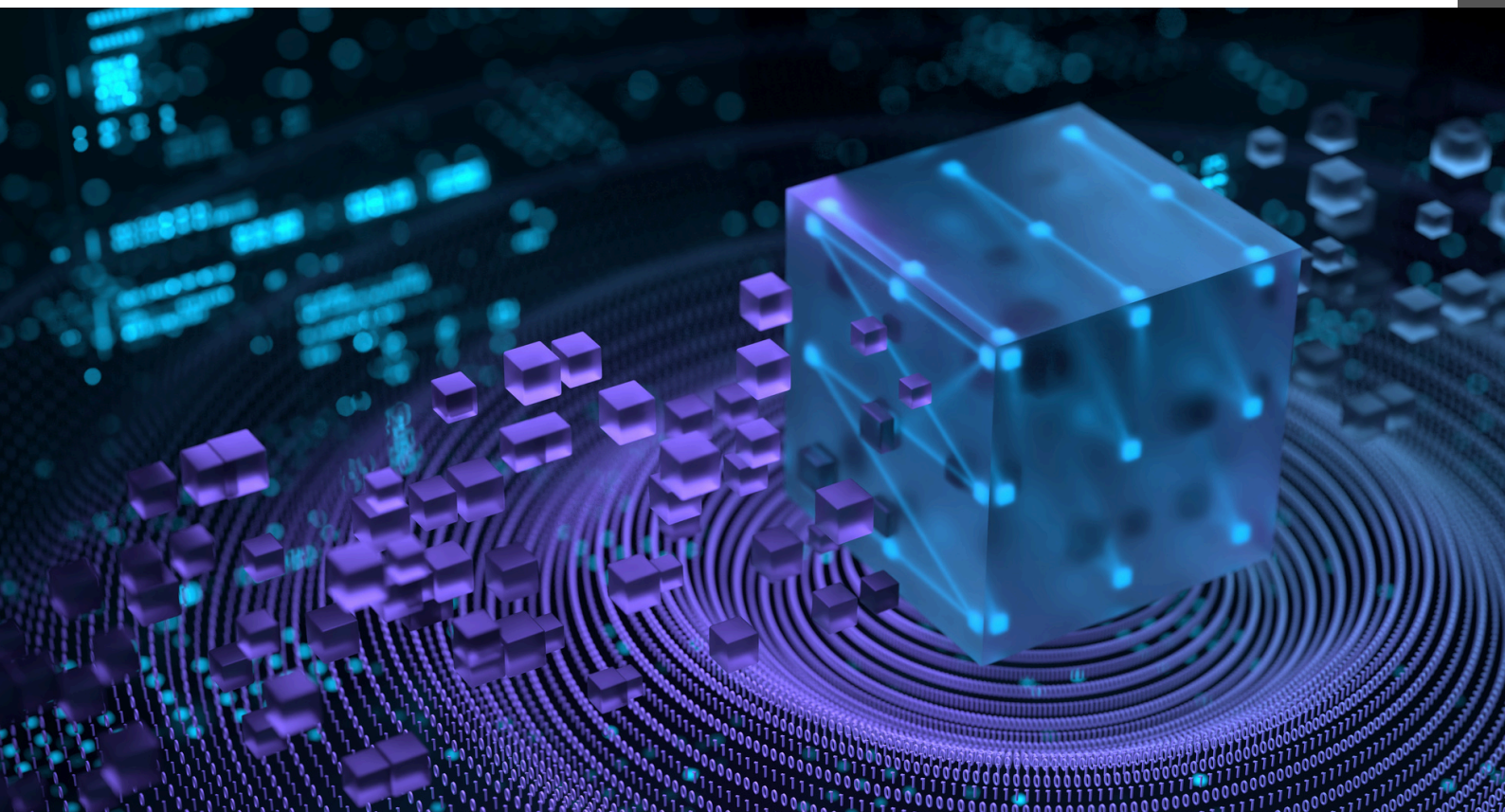**Theta Academy**

WEB DEVELOPMENT
USING
PYTHON-DJANGO

About Theta Academy

Theta Academy is committed in delivering specialized data science training tailored to the diverse interests and backgrounds of the students. Whether you're passionate about Data Science Mechanical Engineering, Electrical Engineering, Commerce, Biology, Management, Bio-Informatics, Computer Science. Theta's tailored courses are designed to empower you with the data-driven skills and knowledge you need to excel in your chosen domain.

List of the Courses:
- Artificial Intelligence for Mechanical and Electrical Engineering
- Data Science for Commerce and Management
- Data Analytics for MBA
- Data Science and Bio-Informatics
- Data Science for Computer Science and Applications
- AI for 11 and 12th Grade Students
- Python- Django Framework
- Front End Development

Why **Theta Academy** ?

🚀 Provide data science training to students of different fields with unique content and structure for each field like Mechanical, Electrical, Commerce, Biology, Management etc.

🚀 Experience real-world hiring scenarios with our online coding and mock interview platform,while AI-driven analytics of student's weekly and daily performance.

🚀 Dedicated placement assistance for your seamless transition into the professional world.

🚀 Late Evening and early morning batches for college students and working professionals

🚀 No prior knowledge of coding or applied mathematics needed before starting any course..

THETA ACADEMY

Features of Theta Academy

- Our courses blend the power of AI with real-world applications in your specific field, making learning engaging, relevant, and impactful

- Technical assessment,Interview and competition Platform is provided as Learning Management System with each course

- Provide Placements from Startups to Big level Firms with high salary packages

Learning Path of Django

01 HTML/CSS/Javascript

02 Python

03 Django Basics- MVT

04 Advance Concepts and Deployment

04 MySql

Tools We Covered :

python™

NumPy

MySQL

ChatGPT

DALL·E 2

✔

django

01 Python

Python is a high-level, interpreted programming language known for its simplicity and readability, making it ideal for beginners and experienced developers alike. Its syntax emphasizes code readability and allows programmers to express concepts in fewer lines of code compared to other languages like C++ or Java. Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python's community-driven development ensures continuous improvements and extensive support.

Topics Covered:

- ✔ **Introduction to Python programming**
- ✔ **Basic Data types**
- ✔ **List/tuples/dictionary/set**
- ✔ **If-else and loops**
- ✔ **Exception Handling**
- ✔ **File Handling**
- ✔ **Object Oriented Programming**
- ✔ **Dacorators and Generators**
- ✔ **System Programming**
- ✔ **API fetching using python**

02 Front End Development

Front-end development of a website involves creating the user interface and user experience aspects using HTML, CSS, and JavaScript. It focuses on the visual layout, interactive elements, and overall design to ensure the site is responsive and user-friendly across different devices and browsers. Developers work to optimize performance and accessibility to improve loading times and inclusivity for all users. They also ensure compatibility across various platforms and devices. Testing and debugging are crucial to maintaining a seamless and error-free experience.

Topics Covered:

✔   **HTML (HyperText Markup Language)**

   -Basics of HTML elements and attributes

   -Semantic HTML

   -Forms and input handling

   -Accessibility considerations (ARIA)

Topics Covered:

✓ **CSS (Cascading Style Sheets)**

- **CSS syntax and selectors**
- **Box model (margin, border, padding, content)**
- **Flexbox and Grid layout**
- **Responsive design principles (media queries)**
- **CSS preprocessors (SASS, LESS)**
- **Animations and transitions**

✓ **JavaScript**

- **JavaScript syntax and operators**
- **DOM manipulation and events**
- **ES6+ features (let/const, arrow functions, promises)**
- **Asynchronous JavaScript (AJAX, Fetch API)**
- **JSON and data manipulation**
- **JavaScript libraries (jQuery)**

✓ **Version Control with Git**

- **Git basics (init, add, commit, push, pull)**
- **Branching and merging**
- **Using GitHub for collaboration**
- **Handling merge conflicts**
- **Commit message best practices**

Topics Covered:

✓ **Responsive Web Design**

- **Mobile-first design approach**

- **Viewport and media queries**

- **Fluid grids and flexible images**

- **Using frameworks like Bootstrap or Foundation**

- **Testing responsiveness on different devices**

✓ **Web Performance Optimization**

- **Minification and concatenation of files**

- **Lazy loading images and videos**

- **Reducing HTTP requests**

- **Browser caching techniques**

- **Using CDNs (Content Delivery Networks)**

03 **Django**

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. It follows the model-template-views (MTV) architectural pattern and includes an ORM for database interactions, making it easier to manage database schema and migrations. With built-in features for authentication, URL routing, and template rendering, Django simplifies the process of building complex web applications. It emphasizes reusability and "pluggability" of components, enabling developers to use various third-party plugins for extended functionality. Django's robust security features help protect applications from common web vulnerabilities.

Topics Covered:

1. Introduction to Django

- **Overview of Django framework**

- **Setting up a Django development environment**

- **Creating a Django project and app**

- **Django project structure**

2. Django Models

- **Defining models and fields**

- **Model inheritance**

- **QuerySets and managers**

- **Database migrations**

- **Relationships (one-to-one, one-to-many, many-to-many)**

Topics Covered:

. **Django Admin Interface**

- **Setting up the admin site**

- **Customizing the admin interface**

- **Registering and managing models in the admin**

4. Django Views

- **Function-based views (FBVs)**

- **Class-based views (CBVs)**

- **Handling form submissions**

- **URL routing and configuration**

- **Using context and template context processors**

5. Django Templates

- **Template language syntax**

- **Template inheritance and extending templates**

- **Template filters and tags**

- **Using static files (CSS, JavaScript, images)**

- **Custom template tags and filters**

6. Django Forms

- **Creating and handling forms**

Topics Covered:

Form validation

- Model forms

- Advanced form processing

- Handling file uploads

7. Django REST Framework (DRF)

- Introduction to DRF

- Serializers and deserialization

- Class-based views and viewsets

- Routers and URL routing in DRF

- Authentication and permissions

- Pagination and filtering

8. Django Authentication

- User authentication and authorization

- Custom user models

- Managing user sessions

- Password management and resetting

- Social authentication (using third-party services)

Topics Covered:

9. Advanced Django

- **Middleware in Django**

- **Signals and how to use them**

- **Caching strategies (using Redis, Memcached)**

- **Testing in Django (unit tests, integration tests)**

- **Deploying Django applications (Heroku, AWS, Docker)**
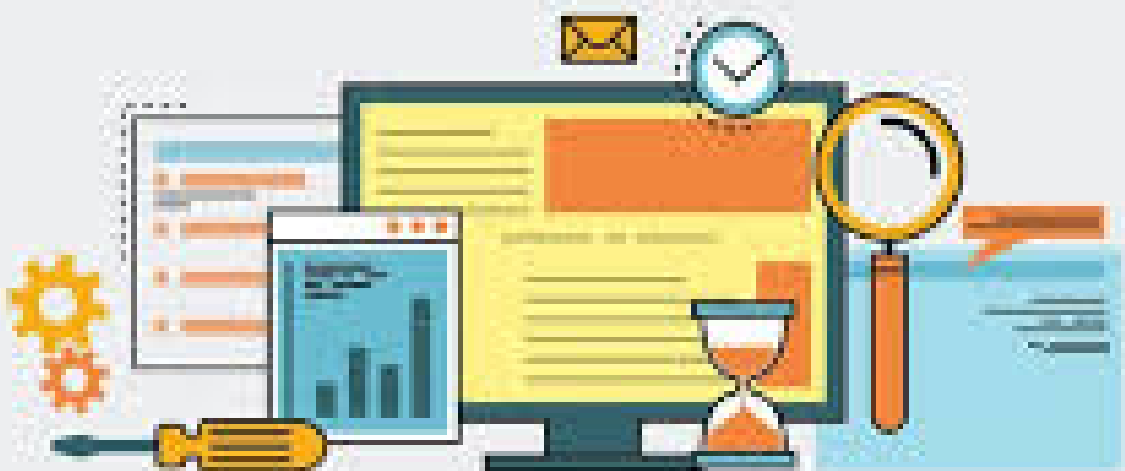
10. Django Security

- **Common security practices**

- **Preventing SQL injection and XSS**

- **Cross-Site Request Forgery (CSRF) protection**

- **Content Security Policy (CSP)**

- **Managing security updates and patches**

04 Advance Django Concepts

Topics Covered:

✔ **Django ORM Queries**

- **QuerySets and managers**

- **Filtering and chaining queries**

- **Aggregation and annotation**

- **Select_related and prefetch_related**

- **Raw SQL queries**

- **Optimizing ORM queries**

✔ **Blockchain Implementation in Django**

✔ **Payment Gateways implementation in Django**

05 Introduction to Databases and MySQL

Topics Covered:

✔ **Introduction to Databases and MySQL**

- **Overview of Databases**

- **Definition of databases and their role in storing and managing structured data.**

- **Comparison of databases with other data storage solutions.**

- **Introduction to MySQL**

- **Overview of MySQL as an open-source relational database management system (RDBMS).**

- **Historical context and popularity in web application development.**

✔ **Data Types and CRUD Operations in SQL**

- **Data Types in SQL**

- **Detailed exploration of SQL data types such as INT, VARCHAR, DATE, etc.**

- **Understanding the importance of choosing appropriate data types for columns.**

- **CRUD Operations**

- **In-depth coverage of CRUD (Create, Read, Update, Delete) operations in SQL.**

- **Practical examples and scenarios for applying each CRUD operation using select,update,delete,insert,create etc.**

- **AUTOINCREMENTING the columns**

Topics Covered:

☎ ✔ **Constraints and Modifying Tables Using ALTER**

- **Constraints in SQL**

- **Comprehensive study of constraints (PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK,DEFAULT,INDEX).**

- **Application of constraints to maintain data integrity..**

- **Modifying Tables Using ALTER**

- **Detailed examination of the ALTER TABLE statement.**

- **Hands-on exercises on adding, modifying, and dropping columns in existing tables.**

✔ **Operators in SQL**

- **Comparison Operators**

- **In-depth explanation and usage of comparison operators (=, !=, <, >).**

- **Practical examples illustrating how to filter data using comparison operators.**

- **Logical Operators**

- **Exploration of logical operators (AND, OR, NOT) for constructing complex conditions.**

- **Application of logical operators in querying databases.**

- **Arithmetic Operators**

- **Understanding arithmetic operators (+, -, *, /) for mathematical operations in SQL.**

- **Real-world examples showcasing the use of arithmetic operators.**

- **Advance Operators**

Topics Covered:

✔ **Nested Queries**

- Joins in SQL

- Thorough coverage of INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN.

- Practical exercises for combining data from multiple tables.

- Group By: - Comprehensive study of the GROUP BY clause

- Hands-on examples illustrating the use of nested queries.

- Set theory Functions: - Union,Union all,intersection, minus etc.

✔ **ER Diagrams and Types of Relationships**

- ER Diagrams

- Explanation of Entity-Relationship (ER) diagrams as a visual representation of database structures.

- Drawing and interpreting ER diagrams for various scenarios.

- Types of Relationships

- Detailed exploration of one-to-one, one-to-many, many-to-one, and many-to-many relationships.

- Real-world examples illustrating different types of relationships.

✔ **Stored Procedures and Functions**

- Stored Procedures

- Creation and execution of stored procedures for code reusability.

- In,out,and inout variables

- Functions: Comparison of stored procedures and functions.

- Writing and using functions to return values in SQL.

![THETA ACADEMY]

Topics Covered:

✔ **Nested Queries**

- Joins in SQL

- Thorough coverage of INNER JOIN, LEFT JOIN, RIGHT JOIN, and
  FULL JOIN.

- Practical exercises for combining data from multiple tables.

- Group By: - Comprehensive study of the GROUP BY clause

- Hands-on examples illustrating the use of nested queries.

- Set theory Functions: - Union,Union all,intersection, minus etc.

✔ **ER Diagrams and Types of Relationships**

- ER Diagrams

- Explanation of Entity-Relationship (ER) diagrams as a visual
  representation of database structures.

- Drawing and interpreting ER diagrams for various scenarios.

- Types of Relationships

- Detailed exploration of one-to-one, one-to-many, many-to-one,
  and many-to-many relationships.

- Real-world examples illustrating different types of relationships.

✔ **Stored Procedures and Functions**

- Stored Procedures

- Creation and execution of stored procedures for code reusability.

- In,out,and inout variables

- Functions: Comparison of stored procedures and functions.

- Writing and using functions to return values in SQL.

Capstone Projects

01 Crypto Stacking Platform where we can invest in any crypto currency in USDT

02 Learning Management System for proper learning of students with inbuilt compiler and performance tracker

03 Clone of Just-Dial Marketing tool to promote local brands of search area

Last Month Placements :

Lalita (Data Scientist)
Placed : HDFC
8 Lac/ Annum

Karan Rana(Django Developer)
Placed : Netsmartz
6 Lac/annum

Gurdeep (Data Analyst)
Placed : Bajaj Finance
6 Lac/ Annum

THETA ACADEMY

Companies Where Our Students Placed :

accenture

amazon.com

BOSCH

Walmart

CISCO

Myntra
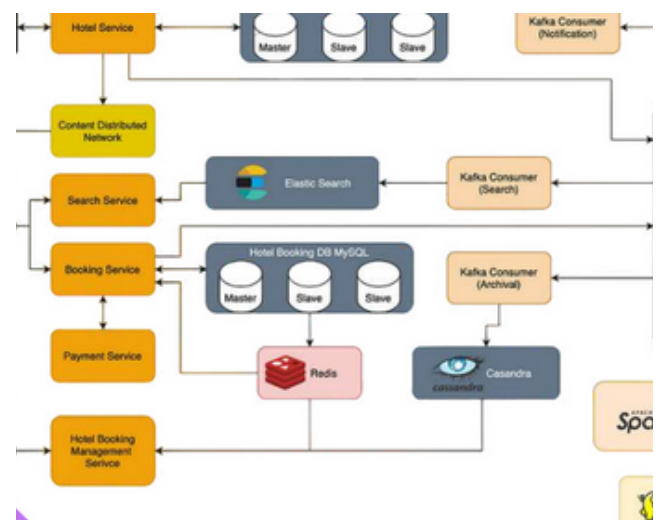
OLA

Our Industrial Projects

Education LMS

**Auto ACB
Trip Recorder**

**Financial Comparison
Portal**

**Recomendation System
for Hotel Management**

Contact us

📱 +91 9501981164 , +91 6239006443

🔗 www.thetacademy.com

✉️ info@thetacademy.com

📍 Cabin No. 413, SCO 107-108-109, 4th Floor, Sector 34-A, Chandigarh, 160022