Teeny Coders

PRODUCING TECH LEADERS FOR FUTURE

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.

{ } 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.

BLOOM

FACE MODEL

FACE

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.

STEM.ORG ACCREDITED™ EDUCATIONAL EXPERIENCE

Teeny Coders

MACHINE LEARNING CURRICULUM

{ } Teeny Coders

INTERMEDIATE LEVEL

Course Contents

STEM.ORG ACCREDITED™ EDUCATIONAL EXPERIENCE

22 Lectures • 26 Activities •Duration: 3-4 Months

LECTURE NO.	TOPICS : ACTIVITIES
Lecture 1	What is audio dataset : Collecting different audios for dataset creation, how to label them.
Lecture 2	Audio conversion to numeric array : Audio files conversion in .csv file using python
Lecture 3	Audio feature extraction techniques : Case studies on what type of projects we can do using Audio datasets
Lecture 4	Signal pre-processing : Signal preprocessing on audio anomaly dataset.
Lecture 5	Time-domain analysis : Time domain analysis on a suited dataset.
Lecture 6	Frequency-domain analysis : Frequency domain analysis Implementation via python
Lecture 7	Temporal analysis : Temporal analysis Implementation via python on a dataset.
Lecture 8	Statistical analysis : Statistical analysis Implementation via python on 2 datasets.
Lecture 9	Transformation : Transformation of dataset Implementation via python
Lecture 10	Selection and extraction of relevant features : Train, Test, & validation split via python
Lecture 11	Linear regression and gradient descent : Implementation of logistic and gradient via python
Lecture 12	Logistic regression and regularization : Logistic regression and regulation Implementation via python
Lecture 13	Decision trees and random forests : Decision tree and random forests implementation on a dataset.
Lecture 14	Support Vector Machines (SVMs) : Implementation of SVM via python
Lecture 15	Neural networks and deep learning : Implementation of basic neural model via python
Lecture 16	Clustering and dimensionality reduction : Implementation via python
Lecture 17	Evaluation of models and performance metrics Implementation via python to evaluate models. Implementation of confusion matrix
Lecture 18	Handling imbalanced data and overfitting : Overfitting and imbalanced data management using python.
Lecture 19	Feature engineering and selection : Discussion on how to select features along with implementation.
Lecture 20	Ensemble methods : Implementation of ensemble methods via python
Lecture 21	Introduction to reinforcement learning : What type of projects can be developed using it, case studies
Lecture 22	Current trend and future of machine learning : Intro to famous AI, ML, & DL based applications