services (AWS, Azure, GCP), IaaS, PaaS, SaaS

4. Soft Skills and Project Management Basics (10)

- Team Collaboration: Effective communication, conflict resolution, working in cross-functional teams
- Time Management: Prioritization, meeting deadlines, managing workloads
- Problem Analysis: Root cause analysis, critical thinking, decision-making
- Basic Project Management: Task estimation, risk management, stakeholder communication

5. Database Foundations (12)

- Database Models and Types
- Relational Database Concepts
- ER Modeling & Normalization (1NF to 3NF, BCNF)
- Keys & Constraints
- Database Design Principles

6. SQL Programming (12)

- DDL, DML, DCL, TCL commands
- Joins, Subqueries, Views, Complex Queries
- Stored Procedures, Functions, Triggers Cursors
- Transactions and Savepoints

7. Database Optimization (12)

- Indexing strategies
- Query Optimization & Execution Plans
- Partitioning, Clustering, Sharding & Replication
- Database Caching Methods
- Performance Tuning

8. Transaction Management & Security (12)

- ACID Properties & Isolation Levels
- Concurrency Control & Deadlocks
- Deadlock Prevention
- Database Security: roles, privileges, encryption

9. Advanced Database Concepts (12)

- Backup & Recovery Strategies
- Data Replication and High Availability
- Data Warehousing Basics
- ETL Concepts
- Introduction to NoSQL (MongoDB, Cassandra, Redis)