



PYTHON FULLSTACK TRAINING SYLLABUS



+91 7092909192
+91 7200117161

www.hitechinstitution.com



HI-TECH INSTITUTION

Hi-tech Institution is a trusted IT training institution established in 2017, dedicated to transforming careers through industry-relevant, real-time project-based learning. With a strong presence in Pondicherry, Thiruchitrambalam, and Chennai, we specialize in creating future-ready professionals.



OUR VISION

Our vision is to bridge the gap between traditional learning and industry needs. Our mission is simple: to empower individuals with hands-on skills, globally recognized certifications, and the confidence to excel in the IT world.

CORPARATE TRUST US

We've successfully partnered with IBM and other forward-thinking organizations to deliver impactful team training. With over 200 batches completed, our track record is built on results, retention, and referrals.

EXPERT FACULTY

At Hi-Tech Institution, training isn't delivered by traditional lecturers — it's led by real-time industry experts. Our Python Full Stack faculty are professionals with 10+ years of experience in front-end, back-end, and database development, specializing in Python, Django, HTML, CSS, JavaScript, React, and SQL. They bring real project expertise in building, integrating, and deploying full-scale web applications used in the industry today.

WHAT LEARNERS SAY

Our students not only succeed—they return. Many alumni re-enroll for additional courses and refer friends, a testament to our quality, trainer expertise, and hands-on approach.



WHO WE SERVE

- ◆ **Fresh Graduates** – Gain practical, job-ready skills and enter the IT workforce with confidence.
- ◆ **Career Switchers** – Move from non-IT to IT through structured, mentor-led training.
- ◆ **Working Professionals** – Upgrade your skillset, achieve promotions, and switch to high-demand roles.
- ◆ **Corporate Teams** – Custom programs designed to upskill entire teams for upcoming projects.
- ◆ **Final-Year Students** – Strengthen your resume with internships, projects, and certifications before graduation.
- ◆ **Non-Technical Backgrounds** – Step-by-step guidance to transition into IT even without coding experience.
- ◆ **Job Seekers** – Resume, interview prep, and portfolio building for faster placements.

JOB READY SKILL

Learn in months what usually takes years! With real-time projects and hands-on training in Python, Django, React, HTML, CSS, and SQL, you'll master the complete full-stack development process — from front-end design to back-end logic and database management — becoming job-ready and confident to build real-world web applications from Day 1.



400+
BATCH TRAININGS
SUCCESSFULLY COMPLETED

9000+
PROFESSIONALS
SUCCESSFULLY TRAINED



01 INTERACTIVE LIVE TRAINING

100% live sessions with real-time Q&A and doubt-solving. Engaging, interactive, and never boring like recorded videos.

02 DAILY HANDS-ON LABS

Every topic comes with guided labs and tasks so you learn by doing, not just watching.

03 ONE-TO-ONE DOUBT SOLVING

Struggling with a concept? Get personal 1:1 mentoring to clear doubts quickly.

04 MENTOR SUPPORT BEYOND CLASS

Get lifetime trainer guidance and alumni group access for continuous support.

05 FLEXIBLE LEARNING

Morning, evening, and weekend batches to fit your schedule.



DOMAIN EXPERTISE

A Journey of Innovation and Success



CAREER GROWTH IN THIS DOMAIN

A career in Python Full Stack Development offers high salaries and global opportunities — with Full Stack Developers earning ₹6–18 LPA and Software Engineers ₹10–25 LPA+, driven by the growing demand for end-to-end web application development, automation, and scalable digital solutions across industries worldwide.

OUR SPECIALIZATION

Unlike generic IT courses, we focus on Python Full Stack mastery. Our training dives deep into front-end (HTML, CSS, JavaScript, React), back-end (Python, Django, REST APIs), and database integration (MySQL, SQL). Every session connects theory with hands-on practice — so you don't just learn coding, you build and deploy complete real-world web applications just like professional full-stack developers do.

INDUSTRY-RELEVANT CURRICULUM

- Designed with corporate inputs
- Updated every quarter
- Covers end-to-end Fullstack development lifecycle

CERTIFICATION SUPPORT

We provide complete support to help you crack top certifications like PCEP & PCAP (Python Institute), Microsoft Python Certification, and Full Stack Developer Certifications — with hands-on projects, mock exams, expert guidance, and personalized mentorship to help you achieve guaranteed success in your Python Full Stack career.

INTERNSHIP

Learning without practice is incomplete. That's why we provide internships with real Python Full Stack projects covering front-end design, back-end logic, API development, and database integration. Every student builds complete, deployable web applications and earns an internship certificate that adds strong value to their resume and interview performance.

PORTFOLIO BUILDING

Learning without practice is incomplete. That's why we offer internships with real Python Full Stack applications and hands-on development challenges. Students build practical solutions across front-end, back-end, and database integration, gaining real project experience and earning certificates that strengthen resumes and boost interview success.

PRACTICAL-FIRST APPROACH

From Day 1, you'll learn by doing through labs, tasks, and real-world projects — gaining hands-on confidence to apply skills instantly.



Table of Contents

HTML Syllabus

◆ Getting Started with HTML

- Introduction to HTML and its role in web development
- Creating your first web page using `<html>`, `<head>`, `<body>`
- Adding line breaks `
` and spacing for readability
- Using comments `<!-- -->` for documentation

◆ Text Formatting & Lists

- Ordered lists `` and unordered lists ``
- Nested lists for hierarchical content
- Emphasizing text with ``, `<i>`, ``, ``

◆ Tables & Links

- Creating tables with `<table>`, `<tr>`, `<td>`
- Adding headers `<th>`, captions, and table borders
- Creating external links ``
- Creating internal page links (anchors) within the same document

◆ Images & Multimedia

- Adding images with `` tag
- Resizing and positioning images using attributes
- Title & alt attributes for accessibility
- Adding audio `<audio>` and video `<video>` elements
- Controlling media with attributes: `autoplay`, `loop`, `controls`

◆ Forms & User Input

- One-line text box `<input type="text">` with labels
- Multi-line text area `<textarea>`
- Radio buttons & checkboxes
- Number input box and date picker
- Dropdown menus `<select><option>`
- Fieldset & legend elements for grouping form items

◆ **Advanced HTML Elements**

- Meta element for SEO & responsiveness
- Escape characters (<, >, ©)
- Superscript <sup> and subscript <sub>
- Inline frame <iframe> for embedding maps & videos
- Doctype declaration and semantic HTML5 structure (<header>, <section>, <footer>)

◆ **HTML5 Graphics**

- Introduction to <canvas> element
- Drawing shapes: rectangle, lines, circles
- Applying colors & gradients
- Adding interactivity with simple JavaScript

CSS Syllabus

◆ **Introduction to CSS**

- What is CSS & how it enhances HTML
- Syntax: selectors, properties, values
- Inline, internal, and external CSS

◆ **Fonts & Text Styling**

- Changing font type, size, and color
- Using Google Fonts & web-safe fonts
- Font-weight, line-height, and letter-spacing
- Text-decoration, italics, and text-align

◆ **Colors, Backgrounds & Borders**

- Web colors (RGB, HEX, HSL)
- Background images and repeat properties
- Border styles: solid, dashed, dotted, radius
- Box model: margin, padding, border, content



Classes, IDs & Selectors

- Difference between classes (.class) & IDs (#id)
- Multiple selectors for styling efficiently
- Nesting styles & overriding rules
- Using div and span for layout



Layouts & Positioning

- Block vs inline elements
- Float property for side-by-side layouts
- Relative, absolute & fixed positioning
- Overflow property for handling large content
- Flexbox basics for modern responsive layouts



Menus & Navigation

- Vertical menu using lists
- Horizontal menu with inline-block
- Styling navigation bars with hover effects
- Dropdown menus using CSS only



Advanced CSS Effects

- Opacity & transparency
 - Text-shadow & box-shadow for depth
 - Custom cursor designs
 - CSS transitions & animations
 - Responsive design concepts with media queries
- 

React JS

React Basics

- What is React & why use it
- ES6 refresh: classes, arrow functions, destructuring
- Modules & imports
- Ternary operator for conditional rendering

JSX & Components

- React JSX syntax & rules
- Rendering HTML inside React
- Functional vs Class components
- Props for passing data between components

Events & Lists

- Handling events like clicks & input changes
- Conditional rendering with if, ternary & short-circuit
- Rendering lists with `.map()`
- Assigning unique keys in lists

Forms & Routing

- Controlled components in forms
- Handling form submissions
- React Router for navigation
- Route parameters & navigation links

React Styling

- Inline CSS styling
- Using CSS stylesheets in React
- React-specific `className` usage
- Sass basics for React projects

Python Syllabus

◆ Introduction to Python

- Overview of Python and its features
- Installing Python and setting up IDE (PyCharm, VS Code, Jupyter)
- Writing and executing first Python program
- Python syntax, indentation, and comments
- Role of Python in Web Development, Data Science, Automation, and Cloud

◆ Python Tokens

- Keywords, Identifiers, and Literals
- Constants and Variables
- Delimiters in Python
- Writing clean and meaningful identifiers
- Case sensitivity and naming conventions

◆ Data Types

- Built-in data types: Numbers, Strings, Boolean
- Collections: List, Tuple, Set, Dictionary
- Mutable vs Immutable types
- Type conversion (casting)
- Practical examples: storing student details, login systems, etc.

◆ Operators

- Arithmetic, Assignment, Relational, Logical operators
- Identity and Membership operators
- Operator precedence and associativity
- Practical usage: expressions in real-time scenarios

◆ Flow Control Statements

- Conditional statements (if, if-else, nested if)
- Looping statements (for, while)
- Range function and iterators
- Break, Continue, and Pass statements
- Real-time programs: calculators, menu-driven apps

◆ Functions

- Defining and calling functions
- Function arguments: positional, keyword, default, variable-length
- Return values
- Lambda functions
- Scope of variables (local, global, nonlocal)
- Practical examples: utility functions in projects

◆ Introduction to OOPs

- Need for Object-Oriented Programming
- Classes and Objects in Python
- Constructors (`__init__` method)
- Instance and Class variables
- Encapsulation and Abstraction concepts

◆ Inheritance

- Types of Inheritance: Single, Multiple, Multilevel, Hierarchical, Hybrid
- Method Overriding and `super()` function
- Polymorphism with functions and operators
- Real-life examples (Employee → Manager → Developer hierarchy)



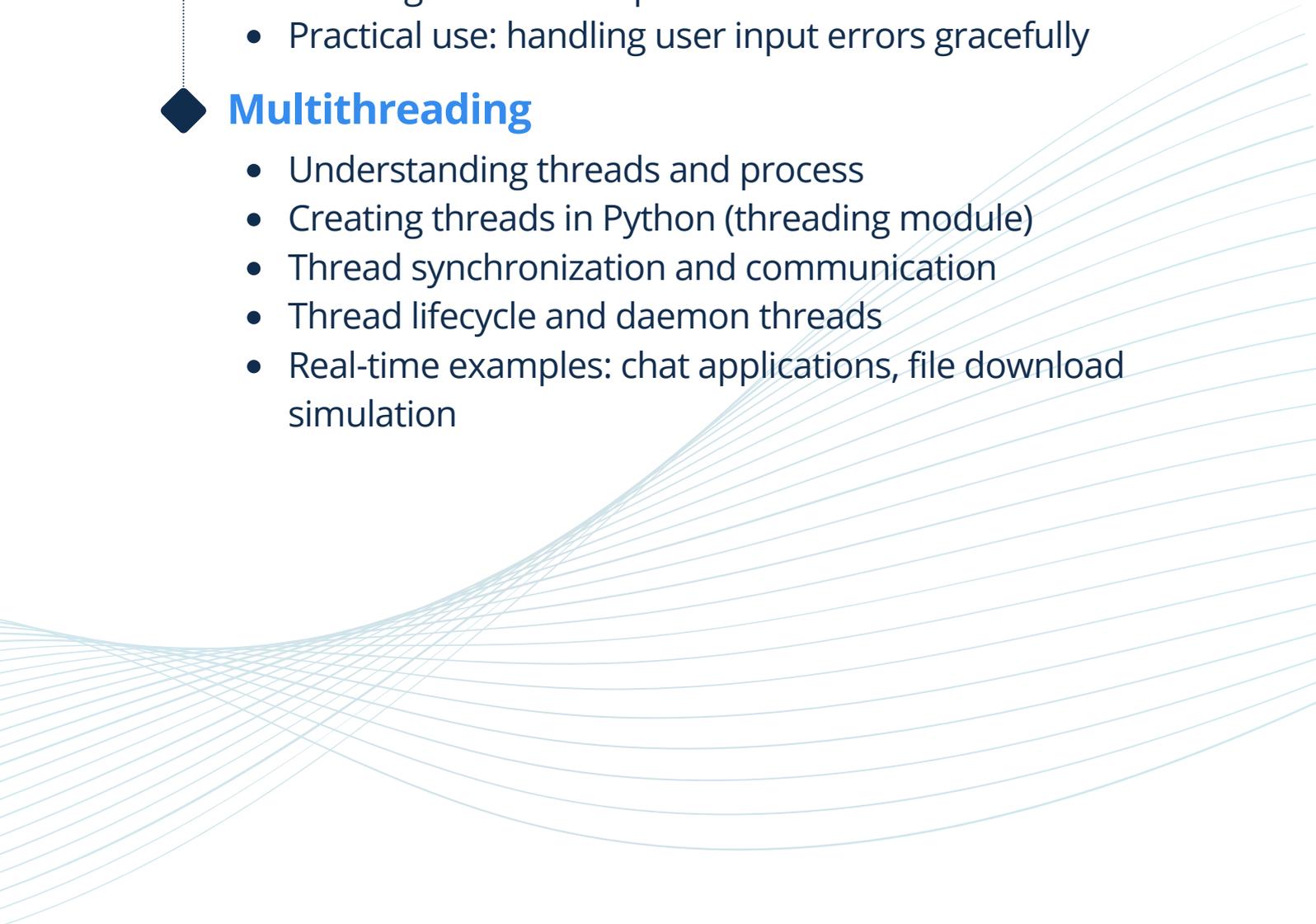
◆ File Handling

- Opening, Reading, and Writing files
- Different modes (r, w, a, rb, wb)
- Handling CSV and JSON files
- File pointer and file closing
- Real-time applications: log files, student data storage

◆ Exception Handling

- Need for exceptions
- Try, Except, Finally blocks
- Handling multiple exceptions
- Raising exceptions with raise keyword
- Creating custom exceptions
- Practical use: handling user input errors gracefully

◆ Multithreading

- Understanding threads and process
 - Creating threads in Python (threading module)
 - Thread synchronization and communication
 - Thread lifecycle and daemon threads
 - Real-time examples: chat applications, file download simulation
- 

Tkinter GUI in Python

◆ The Root Window

- Introduction to Tkinter library
- Creating the main/root window
- Window title, size, and geometry
- Resizing and positioning windows
- Adding icons and configuring background

◆ Fonts and Colors

- Using system fonts and custom fonts
- Setting text styles (bold, italic, underline)
- Applying colors to background and foreground
- Hexadecimal and RGB color codes in Tkinter
- Theme-based styling

◆ Canvas

- Introduction to Canvas widget
- Drawing shapes: line, rectangle, oval, polygon
- Adding images on Canvas
- Event handling on Canvas (mouse clicks, drags)
- Building simple drawing applications

◆ Frames

- Purpose of Frames in GUI
- Creating nested frames for layout design
- Organizing multiple widgets using frames
- Frame border styles and configurations

◆ Widgets Overview

- Importance of widgets in GUI
- Basic vs Advanced widgets
- Real-time usage of widgets in applications



◆ Arranging Widgets in Frame

- Geometry managers: Pack, Grid, Place
- Aligning widgets (left, right, top, bottom)
- Padding and spacing options
- Responsive layouts with grid system

◆ Button

- Creating and configuring buttons
- Adding text and images to buttons
- Command functions (event handling)
- Advanced button styling

◆ Label

- Displaying static and dynamic text
- Formatting text (font, size, color)
- Adding images to labels
- Updating label values in real-time

◆ Message

- Difference between Label and Message
- Creating multi-line text messages
- Configuring width, font, and justification

◆ Text

- Multi-line text input widget
 - Adding scrollbars to text areas
 - Real-time examples: Notepad app, Log viewer
 - Methods: insert, delete, get
- 



◆ Scrollbar

- Vertical and horizontal scrollbars
- Linking scrollbars with Text, Canvas, and Listbox
- Practical example: scrolling text editor

◆ Check Button

- Creating checkboxes
- Associating variables with checkbuttons
- Multi-option selection handling
- Example: Form with multiple preferences

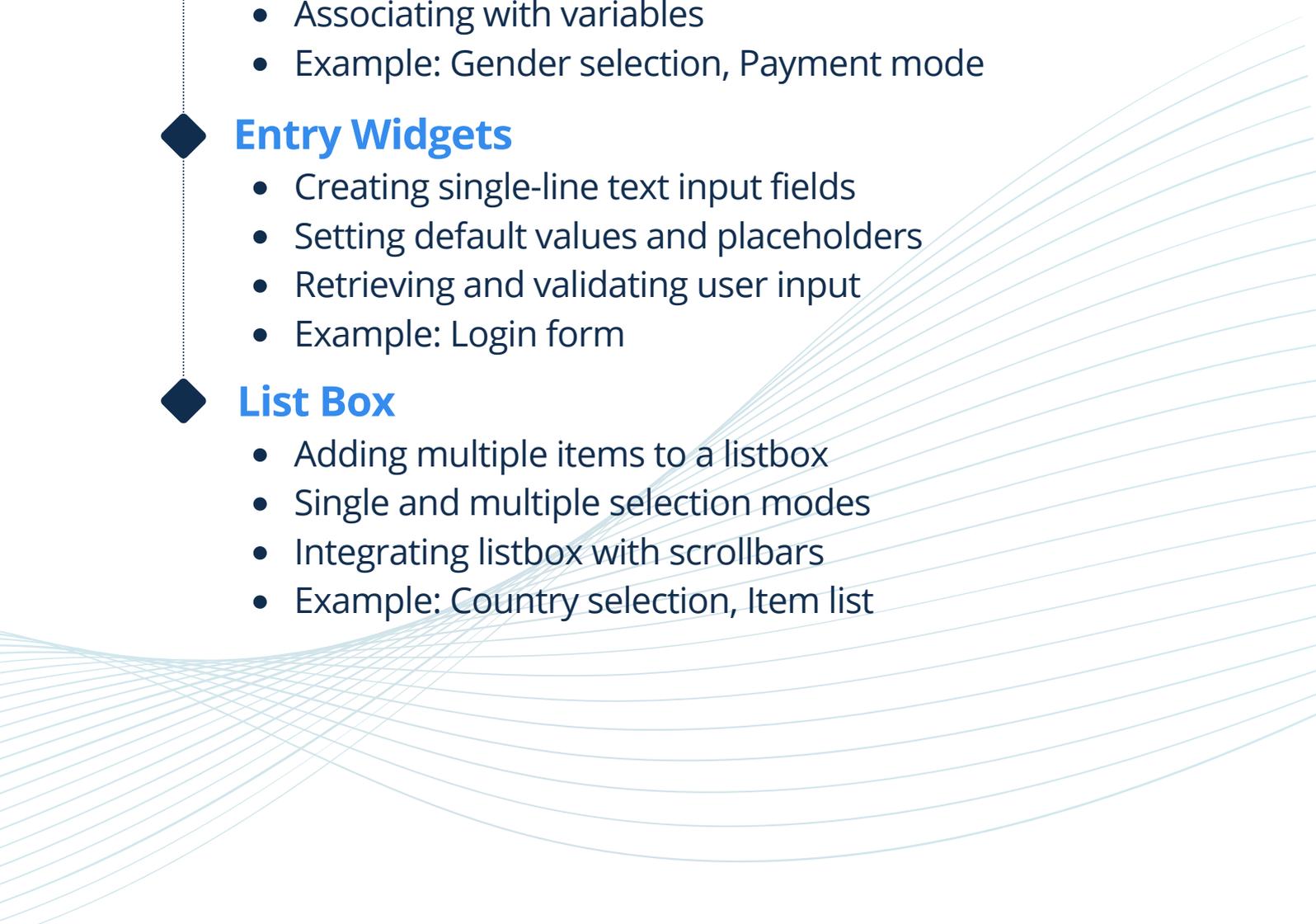
◆ Radio Button

- Creating grouped radio buttons
- Handling single-option selection
- Associating with variables
- Example: Gender selection, Payment mode

◆ Entry Widgets

- Creating single-line text input fields
- Setting default values and placeholders
- Retrieving and validating user input
- Example: Login form

◆ List Box

- Adding multiple items to a listbox
 - Single and multiple selection modes
 - Integrating listbox with scrollbars
 - Example: Country selection, Item list
- 



Menu Widgets

- Creating menubars and dropdown menus
- Adding submenus and separators
- Context (right-click) menus
- Example: Menu-driven text editor



Creating Tables

- Using Treeview widget for tables (from ttk)
- Adding rows, columns, and headers
- Styling tables (width, font, colors)
- Example: Student records table

Django Syllabus



Introduction to Django

- What is Django and why use it?
- MVC vs MVT architecture
- Django installation and setup
- Creating your first Django project
- Understanding project structure (manage.py, settings.py, urls.py, wsgi.py)
- Running the development server
- Advantages of Django in real-world projects



Django Views

- Understanding Views in Django
- Function-based views (FBV)
- Class-based views (CBV)
- Returning HTML responses
- Returning JSON and HTTP responses
- Rendering templates in views
- Example: Creating a “Hello World” view



URL Mapping

- Introduction to URL dispatcher
- Mapping URLs to views (urls.py)
- URL patterns and path converters
- Including multiple URL files (project-level vs app-level)
- Handling dynamic URLs with parameters
- Example: User profile page with dynamic ID



Django Templates

- Understanding the template system
- Template inheritance (base templates and child templates)
- Template tags and filters
- Variables and expressions in templates
- Using static files (CSS, JS, Images)
- Example: Creating a reusable navigation bar

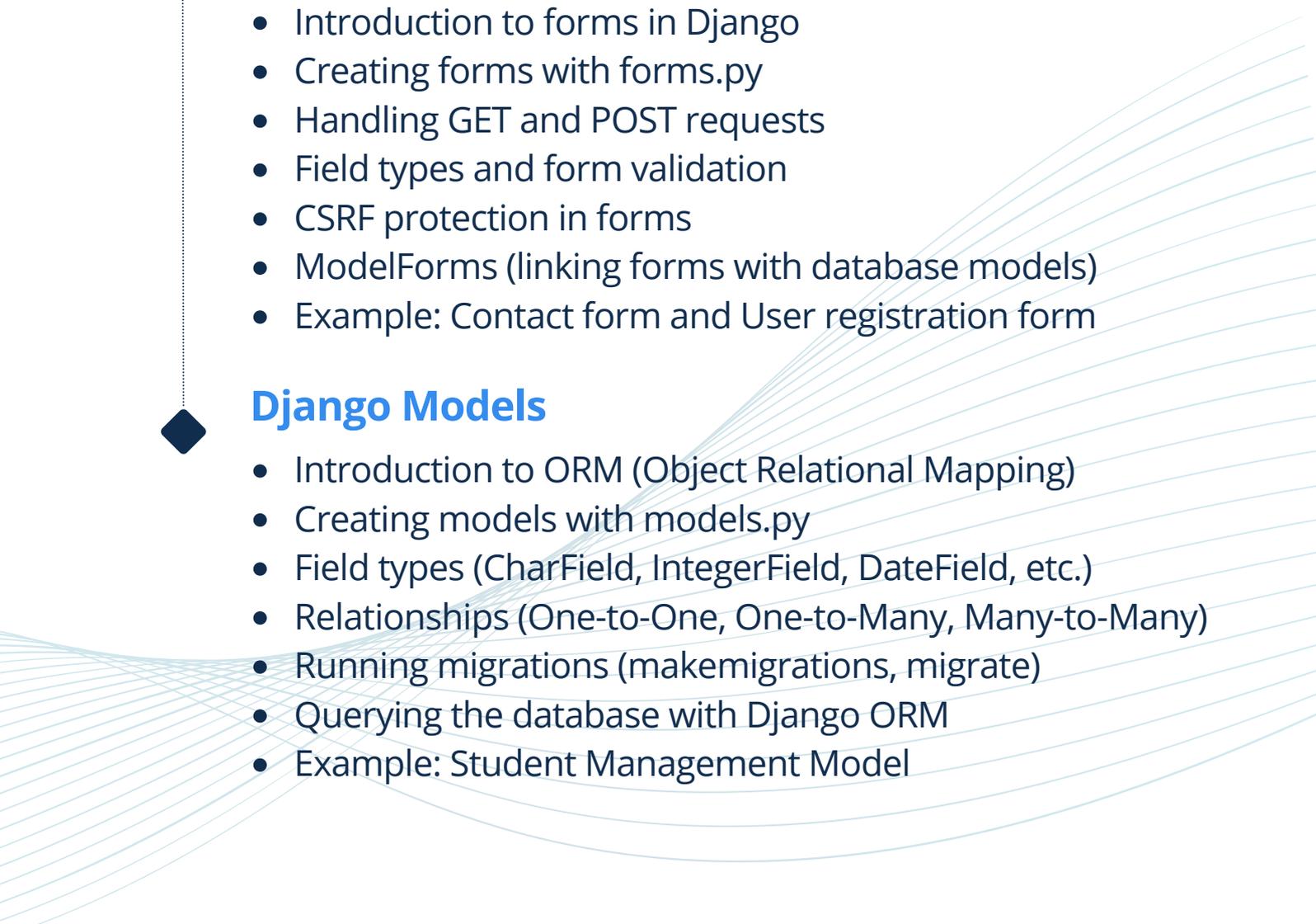


Django Forms

- Introduction to forms in Django
- Creating forms with forms.py
- Handling GET and POST requests
- Field types and form validation
- CSRF protection in forms
- ModelForms (linking forms with database models)
- Example: Contact form and User registration form



Django Models

- Introduction to ORM (Object Relational Mapping)
 - Creating models with models.py
 - Field types (CharField, IntegerField, DateField, etc.)
 - Relationships (One-to-One, One-to-Many, Many-to-Many)
 - Running migrations (makemigrations, migrate)
 - Querying the database with Django ORM
 - Example: Student Management Model
- 

◆ Session and Cookies

- Introduction to sessions and cookies in Django
- Storing user data across requests
- Setting and getting cookies in Django
- Managing session data (login/logout handling)
- Example: Implementing a shopping cart using sessions

◆ Django Admin Interface

- Introduction to Django Admin Panel
- Creating a superuser
- Registering models in the admin interface
- Customizing the admin dashboard
- Adding search, filters, and list displays
- Real-world use case: Managing blog posts via Admin

SQL Syllabus

◆ Introduction to SQL

- Overview of Databases and SQL
- Importance and Applications of SQL
- Understanding RDBMS (Relational Database Management System)
- Introduction to SQL Commands (DDL, DML, DCL, TCL)
- Connecting to SQL Server / MySQL / Oracle Database
- Creating and Managing Databases

◆ SQL Syntax

- SQL Language Rules and Conventions
- Structure of SQL Queries
- Using Clauses (SELECT, FROM, WHERE, ORDER BY, GROUP BY, HAVING)
- Writing Readable and Optimized SQL Statements
- Understanding Case Sensitivity and Naming Conventions

SQL Operations

- Data Definition Language (DDL): CREATE, ALTER, DROP, TRUNCATE
- Data Manipulation Language (DML): INSERT, UPDATE, DELETE
- Data Control Language (DCL): GRANT, REVOKE
- Transaction Control Language (TCL): COMMIT, ROLLBACK, SAVEPOINT
- Practical Examples with Each Command

Search Operations

- Using the SELECT Statement
- Filtering Records with WHERE
- Pattern Matching using LIKE and Wildcards
- Sorting Results with ORDER BY
- Using DISTINCT, BETWEEN, IN, and IS NULL
- Aggregate Functions: COUNT, SUM, AVG, MAX, MIN
- Grouping Data using GROUP BY and HAVING

Joins and Their Types

- Introduction to Joins and Relationships
- INNER JOIN – Combining Matching Records
- LEFT JOIN / RIGHT JOIN – Handling Missing Data
- FULL OUTER JOIN – Retrieving All Data
- SELF JOIN – Joining a Table to Itself
- CROSS JOIN – Cartesian Product of Tables
- Real-Time Examples of Multi-Table Queries

Comments in SQL

- Importance of Commenting for Readability
- Writing Single-Line and Multi-Line Comments
- Best Practices for Documenting SQL Queries and Scripts

Database Operations

- Creating and Managing Tables
- Adding, Modifying, and Deleting Columns
- Renaming and Dropping Tables
- Creating and Managing Schemas
- Viewing Metadata (DESCRIBE, SHOW TABLES, INFORMATION_SCHEMA)

Setting Conditions

- Logical Operators: AND, OR, NOT
- Conditional Expressions: CASE, IF, NULLIF, COALESCE
- Conditional Filtering in Queries
- Using Subqueries for Conditional Selection

Keys

- Understanding Primary Key and Foreign Key
- Unique, Composite, and Candidate Keys
- Constraints: NOT NULL, DEFAULT, CHECK, AUTO_INCREMENT
- Establishing Relationships between Tables (One-to-One, One-to-Many, Many-to-Many)

Set Operations

- UNION and UNION ALL
- INTERSECT
- EXCEPT / MINUS
- Combining Results from Multiple Tables
- Practical Scenarios and Use Cases

Stored Procedures

- Introduction to Stored Procedures
- Benefits of Using Procedures
- Creating, Executing, and Modifying Procedures
- Input and Output Parameters
- Using Conditional Logic Inside Procedures
- Error Handling and Transactions within Procedures

WHAT OUR STUDENTS SAY

We believe results speak louder than promises — and our students' reviews are proof of the trust and transformation we deliver every batch.



Kirubagaran .D
5 reviews

★★★★★ 8 months ago

I joined this institute to take training for AWS & Devops. My all over experience is very good 🙌. The knowledge given by the sir is very helpful to me 🙌. Thank you.



Harish Kumar
4 reviews

★★★★★ 8 months ago

While I joining this institute, I have less knowledge in aws and devops , after the guidance and good teaching by the Mathesh sir I have gained more knowledge. Thank you for your guidance and Teaching.



harshavardhana
2 reviews

★★★★★ 11 months ago

I recently completed an AWS and DevOps course at Hi-Tech institution, and I couldn't be more satisfied with the experience. The course was incredibly well-structured, covering both fundamental and advanced concepts in a way that was easy to follow yet challenging enough to push my learning boundaries.



Sridhar sri
1 review

★★★★★ 5 months ago

"Well-organized AWS and DevOps training with experienced mentors. A great place to build practical skills!"

❤️ 3



Bhuvanesh.V
2 reviews

★★★★★ 5 months ago

Good learning environment! The AWS and DevOps sessions were clear, practical, and highly beneficial for beginners

❤️ 2



AKASH
2 reviews

★★★★★ 5 months ago

Excellent training on AWS and DevOps! The mentors were knowledgeable, and the sessions were very insightful. Highly recommended!

❤️ 3



Sanjeevi Kumar
1 review

★★★★★ 3 months ago

I recently completed the AWS DevOps course at HiTech and I'm extremely satisfied with the experience. The trainers were knowledgeable and provided excellent support throughout the course. The hands-on labs and real-world examples helped me gain practical skills and confidence in using AWS services.



Kamal Raj
1 review · 1 photo

★★★★★ 11 months ago

Beyond the classroom, the Hi-Tech Institution center offers excellent support for AWS & Devops The availability of additional resources, such extra practice sessions, is incredibly beneficial. The staff is always approachable and ready to assist with any queries or challenges.



Anuj Jena
3 reviews

★★★★★ 7 months ago

I have attend AWS batch (Batch Number - B184 & Timing - 7:00 AM - 9:00 AM (Weekend)) & it was my good luck to had Hari as my batch instructor. Hari has tremendous knowledge on AWS & DevOps. I learnt lot of new topics related to AWS & DevOps. I want to highlight some point here according to my experience.



Sathyaswarubini Kannan
4 reviews

★★★★★ 7 months ago

I am Sathyaswarubini from Batch -B184, honestly speaking that Hari is a perfect instructor for AWS DevOps session. I am completely impressed with his Youtube videos and coaching. Every session started on time with clear note of him. I spent a great period of time on his sessions and got some confident level to attend the interviews in future. Thank you hari.



Naveen Kumar
1 review

★★★★★ 5 months ago

Had a great experience learning AWS and DevOps at Hi-Tech Institution. The structured training and expert guidance made it worthwhile

❤️ 2



D.Chandru
1 review

★★★★★ 2 weeks ago **NEW**

I had the opportunity to learn AWS Cloud Computing at HTGE Technologies Pvt Ltd for one month, and it was an excellent experience. The training was well-structured, covering both fundamental concepts and practical applications of AWS services. The instructors were knowledgeable, supportive, and always ready to clarify doubts with real-world exampes. which helped me gain confidence in working with cloud tools. The learning environment was friendly, interactive, and career-focused.



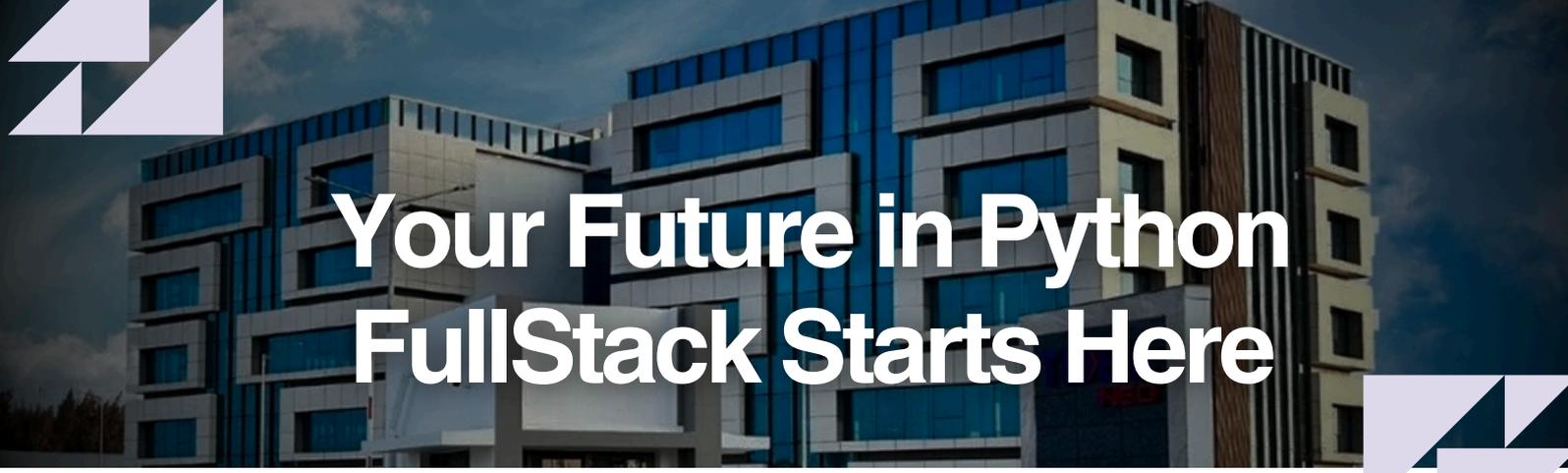
vindhiya e
1 review

★★★★★ 4 days ago **NEW**

I recently completed a one-month Cloud Computing internship at HTGE Technologies Pvt Ltd. It was a great learning experience where I got to work with real cloud platforms like AWS . The team was supportive, and the work environment made it easy to grow and ask questions. I learned a lot in a short time and truly appreciated the hands-on exposure. Highly recommend HTGE for anyone looking to build practical skills in tech.

Check out what our students are saying about us on Google :

<https://share.google/hm4sQdfCGAIA3CYHN>



Your Future in Python FullStack Starts Here

You've explored the course, the curriculum, the career scope, and the success stories. Now comes the most important step — taking action to build your career.

At HTGE Technologies, we are not just another training institute. We are a career-launchpad trusted by 200+ batches and corporate clients like IBM. Our mission is simple: to bridge the gap between learning and real-world application.

MEDIA MENTIONS

Youtube : <https://www.youtube.com/@hitechinstitution>

LinkedIn : <https://www.youtube.com/@hitechinstitution>

Instagram : <https://www.instagram.com/hitech.institution>

Website : www.hitechinstitution.com

COURSE DETAILS

Duration: 3 Months + Internship

Mode : In-Person & On-Line

Language : Tamil & English

Mode of Training: Live Online Sessions (with recordings)

Eligibility: Freshers, Working Professionals, Final-Year Students, Career Switchers

READY TO TAKE THE NEXT STEP

Visit: hitechinstitution.com

Call / WhatsApp: +91 7092909192 | +91 7200117161



**Congratulations on taking the first step
toward Python Fullstack mastery!**