Python with SQL

A 3-Month Module-Wise Curriculum

Designed for School Students — Grades 8 to 11
Duration: 12 Weeks | Format: 3 sessions/week, 60–75 minutes each
No prior coding experience required

Total Duration	3 Months (12 Weeks, ~36 Sessions)
Target Audience	Students of Grade 8, 9, 10 and 11
Prerequisites	Basic computer literacy; no prior programming needed
Tools Used	Python 3, VS Code / Jupyter Notebook, SQLite / MySQL
Teaching Approach	Concept + Live Demo + Hands-on Practice + Weekly Mini Project
Final Outcome	Capstone project combining Python and SQL (a working mini application)

Curriculum Snapshot

Week	Focus Area	Key Skill Gained
1-4	Python Programming Foundations	Logic building, syntax, problem solving
5-8	Python for Data Handling + SQL Basics	File handling, functions, database querying
9-12	Python-SQL Integration + Capstone Project	Connecting apps to databases, real project delivery

The course is split into three monthly modules, each with four weekly units. Every week combines a short concept walkthrough, a guided coding demo, and independent practice, ending in a small task students complete on their own.

Month 1 — Module 1: Python Programming Foundations

Goal: Build strong logical thinking and comfort with core Python syntax through simple, relatable examples (games, calculators, quizzes).

Week 1 — Getting Started with Python

- What is programming? Installing Python and setting up the editor
- Writing and running your first program
- Variables, data types (int, float, string, boolean)
- Taking input from users and printing output

Hands-on activity: Build a program that greets the user by name and displays their age next year.
Mini output: “About Me” interactive intro program

Week 2 — Operators and Decision Making

- Arithmetic, comparison, and logical operators
- If, elif, else statements
- Nested conditions
- Building simple decision-based programs

Hands-on activity: Create a program that checks eligibility for a school event based on age and marks.
Mini output: Simple grading system (A/B/C/D based on marks)

Week 3 — Loops and Repetition

- For loops and while loops
- Loop control: break and continue
- Nested loops
- Pattern printing

Hands-on activity: Print number patterns and multiplication tables using loops.
Mini output: Simple guessing game using while loop

Week 4 — Lists, Strings and Basic Functions

- Lists: creating, indexing, slicing, looping through lists
- Common string operations
- Defining and calling functions
- Parameters and return values

Hands-on activity: Write functions to find the largest number in a list and to reverse a string.
Mini output: “Mini Toolkit”: a set of 5 reusable functions

Month 2 — Module 2: Python for Data Handling & SQL Basics

Goal: Move from logic-building to working with structured data, and introduce databases as a natural next step for storing information.

Week 5 — Dictionaries, Tuples and File Handling

- Dictionaries: key-value pairs and common operations
- Tuples and when to use them
- Reading from and writing to text/CSV files
- Organizing student-style records in code

Hands-on activity: Store a class of 10 students (name, marks) in a dictionary and save it to a file.
Mini output: Simple student record file manager

Week 6 — Introduction to Databases and SQL

- Why databases? Files vs databases
- Introduction to SQLite/MySQL and setting up a database
- Tables, rows, columns, data types
- CREATE TABLE and INSERT INTO statements

Hands-on activity: Design and create a database table for a school library or a class record.
Mini output: “School Library” database with sample records

Week 7 — Querying Data with SQL

- SELECT statements and filtering with WHERE
- Sorting with ORDER BY, limiting with LIMIT
- Aggregate functions: COUNT, SUM, AVG, MAX, MIN
- GROUP BY for summarizing data

Hands-on activity: Write queries to find top scorers, subject-wise averages, and book counts by category.
Mini output: Query worksheet: 10 real questions answered in SQL

Week 8 — Updating Data and Multi-Table Basics

- UPDATE and DELETE statements (safe practices)
- Primary keys and foreign keys
- Introduction to JOIN (combining two tables)
- Designing a simple two-table database

Hands-on activity: Create linked “Students” and “Attendance” tables and query them together.
Mini output: Two-table mini database with a working JOIN query

Month 3 — Module 3: Python + SQL Integration and Capstone Project

Goal: Connect Python programs directly to a database, so students can build a working mini application from start to finish.

Week 9 — Connecting Python to SQL

- Using Python's sqlite3 module to connect to a database
- Running SQL queries from inside a Python program
- Fetching and displaying results in Python
- Handling errors gracefully (try/except)

Hands-on activity: Write a Python script that connects to the library database and lists all available books.
Mini output: Python script that reads live data from the database

Week 10 — Building an Interactive Application

- Inserting and updating records from Python (parameterized queries)
- Building a simple text-based menu (Add / View / Search / Delete)
- Input validation basics
- Structuring code into functions for each menu option

Hands-on activity: Extend the library/student system into a menu-driven Python application.
Mini output: Working “Mini App”: add, search, update and delete records

Week 11 — Capstone Project Planning & Development

- Choosing a capstone project (e.g. Student Management System, Library Manager, Expense Tracker, Quiz Score Tracker)
- Planning tables, fields and program flow
- Building the core features
- Testing with sample data

Hands-on activity: Work individually or in pairs to build the chosen project with guided support.
Mini output: 80% complete capstone project

Week 12 — Finishing Touches, Testing and Presentation

- Polishing the menu/output formatting
- Debugging and handling edge cases
- Preparing a short project write-up
- Final presentation and demo to the class

Hands-on activity: Present the completed project: explain the database design and walk through each feature live.
Mini output: Completed, presented capstone project

Sample Capstone Project Ideas

- Student Management System — store, search and update student records with marks and attendance
- School Library Manager — issue/return books, track availability, overdue alerts
- Personal Expense Tracker — log expenses by category and generate monthly summaries
- Quiz Score Tracker — store quiz questions in a database and track player scores
- Simple Contact Book — add, search, edit and delete contacts with SQL storage

Assessment & Evaluation

Component	Weight	What is Assessed
Weekly mini tasks	40%	Completion and correctness of each week's hands-on activity
SQL query worksheet (Week 7)	10%	Accuracy of query writing
Mid-course check (end of Month 2)	15%	Two-table database design + queries
Capstone project	25%	Functionality, database design, code structure
Final presentation	10%	Clarity of explanation and live demo

Batch Suggestions by Grade

Grades 8–9: keep pace steady, spend extra time on logic-building in Month 1, and simplify capstone scope (e.g. Contact Book or Quiz Tracker). Grades 10–11: can move faster through Month 1, and take on richer capstones such as the Student Management System or Expense Tracker with multiple linked tables.

Note: Session count and pacing can be adjusted based on class frequency (e.g. 2 vs 3 sessions per week) without changing the module structure.