

All About IT Free Course Guide

A free learning roadmap for IT students.

Eight modules in order, one hands-on project in every module, and a directory of genuinely free courses you can open on a phone or a computer. Save this file, follow it week by week, and build something at every step.

```
$ whoami
future_it_student
$ start --module 1
> Loading Computer Basics & Computer Systems ... ready.
> ERROR 404: sleep not found (keep going, one module at a time)
```

Who this is for

Beginners, senior-high and college students, and self-learners - anyone starting in information technology, with no experience assumed.

What you need

A phone or computer, an internet connection, and a few hours a week. Every course listed in this guide is free to access.

How to use this guide

- **Go in order.** Modules 1 to 8 build on each other, like a real course.
- **Do the project.** Reading alone does not stick - building does.
- **Tick the tracker** on the last page as you finish each module.
- **Check certificates honestly.** Page 7 labels every course: free certificate, paid verified certificate, or no certificate.

The roadmap: 8 modules in order

One module is roughly one to three weeks at a few hours a week. There is no deadline - finish a module properly before moving on.

#	Module	You will be able to	Project
1	Computer Basics & Computer Systems	Explain what a computer is, what each part does, and how data moves inside it.	Label a computer system
2	Internet & Networking	Explain how the internet works, from a browser to a server and back.	Map a home network
3	Coding with Python	Write small programs using variables, conditions, loops and functions.	Number guessing game
4	Web: HTML, CSS, JavaScript	Build and style a web page and make it respond to clicks.	Personal profile page
5	Databases & SQL	Store data in tables and answer questions with SQL queries.	Student records database
6	Operating Systems	Use files, processes and the command line with confidence.	Command-line file cleanup
7	Cybersecurity & Ethical Hacking	Explain common threats, defence basics, and what ethical hackers legally do.	Home security checklist
8	Next Steps	Compare IT fields and choose a degree or self-taught path.	One-page study plan

A simple weekly rhythm

Learn - 2 sessions

Watch or read the lessons for the module. Take short notes in your own words.

Practice - 1 session

Redo the examples yourself. Change the code or settings and see what happens.

Build - 1 session

Finish the module project, even if it is small or imperfect.

Review - 15 minutes

Re-read the matching cheat sheet on pages 8-9 and tick the tracker.

The one rule that matters

Do not collect courses. Finish one module, build its project, then move on. A small finished project teaches more than five half-watched courses.

① Computer Basics & Computer Systems

What you will learn: A computer takes input, processes it, stores data, and gives output. This module names every part in that loop so later modules make sense.

Key topics

- Hardware vs software; input, processing, storage and output
- Central Processing Unit (CPU), Random Access Memory (RAM), storage (HDD and SSD), motherboard, power supply
- Binary: why computers count in 0s and 1s (cheat sheet, page 9)
- Files, folders and common file extensions (page 9)
- How a computer boots, step by step

Project: Draw or photograph a computer or phone and label its CPU, memory, storage, input and output parts. Write one sentence for each: what it does.

Free courses: Khan Academy - Computing (how computers and the internet work) and Cisco Skills for All - Computer Hardware Basics. Directory links: page 7.

② Internet & Networking

What you will learn: How devices talk to each other, and what really happens between typing a web address and seeing a page.

Key topics

- Local Area Network (LAN) vs Wide Area Network (WAN); router, switch, modem, Wi-Fi
- Internet Protocol (IP) addresses and Domain Name System (DNS) - the internet's phone book
- Client and server; request and response; HTTP and HTTPS
- Packets, and a simple picture of the OSI / TCP-IP model in plain words
- Bandwidth, latency and basic troubleshooting: restart, check cables, isolate the problem

Project: Map a real network you use: devices, router, and connection to the internet. Label each device's job, then trace one request - for example, opening a website - through the map.

Free courses: Cisco Skills for All - Networking Basics, and Khan Academy - Computing (how the internet works). Directory links: page 7.

```
browser --> internet --> server
request <-- data travels in packets --> response
```

③ Coding with Python

What you will learn: Programming is giving a computer step-by-step instructions. Python is a good first language because its syntax reads close to plain English.

Key topics

- Variables and data types: text (string), whole numbers (integer), decimals (float), true/false (boolean)
- Input and output: `input()` and `print()`
- Conditions: `if`, `elif`, `else`; loops: `for` and `while`
- Functions: grouping steps you reuse; lists and dictionaries for collections
- Reading errors calmly: find the line, read the message, fix one thing at a time

Project: A number guessing game - the program picks a secret number, gives higher/lower hints, and counts attempts. Full starter logic is on the Python cheat sheet, page 8.

Free courses: freeCodeCamp - Python courses and certifications; Harvard CS50x for a broader computer science foundation; W3Schools - Python Tutorial for quick practice. Directory: page 7.

④ Web: HTML, CSS, JavaScript

What you will learn: Every web page is three layers: HTML for structure, CSS (Cascading Style Sheets) for presentation, and JavaScript for behaviour.

Key topics

- HTML: headings, paragraphs, links, images, lists, forms, semantic tags
- CSS: selectors, colours, fonts, spacing, the box model, flexbox for layout
- JavaScript: variables, events (clicks), changing page content
- Front-end (what the user sees) vs back-end (server and database)
- Responsive design: a page that works on a phone and a computer

Project: A one-page profile site about a topic you study: heading, short sections, one image with alt text, styled with CSS, and one button that changes something with JavaScript.

Free courses: freeCodeCamp - Responsive Web Design; MDN - Learn Web Development; W3Schools - HTML, CSS and JavaScript Tutorials. Directory: page 7.

5 Databases & SQL

What you will learn: A database stores organised data in tables - rows and columns. SQL (Structured Query Language) is the standard language for asking a database questions and changing its data.

Key topics

- Tables, rows, columns, and one primary key that identifies each row
- `SELECT` , `WHERE` , `ORDER BY` - reading data
- `INSERT` , `UPDATE` , `DELETE` - changing data, carefully
- Joining two tables with `JOIN` ; grouping with `GROUP BY`
- Beginner-friendly systems: SQLite and MySQL

Project: A student records database: one table for students, one for subjects, and five queries that answer real questions (for example, all students in one subject, sorted by name). SQL cheat sheet: page 8.

Free courses: freeCodeCamp - Relational Database; W3Schools - SQL Tutorial; Khan Academy - Computing (introductory SQL). Directory: page 7.

6 Operating Systems

What you will learn: An operating system (OS) - such as Windows, Linux, macOS, Android or iOS - manages hardware, runs programs, organises files, and handles users and permissions.

Key topics

- What an OS does: process, memory, file and device management
- Files and folders, paths, and file permissions
- Graphical interface vs command line; basic commands: `cd` , `ls` / `dir` , `mkdir` , `cp` / `copy` , `mv` / `move`
- Installing and updating software safely; user accounts
- Why Linux matters in IT: most servers run it

Project: Using only the command line, create a tidy folder structure for one term of study, move example files into it, and write down each command you used.

Free courses: Cisco Skills for All - introductory Linux and operating system courses in its catalog; freeCodeCamp lessons that use Linux and Git. Directory: page 7.

7 Cybersecurity & Ethical Hacking

What you will learn: Cybersecurity protects systems and data. Ethical hacking is the legal side of security testing: professionals test systems only with the owner's permission, then help fix what they find.

Key topics

- Common threats in plain words: phishing, malware, weak or reused passwords
- Defence basics: strong unique passwords, a password manager, two-factor authentication (2FA), updates, backups
- White hat (authorised testing), black hat (criminal attacks) - and why permission is the line between them
- Legal practice only: your own devices, your own lab, and organised Capture The Flag (CTF) practice platforms
- Important: accessing a system or account without permission is a crime in many countries. This guide teaches concepts and defence only - no attack steps.

Project: A home security checklist: update every device, turn on 2FA where available, check password reuse, and set up a backup. Write what you changed and why.

Free courses: Cisco Skills for All - Introduction to Cybersecurity; freeCodeCamp - Information Security. Directory: page 7.

8 Next Steps: Choosing a Field

What you will learn: IT is not one job. This module compares the main fields so you can choose what to study next, at a school or on your own.

Two valid paths

- **Degree path:** A program such as Information Technology, Computer Science or Computer Engineering gives structure, teachers, classmates and a credential. Compare subjects, not just school names.
- **Self-taught path:** The free courses in this guide, a portfolio of projects, and free certificates where they exist. It needs more self-discipline, and the projects are the proof.
- **Mixed path:** Study formally and keep building projects outside class. Most working IT professionals keep learning either way.

Project: A one-page study plan for your next three months: one field (page 9), one course (page 7), and one project per month.

IT vs Computer Science vs Computer Engineering, in one line each: IT focuses on using and supporting technology in organisations; Computer Science focuses on computing theory and software; Computer Engineering focuses on hardware and electronics together with software.

Free courses directory

Every provider below was checked against its official page in October 2026. **Certificate labels are honest:** read them before you start, because free learning and a free certificate are not always the same thing.

Provider / course	Best for	Official link	Certificate
freeCodeCamp Core curriculum and certifications	Web, Python, JavaScript, SQL, security - Modules 3, 4, 5, 7	freecodecamp.org	Free certificate - official site: courses, projects and certifications are 100% free
Harvard CS50x Introduction to Computer Science	Computer science foundation - Modules 1, 3, 5	cs50.harvard.edu/x	Course free via OpenCourseWare. Verified certificate via edX is paid - check the edX enrolment page
Khan Academy Computing	Gentle introductions - Modules 1, 2, 5	khanacademy.org/computing	No certificate - free lessons and practice only
MDN Learn Web Development (Mozilla)	Web done properly - Module 4	developer.mozilla.org - Learn web development	No certificate - free structured lessons and reference
W3Schools Tutorials and references	Quick lessons and try-it practice - Modules 3, 4, 5	w3schools.com	Tutorials free . Certificates are separate paid products - you do not need one to learn
Cisco Skills for All Networking Academy free courses	Networking, hardware, Linux, security - Modules 1, 2, 6, 7	skillsforall.com	Free courses . A certificate of completion / digital badge is offered on selected courses - check each course page

Organised by topic: **Foundations** - Khan Academy Computing, CS50x. **Networking and hardware** - Cisco Skills for All. **Coding** - freeCodeCamp, W3Schools. **Web** - MDN Learn, freeCodeCamp. **Databases** - freeCodeCamp, W3Schools SQL. **Security** - Cisco Introduction to Cybersecurity, freeCodeCamp Information Security.

Cheat sheet: Python basics

```
name = "Alex"          # variable (string)
age = 16                # integer
print("Hello", name)   # output

score = int(input("Score: ")) # input as a number
if score >= 75:
    print("Passed")
elif score >= 60:
    print("Almost there")
else:
    print("Try again")

for i in range(5):     # loop: 0, 1, 2, 3, 4
    print(i)

def greet(person):     # function
    return "Hi " + person
print(greet("Alex"))

fruits = ["apple", "mango"] # list
student = {"name": "Alex", "age": 16} # dictionary
```

Guessing game idea (Module 3): store a secret number, loop while the guess is wrong, print "higher" or "lower", count the attempts, and stop when the guess matches.

Cheat sheet: SQL basics

```
SELECT name, subject FROM students;           -- read columns
SELECT * FROM students WHERE subject = 'IT';   -- filter rows
SELECT * FROM students ORDER BY name;         -- sort
INSERT INTO students (name, subject)
VALUES ('Alex', 'IT');                         -- add a row
UPDATE students SET subject = 'CS'
WHERE name = 'Alex';                          -- change (always use WHERE)
DELETE FROM students WHERE name = 'Alex';      -- remove (always use WHERE)
SELECT s.name, t.title
FROM students s JOIN subjects t
ON s.subject_id = t.id;                       -- combine tables
```

Warning worth remembering: `UPDATE` and `DELETE` without `WHERE` affect every row in the table.

Cheat sheet: Binary

Computers store everything as bits - 0 or 1. Place values double from the right: 128, 64, 32, 16, 8, 4, 2, 1.

Decimal	0	1	2	3	5	10	65	255
Binary	0	1	10	11	101	1010	01000001	11111111

Worked example: $65 = 64 + 1$, so bits 64 and 1 are on: 01000001. In the ASCII code, 01000001 is also the letter "A", and 01000010 (66) is "B".

Cheat sheet: Common file extensions

Extension	What it is	Extension	What it is
.txt	Plain text	.jpg / .png	Image
.pdf	Document (this file)	.mp3 / .mp4	Audio / video
.docx	Word document	.zip	Compressed folder
.xlsx	Spreadsheet	.html / .css / .js	Web page / style / script
.py	Python program	.exe	Windows program - only run files you trust

IT fields overview

Field	What the job does, in one line
Programming / Software Development	Writes and tests the programs and apps people use.
Web Development	Builds and maintains websites and web applications.
Networking	Sets up and keeps the connections between computers working.
Cybersecurity	Protects systems and data, and responds when something goes wrong.
Databases	Designs and maintains the tables where organisations store their data.
IT Support	Helps users solve computer problems and keeps everyday systems running.

Progress tracker

- ☐ Module 1 - Computer Basics & Computer Systems - project done
- ☐ Module 2 - Internet & Networking - project done
- ☐ Module 3 - Coding with Python - project done
- ☐ Module 4 - Web: HTML, CSS, JavaScript - project done
- ☐ Module 5 - Databases & SQL - project done
- ☐ Module 6 - Operating Systems - project done
- ☐ Module 7 - Cybersecurity & Ethical Hacking - project done
- ☐ Module 8 - Next Steps - study plan written

Sources

This guide was checked against the official pages of the providers it lists:

- freeCodeCamp - official site, freecodecamp.org (free courses, projects and certifications)
- Harvard CS50 - official course site, cs50.harvard.edu/x/ (CS50x course content and how to take it; verified certificates are handled by edX)
- Khan Academy - khanacademy.org/computing (Computing courses)
- MDN Web Docs, Mozilla - developer.mozilla.org, Learn Web Development
- W3Schools - w3schools.com, About page (tutorial list and free developer resource statement)
- Cisco Networking Academy / Skills for All - skillsforall.com and Cisco's official announcements of the free Skills for All program

Closing note

All courses listed in this guide were free to access at the time of writing (October 2026). Course details, catalogs and certificate rules can change, and a free course does not always include a free certificate - always check the official course page before you start, and never pay for a certificate you do not need.

Share this guide freely with other students. The next step is simple: open Module 1, start the first course, and build the first project.

```
$ course --start
> Module 1 loading... good luck, and build as you learn.
```