School Connect Online

SCO INTERNATIONAL CODING OLYMPIAD

CLASS 5 SYLLABUS

A comprehensive syllabus guide for schools, teachers, parents, and students

Designed from Class 5 coding pathways and aligned with SCO's guided preparation, practice, reporting, and future-ready academic growth.

- age-fit learning guidance for Class 5 / upper-primary coding learners globally
- chapter-wise pathway across programming basics, applications, game development, and latest scenario-based coding thinking
- learning outcomes, classroom guidance, practice direction, and future-benefit framing for academic enrichment
- useful for students, teachers, parents, and schools preparing for the SCO International Coding Olympiad

Basics	Python	Scratch	Game Logic	Projects
Debugging	AI Use	Digital Safety	Coding	Scenarios

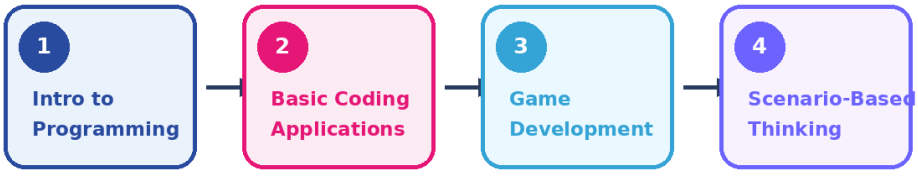
SCO INTERNATIONAL CODING OLYMPIAD

Class 5 Official Syllabus Guide

Purpose of this syllabus

This syllabus introduces coding as a practical, logical, and creative subject for Class 5 learners. It supports classroom teaching, school-level preparation, guided student revision, and Olympiad-style practice through chapter-wise learning outcomes, project ideas, and scenario-based applications.

Class 5 Coding Learning Pathway



Students progress from instructions and data handling to projects, games, debugging, and responsible digital use.

School Connect Online

Syllabus at a Glance

Chapter	Chapter Name	Core Learning Focus	Expected Skill Outcome	Subject / Course
1	**Intro to Programming**	Algorithm and instruction sequence, Variables and value assignment, Integer, float, string, Boolean data	Explain an algorithm as an ordered set of steps for solving a problem.	**Coding Class 5th**
2	**Basic Coding Applications**	Loops and repeated actions, Conditionals and decision-making, Functions and reusable code	Use loops to repeat actions and reduce repeated instructions.	**Coding Class 5th**
3	**Game Development Basics**	Sprites and visual objects, Events and player input, Game loop: input, update, display	Describe sprites, backgrounds, events, scores, levels, and game rules.	**Coding Class 5th**
4	**Theory-Based Inquiries with Latest Scenario Explanations in Coding**	Scenario-based coding reasoning, Responsible use of AI-assisted coding, Privacy and safe data handling	Explain coding concepts using real-world digital scenarios, not only definitions.	**Coding Class 5th**

Learning Goals for Class 5 Coding

Computational Thinking Break a problem into steps, patterns, rules, and repeatable logic.	**Programming Confidence** Read and write simple instructions using variables, loops, conditions, and functions.	**Creative Application** Build or analyse small projects, games, quiz logic, animations, and classroom tools.
Debugging Habit Find errors, test different inputs, and explain how to fix a program.	**Digital Responsibility** Use technology safely, verify outputs, and avoid blind copying from tools or peers.	**Olympiad Readiness** Answer MCQ, output-prediction, error-finding, and scenario-based coding questions confidently.

School Connect Online

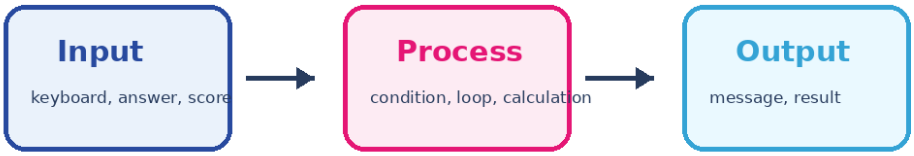
Chapter 1: Intro to Programming

Chapter No.: 1	Subject Name: Coding	Course Name: Class 5th

Chapter Note

This chapter builds the first layer of coding confidence. Students learn that a program is a set of clear instructions, and that computers follow instructions exactly. The chapter introduces algorithms, variables, data types, operators, basic Python-style syntax, and the habit of tracing code before answering.

Programs receive input, process instructions, and produce output.



Learning Outcomes

- Explain an algorithm as an ordered set of steps for solving a problem.
- Identify variables, values, operators, expressions, and common data types such as number, text, and Boolean values.
- Read short Python-style statements and predict simple outputs.
- Recognize common errors such as missing quotes, incorrect variable names, wrong indentation, and type mismatch.
- Use decomposition to break a real-life task into smaller programmable steps.

Key Learning Areas

- Algorithm and instruction sequence
- Variables and value assignment
- Integer, float, string, Boolean data
- Arithmetic and comparison operators
- Input-process-output thinking
- Basic errors and debugging mindset

Suggested Practice and Project Ideas

- Trace a shopping-bill calculator step-by-step before writing any code.
- Convert a daily routine into an algorithm using numbered steps.
- Predict the output of 5 short code snippets involving strings and numbers.

Latest Scenario Explanation

When a student uses an app that counts steps, shows scores, or recommends the next lesson, the app is using stored values, conditions, and calculations. Students learn to see these everyday digital actions as programs made from simple logic.

School Connect Online

Chapter 2: Basic Coding Applications

Chapter No.: 2	Subject Name: Coding	Course Name: Class 5th

Chapter Note

This chapter connects coding concepts to small useful projects. Students practise loops, condition checking, simple functions, lists or collections at a basic level, and Scratch or block-based logic for creating interactive outputs.

Learning Outcomes

- Use loops to repeat actions and reduce repeated instructions.
- Apply if/else conditions to make simple decisions in a program.
- Understand the role of functions as reusable sets of instructions.
- Create or analyse small applications such as quiz logic, score counters, timers, and pattern generators.
- Test small programs with different input values and improve them after finding mistakes.

Key Learning Areas

- Loops and repeated actions
- Conditionals and decision-making
- Functions and reusable code
- Simple counters, quizzes, and calculators
- Scratch events and controls
- Testing with different inputs

Suggested Practice and Project Ideas

- Create a score counter for a classroom quiz using variable logic.
- Design a simple number-checker that decides whether a value is above or below a target.
- Use Scratch-style events to make an object respond when a key is pressed.

Latest Scenario Explanation

A school timetable reminder, a digital quiz, and a simple attendance counter all use basic application logic. Students learn how small pieces of code can support real school and home tasks.

School Connect Online

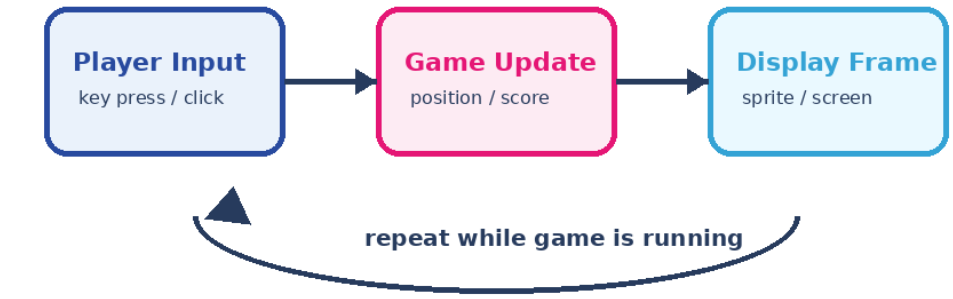
Chapter 3: Game Development Basics

Chapter No.: 3	Subject Name: Coding	Course Name: Class 5th

Chapter Note

This chapter introduces students to games as systems built from rules, objects, inputs, feedback, scoring, and repeated updates. The focus is not only on play, but on designing fair rules, testing interactions, and improving logic.

Game Development Logic



Learning Outcomes

- Describe sprites, backgrounds, events, scores, levels, and game rules.
- Explain how a game loop repeatedly checks input, updates the game state, and displays the result.
- Understand collision detection as checking whether two objects touch or overlap.
- Plan a simple game with a goal, rules, scoring system, and winning or losing condition.
- Improve a game by testing, debugging, balancing difficulty, and explaining design choices.

Key Learning Areas

- Sprites and visual objects
- Events and player input
- Game loop: input, update, display
- Collision detection and scoring
- Levels and difficulty progression
- Animation, sound, feedback, and testing

Suggested Practice and Project Ideas

- Draw a game plan showing the player, goal, obstacle, score, and restart rule.
- Explain what should happen when a sprite touches a wall, coin, or enemy.
- Improve an existing game idea by adding feedback, levels, or a fair scoring rule.

Latest Scenario Explanation

Modern educational games use levels, instant feedback, rewards, and difficulty adjustment. Students learn to think like creators by asking: What is the rule? What happens next? How will the player know the result?

School Connect Online

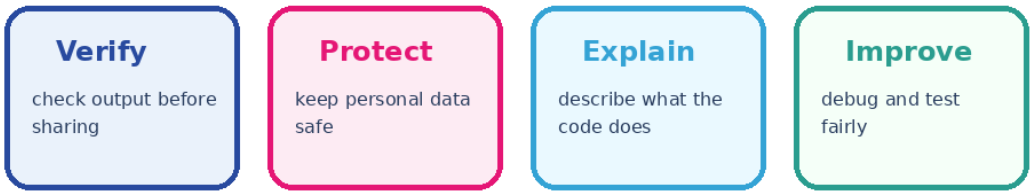
Chapter 4: Theory-Based Inquiries with Latest Scenario Explanations in Coding

Chapter No.: 4	Subject Name: Coding	Course Name: Class 5th

Chapter Note

This chapter prepares students to answer reasoning-based and scenario-based coding questions. Students connect coding with current technology, AI-assisted tools, digital safety, automation, apps, and ethical decision-making.

Responsible Coding in Current Scenarios



Learning Outcomes

- Explain coding concepts using real-world digital scenarios, not only definitions.
- Identify the likely logic behind simple app features such as recommendations, notifications, scores, and filters.
- Discuss responsible coding habits, including testing, privacy awareness, originality, and clear explanation of work.
- Understand that AI tools may suggest code, but students must verify, debug, and understand the output.
- Use structured reasoning to answer latest-scenario inquiry questions in Olympiad format.

Key Learning Areas

- Scenario-based coding reasoning
- Responsible use of AI-assisted coding
- Privacy and safe data handling
- Debugging and verification
- Automation in school and daily life
- Explaining algorithms in simple language

Suggested Practice and Project Ideas

- Read a short app scenario and identify the input, processing logic, and output.
- Compare two possible solutions and choose the safer or clearer one.
- Write a short explanation for why a program output is correct or incorrect.

Latest Scenario Explanation

As AI and smart apps become common in learning tools, students must learn to question results, protect personal information, and explain logic clearly. This chapter builds the habit of being a thoughtful digital creator.

School Connect Online

Recommended Olympiad Preparation Blueprint

Preparation Area	Question Style	Student Should Practise	Weightage Guidance
Intro to Programming	Concept + output prediction	Variables, types, operators, sequence, error spotting	High
Basic Coding Applications	Application-based MCQs	Loops, conditions, functions, small project logic	High
Game Development Basics	Scenario + logic flow	Sprites, events, collision, score, animation loop	High
Latest Scenario Inquiries	Reasoning and safety questions	AI-aware coding, verification, privacy, explainability	Medium-High

Preparation Roadmap for Students, Teachers, and Schools

Step 1: Concept Clarity

- Read chapter notes
- Understand examples before memorising answers
- Use keywords: algorithm, variable, loop, condition, sprite

Step 2: Guided Practice

- Solve topic-wise MCQs
- Trace code manually
- Discuss why wrong options are wrong

Step 3: Project Thinking

- Make or review mini projects
- Explain input-process-output
- Test, debug, and improve logic

Classroom and School Implementation Notes

For Students

- Practise reading code slowly and predicting output before checking the answer.
- Use a notebook to write algorithms, trace tables, and debugging notes.
- Revise vocabulary such as variable, input, condition, loop, sprite, and collision.

For Teachers

- Begin each chapter with one unplugged example and one digital example.
- Use pair discussion for scenario questions so students explain reasoning.
- Assess through concept questions, output prediction, and project explanation.

For Schools

- Use the syllabus as a structured coding pathway for Class 5 learners.
- Encourage project demonstrations, coding clubs, and peer explanation sessions.
- Use SCO practice resources and reports to identify strengths and gaps.

For Parents

- Ask the student to explain what a program does in simple language.
- Encourage safe, age-appropriate digital exploration.
- Value reasoning, debugging, and originality more than speed alone.

School Connect Online

Revision Checklist

Programming Basics

- I can define algorithm and variable.
- I can identify int, float, string, and Boolean values.
- I can predict simple Python outputs.
- I can spot missing quotes, wrong names, and type errors.

Applications and Games

- I can explain loops and if/else logic.
- I can describe Scratch events and blocks.
- I can explain sprites, game loop, score, and collision.
- I can design a small project idea with input, process, and output.

Latest Scenarios

- I can explain how coding is used in apps and smart tools.
- I can describe why AI suggestions must be verified.
- I can protect personal information in digital activities.
- I can answer scenario-based questions with reasons.

Short Glossary for Quick Revision

Term	Simple Meaning for Class 5 Learners
Algorithm	A clear step-by-step method for solving a problem.
Variable	A named container that stores a value used by a program.
Data Type	The kind of value stored, such as number, text, or True/False.
Loop	A structure that repeats instructions.
Condition	A rule that helps a program decide what to do next.
Function	A reusable group of instructions.
Sprite	A character or object in a game or animation.
Collision Detection	Checking whether two objects touch or overlap.
Debugging	Finding and fixing mistakes in a program.
Responsible Coding	Creating and using code safely, fairly, and with clear understanding.

SCO International Olympiad supports structured learning, free preparation resources, practice, reporting, and future-ready academic growth for students.