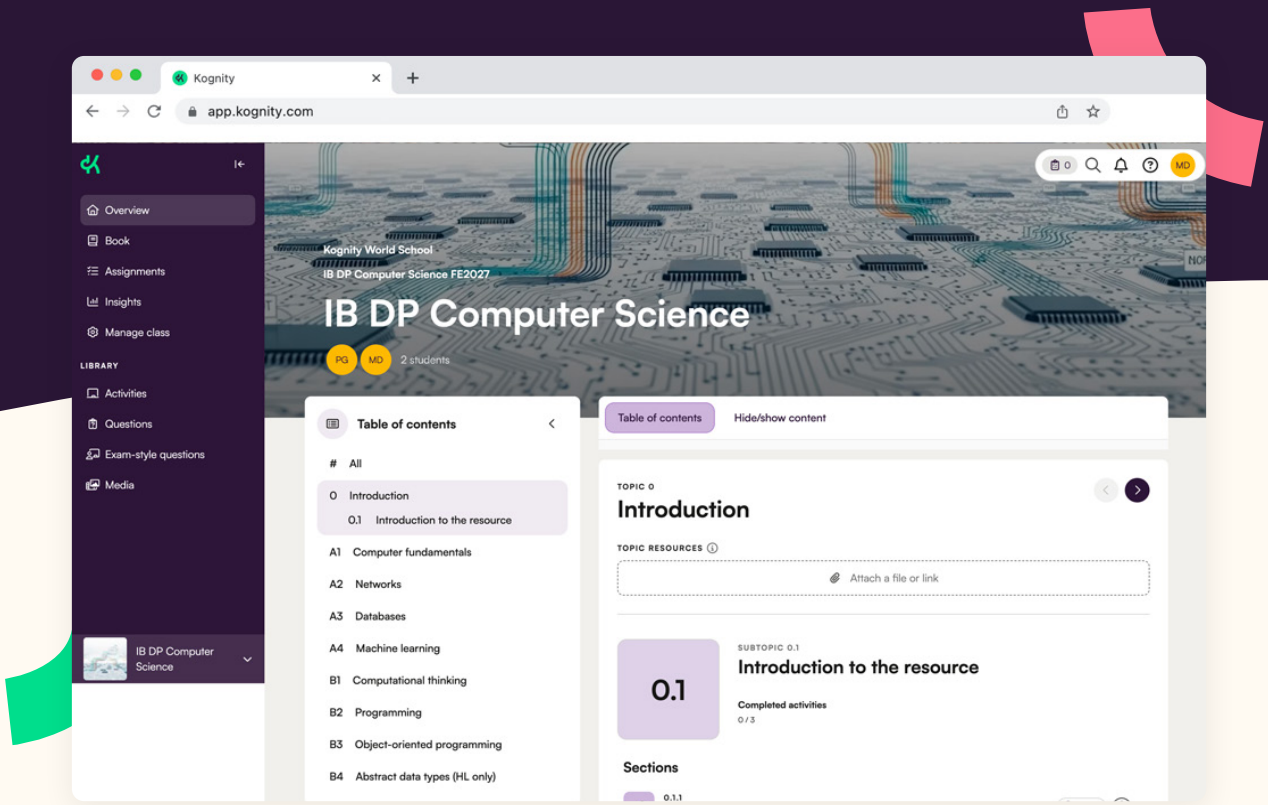
IBDP Computer Science

Kognity's IBDP Computer Science SL and HL subject supports the DP course through a fully syllabus-aligned digital book, classroom assignments, assessment practice and progress data. The resources combine conceptual understanding with practical problem solving, programming and reflection.



The book

Syllabus organisation

The book is organised into topics and subtopics that make the course easy to navigate and teach in manageable sections.

- **Clear learning structure:** Each subtopic includes a guiding question, prior learning, learning objectives and inquiry questions.
- **SL and HL visibility:** HL-only content is clearly identified so teachers can plan common learning and targeted extension.

- **Progressive sequencing:** Concepts are introduced through real-world contexts before students apply precise Computer Science terminology and techniques.

A1 Computer fundamentals

- A1.1 Computer hardware and operation
- A1.2 Data representation and computer logic
- A1.3 Operating systems and control systems
- A1.4 Translation (HL only)

Course content

Students learn Computer Science through real-world situations, visual explanations, worked examples and structured activities.

- **Authentic contexts:** Examples such as emergency transport systems, medical supply tracking and global online stores show how computing responds to human needs.
- **Programming support:** Java and Python examples allow students to compare how the same concepts are implemented in different languages.
- **Conceptual and practical learning:** Students construct problem specifications, trace algorithms, interpret diagrams, compare solutions and write or analyse code.
- **Accessible reading tools:** The glossary, notebook, dictionary and Immersive Reader support subject vocabulary, active reading and language development.

Task 2

Rewrite the function so that it updates the global variable instead of creating a local variable with the same name.

Java

```java
public class WaterSystem {
    static int waterLevel = 100;

    public static void waterPlant() {
        int waterLevel = 50;
        waterLevel = waterLevel - 10;
    }

    public static void main(String[] args) {
        waterPlant();
        System.out.println(waterLevel);
    }
}
```

Python

```python
water_level = 100

def water_plant():
    water_level = 50
    water_level = water_level - 10

water_plant()
print(water_level)
```

Practical skills and Internal Assessment preparation

Activities throughout the book develop habits that transfer directly to practical programming and the Internal Assessment process.

- Define a problem and identify objectives, constraints, inputs, outputs and measurable evaluation criteria.
- Apply decomposition, pattern recognition, abstraction and algorithmic thinking before implementation.
- Justify design decisions, test solutions, evaluate outcomes and communicate improvements clearly.
- Use individual, pair and extension activities to move from guided practice towards independent solution development.

TOK, ethics and international-mindedness

Computer Science is presented as a human discipline whose solutions affect people, communities and environments.

- Theory of Knowledge prompts invite students to question objectivity, classification and the assumptions built into computational models.
- International examples encourage students to consider language, access, infrastructure, privacy, safety and cultural context.
- Making Connections sections link Computer Science to subjects such as Business Management, Design Technology and database design.

Theory of knowledge

If knowledge requires not merely a correct conclusion but a justifiable path to that conclusion, then the opacity of deep learning models forces a genuine philosophical problem. A model trained for medical diagnosis may match the accuracy of a trained clinician on specific tasks, but its internal decision-making process remains opaque, even to its architects. This raises a pointed epistemological question: can a system be said to *know* something if it cannot account for how it arrived at that conclusion? And crucially, does the absence of explicable reasoning disqualify a correct judgement from counting as knowledge at all?

Assignments

For teachers

Assign readings, section questions and exam-style questions directly from the platform. Set instructions, recipients and deadlines, then monitor completion and performance.

For students

Receive assignments in one place, complete readings and questions within Kognity, and view scores and feedback after submission.

The Practice Centre

Targeted independent practice

The Practice Centre gives students structured opportunities to revisit content, test understanding and prepare under increasing time pressure.

- Strength tests help students identify secure and less secure areas of the course.
- Strength battles add a collaborative element to retrieval practice.
- Exam-style questions allow students to practise applying knowledge in unfamiliar contexts.
- The strength bar helps students monitor progress and select their next revision priority.

Insights

Actionable progress data

The Insights page brings together textbook and question data so teachers can see patterns across a class and respond efficiently.

- Identify students who need support with a particular subtopic or skill.
- Review quiz, assignment and practice performance before moving to new content.
- Plan whole-class reteaching, small-group intervention or individual extension from evidence rather than guesswork.