



# VISHNU INSTITUTE OF TECHNOLOGY (AUTONOMOUS)

## Department of Computer Science and Engineering

### Step-by-Step Path to Improve Programming Skills in Java / Python

#### Objective

This freely accessible course is designed to guide students systematically in building strong programming foundations in Python and Java. It uses publicly available platforms for hands-on practice, structured problem-solving, and conceptual understanding, aiming for measurable improvement in coding skills. It is intended for students of engineering colleges and can be easily adopted or replicated by other institutions.

#### Course Structure

##### Phase 1: Foundational Programming Skills

###### 📖 For Python Beginners

Platform: Snakify.org - <https://snakify.org>

- Complete all lessons and exercises in sequence.
- Focus areas: variables, loops, conditionals, strings, functions, and lists.
- Target: Develop syntax familiarity and problem-solving using basic constructs.

###### 📖 For Java & Python Practice

Platform: CodingBat.com - <https://codingbat.com>

For Python:

- Solve at least 5 problems from each of the following sections:
  - String-1
  - List-1
  - Logic-1
  - Warmup-1

For Java:

- Solve at least 5 problems from each of the following sections:
  - String-1
  - Array-1
  - Logic-1

- Warmup-1
- Recursion

## **Phase 2: Guided Contests on HackerRank**

Students should now move to solving structured contests created on HackerRank, arranged in increasing order of difficulty.

### **📖 Beginner Contests**

Beginner Day 1: <https://www.hackerrank.com/thirdbeginnersday1>

Python Begin 1F: <https://www.hackerrank.com/pythonbegin1f>

Python Begin 2F: <https://www.hackerrank.com/pythonbegin2f>

Python Begin 3F: <https://www.hackerrank.com/pythonbegin3f>

Python Begin 4F: <https://www.hackerrank.com/pythonbegin4f>

Python Begin 5: <https://www.hackerrank.com/pythonbegin5>

Python Begin 6: <https://www.hackerrank.com/pythonbegin6>

Python Begin 7: <https://www.hackerrank.com/pythonbegin7>

Python Begin 8: <https://www.hackerrank.com/pythonbegin8>

Python Begin 9: <https://www.hackerrank.com/pythonbegin9>

### **📖 Intermediate Contests**

Middle 1: <https://www.hackerrank.com/middle1>

Middle 2: <https://www.hackerrank.com/middle2>

Middle 3: <https://www.hackerrank.com/middle3>

Middle 4: <https://www.hackerrank.com/middle4>

### **📖 Advanced DSA Contests**

Linked List 108: <https://www.hackerrank.com/linkedlist108>

Stacks 108: <https://www.hackerrank.com/stacks108>

Trees Revision: <https://www.hackerrank.com/treesrevision>

Trees 108: <https://www.hackerrank.com/trees108>

Graphs 108: <https://www.hackerrank.com/graphs108>

### **Phase 3: Object-Oriented Programming**

Java OOPS: <https://www.hackerrank.com/javaoops>

Java OOPS - New Problems: <https://www.hackerrank.com/javaoopsnew>

### **Phase 4: Company-Specific Practice Sets - Wipro**

Wipro Round 1: <https://www.hackerrank.com/wipronlth1>

Wipro Round 2: <https://www.hackerrank.com/wipronlth2>

Wipro Round 3: <https://www.hackerrank.com/wipronlth11>

### **Phase 4: Company-Specific Practice Sets - Accenture**

Accenture 108: <https://www.hackerrank.com/accenture108>

Accenture Additional: <https://www.hackerrank.com/accent108>

Daily Practice 1: <https://www.hackerrank.com/accentdaily1>

Daily Practice 2: <https://www.hackerrank.com/accentdaily2>

Daily Practice 3: <https://www.hackerrank.com/accentdaily3>

Daily Practice 4: <https://www.hackerrank.com/accentdaily4>

### **Phase 5: Striver's DSA Sheet**

After gaining comfort with contests, students should begin solving problems from Striver's DSA Sheet (<https://takeuforward.org/interviews/strivers-sde-sheet-top-coding-interview-problems/>), which links to LeetCode problems grouped by topic.

### **Video Solutions for Guidance**

#### **Video Solutions**

Introduction to Linked Lists: <https://youtu.be/xHAbJDYNcpl>

Counting Commas: [https://youtu.be/p5\\_QPuSsnuU](https://youtu.be/p5_QPuSsnuU)

Pronic Number: <https://youtu.be/XRVaYgPIJWQ>

Unique Sequence: [https://youtu.be/W\\_xgavwnTIU](https://youtu.be/W_xgavwnTIU)

Interchange Digits: [https://youtu.be/hqU\\_GicWPjk](https://youtu.be/hqU_GicWPjk)

Championship Problem: <https://youtu.be/VGkP9Dffzmk>

Longest Palindromic Substring: <https://youtu.be/G7ns8TYvznQ>

Graphs – Topological Sort, Bipartite: <https://youtu.be/cJ0vRtTLNU4>

Dynamic Programming – LCS, Edit Distance: <https://youtu.be/PsLznGqIbBI>

### **Benefits of this Model**

- Cost-effective: Uses free resources (HackerRank, Snakify, CodingBat, YouTube).
- Structured progression: Moves from syntax to real-world coding challenges.
- Reproducible: Easily implementable across institutions.
- Peer-reviewable: Open for suggestions and adaptation.

### **How to Use this Course**

- This material can be hosted on a college LMS, website, or shared as a PDF.
- Students can mark their progress using spreadsheets or shared Google Sheets.
- Teachers can track HackerRank submissions or ask for screenshots/submission links.
- Faculty members from other institutions can adopt this model and customize it.