

Proposal for a New Course  
Department of Intelligent Systems  
Wadhvani School of AI and Intelligent Systems  
Indian Institute of Technology, Kanpur

**Course number:** ESOXXX

**Course title:** Fundamentals of Data Engineering – Part A

**Course prerequisites:** None

**Course credits:** [7] (3-1-3-0)

**Course duration:** Modular (1st half)

**Course type:** ESO (for all UG students)

**Proposing instructor:** Indranil Saha

**Others interested in teaching the course:** Purushottam Kar, Sruti Srinivasa Ragavan, Nisheeth Srivastava, and all DIS faculty members

**Course description:**

- a. *Objectives:* Python is the contemporary choice of programming language for handling data and building AI/ML applications. This course will introduce students to the basics of Python programming. The course will also offer hands-on experience in the form of labs.
- b. *Content:* There will be an equivalent of 21 lectures of 50 minutes each and 7 labs of 3 hours each. A weekly breakup of lecture and lab content is given below.
- c. *Evaluation:* Evaluation will use a combination of graded lab exercises, lab exams, take-home assignments and projects, and traditional sit-down quizzes and exams.

**Lab equipment:** Sufficiently many PCs with internet connectivity, and appropriate software (browsers, IDEs such as VSCode, and Python + IPYNB runtimes with libraries) will be needed.

**Logistics:** The course will be offered in the first semester of every year. However, due to the lack of teaching resources, departments opting for this course would be required to supply the following:

a. **Liaisons (1 liaison per department):** The liaison can be the DUGC convenor or their nominee. The liaison will ensure the availability of tutors and assistants. The liaison will be the single point of contact between the course instructor and the department.

b. **Tutors (1 tutor per 50 students):** Tutors may be faculty members or PG students or 3rd/4th year UG students. It is proposed that this modular course be granted permission to take UG students as tutors. Tutors need not have credited this course and need only have sufficient prior experience in Python programming. Tutors will conduct a weekly tutorial and help organize (e.g., help set questions) and conduct weekly labs, quizzes, and examinations.

c. **Assistants (1 assistant per 25 students):** Assistants may be PG students. They will invigilate, grade weekly labs, quizzes, and exams.

**Weekly breakup of content:** Numbers in square brackets [] against each topic indicate the number of lectures/labs for that topic.

*Lecture content (21 lectures):*

1. The Bare Necessities [9 lectures]
  - a. *Elements*: indentation, keywords, identifiers, distinction between name-object system (Python) vs variable-value system (C, C++, Java), console I/O
  - b. *Operators, expressions, statements*: unary, binary, assignment, conditional, precedence
  - c. *Loops*: while, for loops, notion of invariants, nested loops, break, continue
  - d. *Functions*: notion of argument and return, namespace, global vs local scope, recursion
2. Built-in Goodness [6 lectures]
  - a. *Built-in datatypes*: logical, integer, float, character, string, tuples, lists, dictionaries, sets, mutable vs immutable types
  - b. *Built-in operations*: length, string concatenation, string repeat, tuple unpacking, zipping, list comprehension, indexing, slicing, broadcasting, typecasting
3. Introduction to notebooks and library usage [3 lectures]
  - a. Imports, aliasing
  - b. *Basic operations*: using numpy, scipy to sort, perform numerical computations
4. Code hygiene [1 lecture]: Comments and documentation, writing requirements
5. Resource usage [2 lectures]
  - a. Reading API docs and utilizing community fora (Stack Overflow, GitHub)
  - b. Notebooks, IDEs and development environment (VSCode, Jupyter, Colab, Binder)
  - c. AI assistants (ChatGPT, Copilot)

*Lab content (7 labs):*

1. Familiarization with desktop computers, IDEs and programming environments
2. Basic python programming
3. Use of conditionals and loop invariants
4. Use of built-in and user-defined functions
5. Creative use of built-in datatypes and operations
6. Use of numpy and scipy libraries for numerical computations
7. Use of AI assistants and debugging

**Short summary for inclusion in the Courses of Study booklet:** the course will introduce basics of Python programming.

**Textbook:** There is no textbook for this course

**Course proposer:**

Date:

**Convener DUGC:**

Date:

The course is approved/not approved

**Chairperson, SUGC**

Date: