Government of Pakistan

**National Vocational and Technical Training Commission**

Prime Minister Kamyab Jawan Program

"Skills for All"



**Course Contents/ Lesson Plan**

**Course Title: CIT(Web Development)**

**Duration:** 6 Months

|  |  |
| --- | --- |
| **Trainer Name** | Farhan Karim |
| **Course Title** | **Web Designing and Web Engineering** |
| **Objective of Course** | **The learning objectives of this course are:**   * To understand why Python is a useful scripting language for developers. * To learn how to design and program Python applications. * To learn how to use lists, tuples, and dictionaries in Python programs. * To learn how to identify Python object types. * To learn how to use indexing and slicing to access data in Python programs. * To define the structure and components of a Python program. * To learn how to write loops and decision statements in Python. * To learn how to write functions and pass arguments in Python. * To create complete web front-end design using html/css/javascript * Add animations and different visuals using bootstrap and jquery. * To learn how to build and package Python modules for reusability. * To learn how to read and write files in Python. * To learn how to design object‐oriented programs with Python classes. * To learn how to use class inheritance in Python for reusability. * To learn how to use exception handling in Python applications for error handling. * To learn how to create a fully-fledged web app using flask framework in python. |
| Learning Outcome of the Course | By the end of this course, the trainees should gain the following competencies:   * Understanding of programming techniques * Design and structure of android based application databases * Design and structure of web-based applications * Design and coding skills * Integration with API’S * Problems Solving Skills * Threading * Web Based android application * Understanding of Graphics * Design and structure * Texturing * Rigging * Editing * Drawing * Movements * Modeling |
| Course Execution Plan | Total Duration of Course: **6 Months (26 Weeks)** |
| Class Hours: **4 Hours per day** |
| **Theory: 20% Practical: 80%** |
| Weekly Hours: **20 Hours Per week** |
| Total Contact Hours: **520 Hours** |
| Companies Offering Jobs in the respective trade | 1. Upwork 2. Freelancing 3. Fiverr 4. Government Institutes 5. Software Houses 6. Crossover 7. All Private Institutes who are managing software |
| Job Opportunities | * Software Engineers * Web Developers * Full-Stack Python Developer |
| **No of Students** |  |
| **Learning Place** | Classroom / Lab |
| **Instructional Resources** | **Development Platform:**   * <https://github.com/> **,** * <https://python.org/> , * <https://visualstudio.microsoft.com/>, * <https://getbootstrap.com/><https://laravel.com/>, * <https://jquery.com/>   **Learning Material:**   * **Aptech International Official Books** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Scheduled Week** | **Module Title** | **Learning Units** | **Remarks** |
| **Week 1** | * Introduction | * **Motivational Lecture** * **Course Introduction** * **Success stories** * **Job market** * **Course Applications** * **Institute/work ethics** |  |
| **Week 2** | * Building Next Generation Website (HTML) | * Explain the evolution of HTML * Explain the page structure used by HTML * List the drawbacks in HTML 4 and XHTML * List the new features of HTML * Explain CSS and JavaScript * Explain jQuery * Explain browser support for HTML5 * Explain the elements constituting an HTML tag * Describe container and standalone tags * Explain the role of HTML5 in Mobile devices * Explain the Heading tag * Explain the different tags related to formatting * Explain monospaced font, preformatted text, and block quotation * Describe the different types of lists * Explain the procedure to change the background color and image * Describe hyperlinks * Explain how to hyperlink to a Web page and e-mail address * Explain how to hyperlink to anchors and other content * Identify the new functions of CSS3 * Explain the different types of selectors * Define Classes and IDs for applying styles * Explain the process to apply styles to hyperlink |  |
| **Week 3** | * Building Next Generation Website (HTML) | * List and explain text and font styles * Explain paragraph indentation and application of border * Explain vertical spacing within a paragraph * Explain graphic formatting in Web pages * Explain graphic insertion, sizing, and padding |  |
| **Week 4** | * Building Next Generation Website (HTML | * Explain CSS3 Animation * Describe the use of CSS3 on Mobile devices * Explain HTML 5 semantic tags * Explain HTML 5 semantic tag layouts * Explain the usage of navigation bar * Describe a text-based and graphical navigation bar * Explain image mapping * Explain divisions in HTML5 * Describe the creating and formatting tables * Explain the table size and the width of a column * Describe the table formatting and merging of table cells * Explain the page layout for tables * Describe HTML5 forms * Explain the working of new input types in HTML5 * Explain the new Form attributes and elements * Describe the need for multimedia in HTML5 * Explain the audio elements in HTML5 |  |
| **Week 5** | * Bootstrap | * Learn basics of Bootstrap and set up Bootstrap. * Work with Plug-ins. * Learn jQuery. * Understand Selectors, functions, widgets in jQuery. * Learn the Responsive Web layout in Bootstrap. * Learn Headers and classes in Bootstrap. * Build Webpages using Bootstrap and jQuery. |  |
| **Week 6** | * Javascript | * Define JavaScript * Differentiate between JavaScript and Java * Describe Web Applications and Websites * Compare and contrast between server-side and client-side scripting * Explain basic concepts of JavaScript * Explain Regular Expression (RegExp) in JavaScript and its uses * Identify RegExp object and matching patterns in JavaScript * Describe the uses of modifiers, brackets, and metacharacters * Describe properties and methods of RegExp in JavaScript |  |
| **Week 7** | * Introduction to Python | * What is Python? * Python History and Versions * Documentation and Resources * Python Implementations * Getting Python * Installing the Plugin * Different Ways to Run Python * IDE * The Python Shell (and IPython) * Double-Clicking to Execute Python * Running Python from IDE * Debugging – Stepping Through Code * Passing Command Line Arguments * Accessing Command Line Arguments   Hands-on Lab Exercises   * Identifiers * Naming Conventions |  |
| **Week 8** | * Core Python | * Keywords and Built-ins * The Garbage Collector * Operators * Strings * String Functions * Formatting Strings * Numbers * Array * Capturing Input and Handling Data Conversion * Booleans * Data Structures: Sequences, Sets, and Dictionaries * Functions * Lambda * Checking Data Types * Hands-on Lab Exercises |  |
| **Week 9** | * Working with Modules & Data Structures | * Sequences * Lists * List Operations * The range() and xrange() Functions * Tuples * Looping through Sequences * Slicing Sequences * Comparing Sequences * Dictionaries * Dictionary Operations * Sets * Complex Data Structures * Deep vs. Shallow Copy * Hands-on Lab Exercises * What is a Module? * Importing Modules |  |
| **Week 10** | * Working with Modules & Data Structures | * Understanding the PYTHONPATH * Packages * Compiled Python Code * Python Standard Modules * dir() and help() * Finding and Installing Modules * Installing PIP * Installing and Upgrading Modules with PIP * More PIP Operations * Hands-on Lab Exercises |  |
| **Week 11** | * Program Structure | * Statements * Comments * Joining Lines * Indentation * Operators * Operator Precedence * If Statements * Evaluating Variables * While Loops * For Loops * Tuple Assignment with For Loops * Hands-on Lab Exercises |  |
| **Week 12** | * Functions | * Introduction to Functions * Function Parameters and Default Arguments * Positional vs. Named Notation * Passing by Value vs. Reference * Unpacking Positional Arguments * Unpacking Named Arguments * Overloading Functions * Returning Data from Functions * Function Variable Scope * Documentation Strings in Functions * Hands-on Lab Exercises |  |
| **Week 13** | * Exception Handling | * Exception Handling with try…except * Else and Finally * Exception Class Inheritance Hierarchy * Handling Multiple Exceptions * Explicit Exception Raising * Re-raising Exceptions * Custom Exception Classes * Hands-on Lab Exercises |  |
| **Week 14** | * Built-in Functions and Modules * Working with Files Data Analysis | * Built-in Functions * The Python Standard Library * The date time Module * Time * Time Formats * The scheduled Module * Hands-on Lab Exercises * Reading from Files * Reading Lines from Files * Reading JSON from Files * Writing and Appending to Files * Using ‘with’ to Manage Resources * File Attributes * Hands-on Lab Exercises |  |
| **Week 15** | **Mid-Term Assignment/Exam** | | |
| **Week 16** | * Python Object-Oriented Programming | * Introduction to Object-Oriented Python * Creating Your First Class * Inheritance * Multiple Inheritance and Method Resolution Order * Accessing Attributes * Superclass Methods * Method Overloading * Class Attributes * Static and Class Methods * Hands-on Lab Exercises |  |
| **Week 17** | * Python Database Access | * Python Database API supports * Install MYSQLdb * Install Python module MySQLdb * Create database ,tables on MYSQLdb * Connect MYSQLdb * CRUD (Select, Insert, Update, Delete) Operation |  |
| **Week 18** | * Python Flask Web framework | * Basic python application framework * Use Flask to create basic landing pages * Use WTForms to accept user inputs from a Flask Application * Use Flask and SQLAlchemy as an ORM for a SQL database * Use blueprints to structure larger Flask Applications. |  |
| **Week 19** | * Python Flask Web framework | * Enable User Authentication and Authorization with Flask * Understand OAuth with Flask Applications * Create simple REST APIs with Flask |  |
| **Week 20** | * Python Flask Web framework | * GUI and web forms * Database Integration * E-mail support for web applications * User authentication support for application |  |
| **Week 21** | Employable Project/Assignment (6 weeks (i.e 21-26) in addition of regular classes. | * Guidelines to the Trainees for selection of students employable project like final year project (FYP) * Assign Independent project to each Trainee * A project based on trainee’s aptitude and acquired skills. * Designed by keeping in view the emerging trends in the local market as well as across the globe. * The project idea may be based on Entrepreneur. * Leading to the successful employment. * The duration of the project will be 6 weeks * Ideas may be generated via different sites such as:   <https://1000projects.org/>  <https://nevonprojects.com/>  [[https://www.freestudentprojects.com/](https://technofizi.net/best-computer-science-and-engineering-cse-project-topics-ideas-for-students/)](https://www.freestudentprojects.com/)  <https://technofizi.net/best-computer-science-and-engineering-cse-project-topics-ideas-for-students/>   * Final viva/assessment will be conducted on project assignments. * At the end of session the project will be presented in skills competition * The skill competition will be conducted on zonal, regional and National level. * The project will be presented in front of Industrialists for commercialization |  |
| **Week 22** |  | Working on project |  |
| **Week 23** |  | Working on project |  |
| **Week 24** |  | Working on project |  |
| **Week 25** |  | Working on project |  |
| **Week 26** |  | Working on project |  |

List of Machinery / Equipment

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Name of item as per curriculum** | **Quantity physically available at the training location** |
| 1 | Computers Minimum Corei5   * LCD Display 21” with built in speakers | 25 |
| 2 | Mobiles with Android OS | 25 |
| 3 | DSL Internet Connection (Minimum 1 MB) | Available on every PC |
| 4 | Accessories/Devices  * Connectors * Multimedia * Printer (NW printer) * Audio/visual aid * White Board * Pin Board * Flip Chart Board * Hard copy of Training Material * Mobile Phones | 25 each |
| 5 | Wires, data cables, power plugs, power supply | For every PC |
| 6 | UPS | Available |
| 7 | Generator / Solar Backup | Available |
| 8 | Air Conditioner (2 Tons) | Available |

1. Software List

|  |  |
| --- | --- |
| **Sr. No** | **Software Name** |
|  | Visual Studio Code (Installed on each PC) |
|  | Operating System (Windows 10) |
|  | Python 3.8.1 |
|  | Web Servers including IIS, Apache (Licensed software installed on each PC) |
|  | Databases including MySQL, ERWIN (Licensed software installed on each PC) |
|  | FTP Client including FileZilla, File Manager (Licensed software installed on each PC) |
|  | Web hosting manager/control panel |
|  | Web browser including Internet Explorer, Google Chrome, Mozilla Firefox, Netscape, Opera (installed on each PC) |
|  | Firewall (each PC) |
|  | Security scanning tools including Antivirus (each PC)  Networking |
|  |  |

1. Minimum Qualification of Teachers / Instructor

The qualification of teachers / instructor of this course should be minimum **of bachelors in Computer science with minimum 3 years of development experience** in relevant trade.

* Bachelors of Computers Science / Networks (Hons)

1. Supportive Notes

## Teaching Learning Material

|  |  |
| --- | --- |
| **Books Name** | **Author** |
| Programming with Python | Aptech International |
| Web Application Development using Python- | Aptech International |
| Building Next-Generation Website(HTML) | Aptech International |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |