Teeny Coders

PRODUCING TECH LEADERS FOR FUTURE

STEM.ORG ACCREDITED EDUCATIONAL EXPERIENCE

Teeny Coders

VISION
MISSION & VALUES

Our Vision

To establish a state of the art global online coding school for School kids to catch up with the tech industry quickly

Our Mission

To excel the coding, mathematical and problem solving skills in school kids to explore their hidden talent through advanced programming technologies

Our Values

We believe to inculcate the following core values in our future tech leaders

01

SELF EFFICACY

We generate self-belief in the kids to dig out their hidden abilities to perform any task with confidence to achieve their goals.

02

SEEKING FOR LEARNING

We value inquisitiveness and growth of kids with different learning needs. We encourage them to become creative, logical thinkers and problem solvers for themselves and the society.

03

LEADERSHIP

Our teeny coders are the leader of the digital future. We enlighten them with individual and teamwork abilities, coupled with moral and ethical values, to serve the community.

04

INCULCATION OF SKILLS

Every day, we are moving towards digitalization. We believe in inculcating coding, mathematical and problem solving skills in kids through our quality curriculum to meet the needs of the digital future.

STEM.ORG ACCREDITED EDUCATIONAL EXPERIENCE

Teeny Coders

WHY TEENY CODERS?

BLOOM'S TAXONOMY

We believe that every Teeny Coder is the leader of digital future. Our quality curriculum is designed based on these six levels (Create, Evaluate, Analyse, Apply, Understand and Remember) for effective learning. Teeny Coders have 0% compromise policy towards quality education, and adopt the standard guidelines.

FACE MODEL

Each teeny coder's learning matters. Therefore, we have developed our own FACE FUN-TO-LEARN, ADVANCED, CREATIVE AND EVOLVING model to verify that our curriculum is nourishing every teeny coder.

QUALITY CURRICULUM

Our Quality Curriculum Is one of our main Product. Our Fun-to-Learn, Advanced, Creative, and Evolving Curriculum is Based On Bloom's Taxonomy Standards, which makes Sure That Every Teeny Coder Is Obtaining the best Coding, Problem Solving And Cognitive Skills.

COMPETENT FACULTY

We have selected the best faculty for our Teeny Coders, who are graduates from renowned universities with great teaching experience at academia and industry levels. Our faculty is energetic, efficient and passionate to teach our digital future leaders.

VARIETY OF COURSES

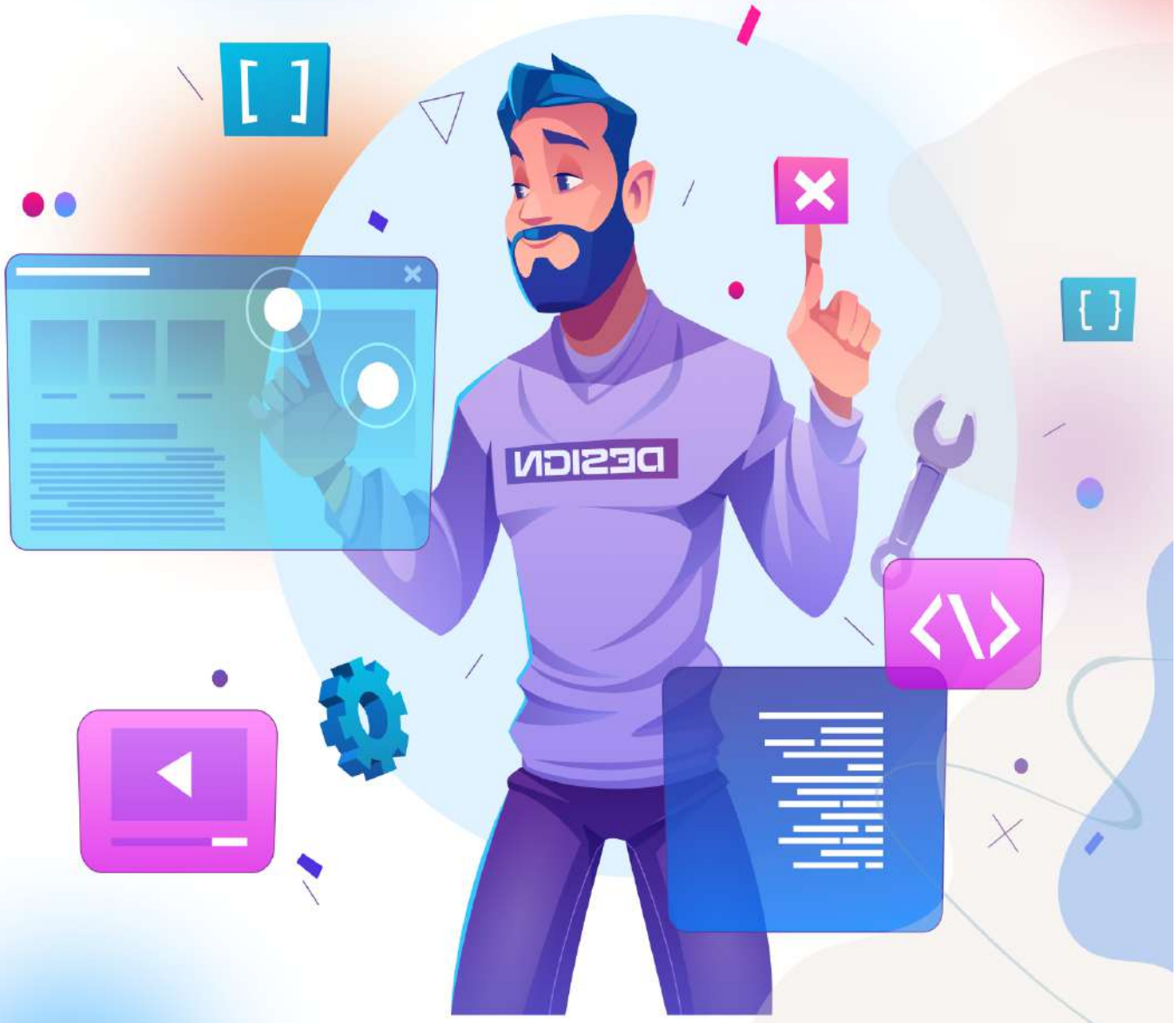
We, at TEENY CODERS, offer a variety of flavours (courses) which are specifically designed for grade 1 to grade 12 kids. Every course comprise of three difficulty levels (Beginner, Intermediate and Expert). We make sure that every TEENY CODER enjoy their code learning journey with solid concepts.

STEERING LEADERSHIP

Teeny Coders leadership have combined experience of more than 25 years in academia and industry. Therefore, every teeny coders future is bright and safe because our leadership knows what is best for your kids.

Teeny Coders

PYTHON PROGRAMMING CURRICULUM



Teeny Coders

BEGINNER LEVEL

Course Contents

20 Lectures • 24 Activities •Duration: 2-3 Months

STEM.ORG ACCREDITED EDUCATIONAL EXPERIENCE

LECTURE NO.	TOPICS : ACTIVITIES
Lecture 1	Introduction to Programming in Python & Use of Python : Geometric Shapes
Lecture 2	Variables : Generate random name
Lecture 3	Data Types and Numbers : Type Conversion
Lecture 4	String : Operations on String
Lecture 5	Basic Math Expressions : Euclidean Distance
Lecture 6	Operators : Types of Operators
Lecture 7	Conditional Structures & Boolean : Astrological Sign
Lecture 8	Loops (For Loop) : Random Password
Lecture 9	Loops (While Loop) : Number Guessing
Lecture 10	Data Structures (Dictionaries) : Quiz app
Lecture 11	Data Structures (List) : Fibonacci Number
Lecture 12	Tuple : Frequency Tuple
Lecture 13	Set : Same Frequencies
Lecture 14	Functions (Basic) : List to Nested Dict
Lecture 15	Functions (Intermediate) : Pangram String
Lecture 16	Iterators and itertools : Running Product
Lecture 17	Fun Coding Problems : Solving Coding Problems
Lecture 18	Fun Coding Problems : Solving Coding Problems
Lecture 19	Project : Fixing a Skeleton Code
Lecture 20	Project : Fixing a Skeleton Code