Oracle Apex Developer Professional

COURSE OBJECTIVE:	1. **Explore Oracle APEX Fundamentals:** Grasp the foundational concepts of Oracle APEX, including its architecture, workspace management, and development environment. 2. **Develop Advanced APEX Applications:** Design and implement advanced web applications utilizing Oracle APEX's rich set of features. Create complex forms, reports, and interactive grids. 3. **Optimize APEX Applications:** Apply best practices for performance tuning and optimization of APEX applications. Implement efficient data handling and ensure fast application performance. 4. **Enhance Application User Experience:** Utilize APEX's built-in themes, templates, and dynamic actions to enhance the user interface and user experience. Incorporate custom HTML, CSS, and JavaScript to tailor applications to specific needs. 5. **Implement Security Best Practices:** Apply security measures within APEX applications to protect against common web vulnerabilities. Manage authentication and authorization to control access to application components. 6. **Integrate with External Systems:** Connect APEX applications with external data sources and web services. Use RESTful services and APIs to extend application functionality. 7. **Deploy and Manage APEX Applications:** Gain skills in deploying APEX applications across different environments. Perform application backups, version control, and workspace management. 8. **Troubleshoot and Debug APEX Applications:** Develop proficiency in debugging techniques and troubleshooting common issues in APEX applications. Use APEX debug tools and logs to identify and resolve problems.
COURSE OUTCOME:	1. Develop database applications on Oracle APEX using the Oracle Autonomous Database. 2. Utilize SQL Workshop to create and manage database objects.

	3. Build interactive reports, grids, and application page controls. 4. Enhance applications with computations, processes, validations, and dynamic actions. 5. Implement search filtering, form creation, navigation, and theming.

Course Duration: 45 Hours

Course Content:

SL.NO	MODULE NAME	MODULE CONTENT
01	Getting Started with Oracle APEX on the Oracle Autonomous Database	1. "Setup": Provision Autonomous Database. 2. "Access": Connect to Oracle APEX. 3. "Develop": Create apps with APEX. 4. "Deploy": Publish apps online. 5. "Manage": Monitor and maintain apps.
02	Using SQL Workshop	1. Getting started and using SQL workshop components 2. Use SQL commands and SQL scripts 3. Generate SQL by using quick SQL
03	Creating a Database Application	1. App builder concepts 2. App builder and the create app wizard 3. Application based on existing tables 4. Progressive Web App(PWA)
04	Working with pages and regions	1. Creating application pages and exploring page components 2. Managing pages and adding page components
05	Developing reports, managing and customizing interactive reports, grids	1. Learning to create various reports 2. Creating faceted search, cards and smart filters 3. Customising interactive reports as end users and developers 4. Customising an interactive grid

06	Advanced Application Development Techniques	1. Design Patterns 2. Modular Development 3. Test-Driven Development 4. Performance Optimization 5. Continuous Integration/Deployment

Test Projects:

Use Cases

LEARNING OUTCOME	**ASSESSMENT CRITERIA**	**USE CASES**
1. Develop database applications on Oracle APEX using the Oracle Autonomous Database.	1. Evaluate the intuitiveness and responsiveness of the application's interface on different devices. 2. Assess the efficiency of data storage, retrieval, and manipulation within the Oracle Autonomous Database. 3. Ensure protection against common vulnerabilities like SQL injection and cross-site scripting. 4. Measure the application's speed and responsiveness, particularly during heavy usage or data processing. 5. Evaluate the application's ability to handle increasing data volumes and user loads.	**Use Case 1**: Online Retail Management System Scenario: A company wants to develop an online retail management system to handle product inventory, customer orders, and shipping logistics efficiently. They decide to use Oracle APEX with the Oracle Autonomous Database for its scalability and reliability. Tasks: Database Design: Create tables for product inventory, customer information, orders, and shipping details using Oracle APEX's intuitive interface. Define appropriate relationships between tables to ensure data integrity. Customize the interface to meet the specific needs of the retail company. **Use Case 2:** Healthcare Patient Management System Scenario: A healthcare provider needs a robust

		patient management system to store patient records, schedule appointments, and manage medical history securely. They opt for Oracle APEX with the Oracle Autonomous Database to ensure compliance with regulatory requirements and data security standards. Tasks: Develop custom forms and reports in Oracle APEX to capture patient demographics, medical history, and appointment scheduling. Integrate validation rules to ensure data accuracy and consistency throughout the application.
2. Utilize SQL Workshop to create and manage database objects.	1. SQL Workshop interface for creating and managing database objects 2. SQL Workshop- comprehensive range of tools for database object creation and management, including tables, views, and procedures 3. Performance: Evaluate the speed and responsiveness of SQL Workshop when executing commands and managing large datasets to ensure efficiency. 4. Assess the security measures implemented in SQL Workshop to protect sensitive data	**Use Case 1**: Creating Tables for an E-commerce Website Scenario: A startup e-commerce company needs to set up its database to store product information, customer details, and order data. Task: Using SQL Workshop, the developer creates tables for storing product information, customer details, and order data. **Use Case 2**: Managing User Authentication for a Web Application Scenario: A web application requires user authentication to access certain features and data.

	and prevent unauthorized access	Task: Using SQL Workshop, the developer creates and manages database objects for user authentication.
3. Build interactive reports, grids, and application page controls.	1.Evaluate intuitiveness of controls and check responsiveness to user actions. 2. Functionality: Ensure all features work as intended and Verify reports generate accurate data, test grid sorting and filtering functionality. 3. Assess ability to customize report layouts and check options for adding new controls. 4. Measure speed and efficiency and test loading times for large datasets. 5. Ensure compatibility across devices and browsers and test functionality on various browsers.	**Use Case 1:** Sales Dashboard Scenario: A sales manager wants to monitor the performance of the sales team and track key metrics in real-time. Task: Develop an interactive sales dashboard that displays metrics such as total revenue, sales growth, and top-selling products.. Implement interactive charts and graphs to visualize trends and comparisons, enabling the manager to make informed decisions quickly. **Use Case 2:** Inventory Management Grid Scenario: A warehouse supervisor needs to efficiently manage inventory levels and track stock movements. Task: Implement intuitive controls for editing, adding, or removing inventory items directly within the grid, streamlining the inventory management process.
4. Enhance applications with computations, processes,	1.Ensure computations and validations produce correct result and verify outputs against expected values, conduct thorough	**Use Case 1:** Financial Calculation Module Scenario: In a banking application, users need to calculate loan repayment

validations, and dynamic actions.	testing with various input scenarios. 2. Analyze algorithms for time and space complexity. Implement caching and algorithmic improvements where possible. 3. Employ robust error-checking mechanisms and monitor system behavior under stress conditions. 4.Evaluate performance metrics under load testing and implement horizontal and vertical scaling strategies.	schedules based on different interest rates and loan terms. Task: Implement validation checks to ensure user-provided data is within acceptable ranges and perform necessary data sanitization. **Use Case 2:** Real-Time Chat Application Scenario: In a messaging platform, users want to see typing indicators and message delivery statuses in real-time. Task:Implement dynamic actions to display typing indicators when users start typing messages, enhancing the interactive experience.
5. Implement search filtering, form creation, navigation, and theming.	1.Assess the speed and responsiveness of the search filter, ensuring quick retrieval of results even with large datasets. 2.Evaluate the ease of creating forms, including intuitive layout options and clear instructions for input fields. 3. Navigation:Measure how easily users can navigate between different sections or pages of the application, assessing the clarity of navigation menus and buttons.	**Use Case 1:** E-commerce Platform Scenario:A user visits an e-commerce platform and wants to filter products by price range, brand, and availability. Task:Develop a filter sidebar allowing users to refine search results based on various criteria.Design a user-friendly registration form with fields for personal information, address, and account credentials. **Use Case 2:** Content Management System Scenario:An admin needs to navigate through different

	4. Evaluate adherence to accessibility standards (e.g., WCAG), ensuring that all users, including those with disabilities, can access and interact with the application effectively.	sections of a content management system to manage articles, media, and user permissions. Task: Implement a navigation menu with dropdowns or tabs for easy access to different sections of the CMS.
6. Manage application data, and security, and migrate development between environments.	1. Assess how effectively the application handles data storage, retrieval, and manipulation. 2. Evaluate the organization of data structures, database performance, and data integrity measures. 3. Analyze the ease and efficiency of transferring the application between development, testing, and production environments 4. Evaluate the application's ability to handle increased workload or user base without compromising performance.	**Use Case 1:** Secure Data Migration Scenario: During the development phase, the development team needs to migrate sensitive user data from a local development environment to a staging environment hosted on a cloud platform for testing purposes. Task: 1. Implement secure data transfer protocols such as TLS encryption to ensure data confidentiality during migration. 2. Configure access controls and authentication mechanisms to restrict unauthorized access to the migrated data in the staging environment. **Use Case 2:** Database Backup and Recovery Scenario: In the production environment, the application encounters a critical failure resulting in data loss. The development team needs to

		restore the application to a previous state using backup data. Task: 1. Regularly schedule automated database backups to capture the application's current state at defined intervals. 2. Implement a robust backup and recovery strategy, including offsite storage of backups and testing of restoration procedures, to ensure minimal data loss and downtime in the event of a system failure.