

KAUR'S CODING ACADEMY

Junior Coding and Game Development - Grades 3 to 5

Module-wise Teacher Syllabus

A friendly beginner course focused on computer confidence, logic building, creative coding, simple games and early programming habits.

Course	Audience	Suggested Duration	Final Output
Junior Coding and Game Development - Grades 3 to 5	Grades 3 to 5 beginners	4 to 6 months / flexible online batch	Animated story, beginner game, first web page and parent demo project

Teacher Note

Use simple examples, visuals, stories and very small steps. Young learners should see a working output in every class.

Learning Outcomes

- Understand basic computer use, files, folders, keyboard practice and online safety.
- Build logic using sequences, events, conditions, loops and debugging activities.
- Create animations, stories, quizzes and simple games using block coding and beginner-friendly tools.
- Create a first simple web page and a small Python art or quiz activity.
- Present a final project confidently to parents or classmates.

Document Use

This PDF is made for teachers to plan classes module by module. Each module includes topic names, teaching points, class practice and revision work.

Complete Module Index

The table below gives a quick view of what will be taught in this course.

Module	Topic Name	Main Things Teacher Will Cover
1	Computer Basics and Online Class Discipline	Keyboard and mouse/trackpad practice, Opening browser, tabs and links safely, Files, folders, saving work and naming files, Online class rules: mute, chat, screen share and homework submission, and more
2	Logic, Sequencing and Algorithms	Step-by-step thinking, Daily-life algorithms, Order of instructions, Cause and effect, and more
3	Flowcharts and Problem Breakdown	Start and end symbols, Process steps, Decision boxes, Yes/No thinking, and more
4	Block Coding Environment	Sprite/character selection, Stage/background setup, Events and actions, Motion blocks, and more
5	Events, Motion and Animation	Click events, Key press events, Move, turn and glide actions, Costume changes, and more
6	Loops, Conditions and Score	Repeat loops, Forever loops, If conditions, If/else decisions, and more
7	Interactive Story Project	Story planning, Characters and dialogue, Background changes, Sound effects, and more
8	Beginner Web Page with HTML	What is a web page?, Headings and paragraphs, Images and links, Simple colors and text styling, and more
9	Python Fun with Turtle and Print	Print statements, Simple variables, Input questions, Numbers and text, and more
10	Quiz and Mini App Logic	Question-answer flow, If/else checking, Score counting, Simple feedback messages, and more
11	Beginner Game Development Project	Game goal, Player and obstacle/enemy, Movement controls, Collision rule, and more
12	Final Showcase and Parent Demo	Project polishing, Naming project properly, Explaining code in simple words, Demo practice, and more

Detailed Module-wise Teaching Plan

Teachers should first explain the concept in simple language, then demonstrate code, then allow students to practice and customize the output.

Module 1: Computer Basics and Online Class Discipline

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Keyboard and mouse/trackpad practice - Opening browser, tabs and links safely - Files, folders, saving work and naming files - Online class rules: mute, chat, screen share and homework submission - Basic internet safety and responsible technology use	- Explain how computers follow instructions step by step - Show how to create a folder for coding work - Demonstrate safe browsing and how to save class files properly
Class Practice / Student Output	Homework / Revision
- Create a personal coding folder - Type a short introduction paragraph - Save a screenshot or small note file	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 2: Logic, Sequencing and Algorithms

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Step-by-step thinking - Daily-life algorithms - Order of instructions - Cause and effect - Debugging as finding and fixing mistakes	- Explain algorithms through brushing teeth, making sandwich and school routine examples - Show how wrong order creates wrong output - Use puzzle activities before starting code
Class Practice / Student Output	Homework / Revision
- Write steps for a daily activity - Arrange shuffled instructions in correct order - Find mistakes in a simple instruction set	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 3: Flowcharts and Problem Breakdown

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Start and end symbols - Process steps - Decision boxes - Yes/No thinking - Breaking a big task into small tasks	- Explain flowcharts with traffic light and quiz examples - Show how decisions become if/else logic later - Help students convert a story into a flowchart
Class Practice / Student Output	Homework / Revision
- Draw a flowchart for a simple quiz - Create a flowchart for a game rule - Explain the flowchart in class	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 4: Block Coding Environment

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Sprite/character selection - Stage/background setup - Events and actions - Motion blocks - Looks and sound blocks - Saving and opening projects	- Introduce the interface slowly and explain every main area - Demonstrate how blocks connect like puzzle pieces - Show how an event starts an action
Class Practice / Student Output	Homework / Revision
- Make a sprite move and speak - Add a background and sound - Create a 20-second animated scene	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 5: Events, Motion and Animation

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Click events - Key press events - Move, turn and glide actions - Costume changes - Basic animation timing	- Explain that events are triggers - Show movement using arrow keys - Demonstrate costume change to create animation
Class Practice / Student Output	Homework / Revision
- Create a dancing character - Move a sprite with keyboard keys - Make a short animated story	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 6: Loops, Conditions and Score

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Repeat loops - Forever loops - If conditions - If/else decisions - Score variable - Timer and lives	- Explain loops as repeating tasks - Explain conditions as yes/no questions - Show how score increases when player catches an object
Class Practice / Student Output	Homework / Revision
- Build a catch game with score - Add timer or lives - Test and improve game difficulty	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 7: Interactive Story Project

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Story planning - Characters and dialogue - Background changes - Sound effects - Scene order - Simple user interaction	- Teach students to plan before coding - Demonstrate scene-by-scene story development - Show how user clicks can change story direction
Class Practice / Student Output	Homework / Revision
- Create a 3-scene animated story - Add at least two characters - Present the story flow	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 8: Beginner Web Page with HTML

Topics to be Covered	What Teacher Will Explain / Demonstrate
- What is a web page? - Headings and paragraphs - Images and links - Simple colors and text styling - Page about self, hobby or favorite topic	- Explain HTML as the structure of a page - Show how tags open and close - Demonstrate a simple page in browser
Class Practice / Student Output	Homework / Revision
- Create an About Me page - Add one image and one link - Change heading and paragraph text	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 9: Python Fun with Turtle and Print

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Print statements - Simple variables - Input questions - Numbers and text - Turtle drawing basics - Loops for shapes	- Explain Python as text-based instructions - Show print and input using friendly examples - Demonstrate Turtle drawing step by step
Class Practice / Student Output	Homework / Revision
- Print name and favorite subject - Draw a square or star with Turtle - Change colors and shape size	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 10: Quiz and Mini App Logic

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Question-answer flow - If/else checking - Score counting - Simple feedback messages - Restart idea	- Explain how apps ask input, check it and give output - Show score logic with easy examples - Help students make their own quiz questions
Class Practice / Student Output	Homework / Revision
- Build a 5-question quiz - Add score at the end - Customize quiz topic	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 11: Beginner Game Development Project

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Game goal - Player and obstacle/enemy - Movement controls - Collision rule - Score, lives and restart - Testing and improvement	- Show how every game has rules and feedback - Demonstrate collision and score changes - Guide students to test the game like players
Class Practice / Student Output	Homework / Revision
- Build a maze, catch or avoider game - Add win/lose message - Improve design and difficulty	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Module 12: Final Showcase and Parent Demo

Topics to be Covered	What Teacher Will Explain / Demonstrate
- Project polishing - Naming project properly - Explaining code in simple words - Demo practice - Feedback and improvement	- Teach students how to present what they built - Ask simple questions about events, loops and conditions - Help students improve confidence
Class Practice / Student Output	Homework / Revision
- Prepare one final project demo - Show project to parent/class - Answer two questions about how it works	- Revise the class notes and complete a small improvement in the class project. - Write down two doubts or two new things learned in the module.

Final Review and Certification Checklist

- Student can explain the main concepts in simple words.
- Student has completed the required practice tasks and mini projects.
- Student can run the project, fix common errors and present the work confidently.
- For advanced courses, student should submit GitHub repository, screenshots and deployment link where possible.