FUTURE KEYS ®

< Learn to code >

# WHO ARE WE ?

Future Keys is a S.T.E.A.M. education academy. We use the approach of game-based curriculum rather than traditional learning. Kids love technology, games, and having fun with friends. Parents want to give their kids the best and brightest future possible. Everyone wins at Future Keys. Kids learn to code through our engaging game-based curriculum that teaches teamwork, logic, math, and problem solving. **They gain 21$^{st}$ Century skills while having fun. That's Future Keys way!**

Technology is advancing at an incredible rate, and we're just getting started. We believes that there's never been a more exciting time to advance our culture - starting with our youngest ones. Every kid deserves the chance to unlock their innate, unlimited capacity to learn.

Whether or not they pursue coding as a career, we want to provide kids with skills that will help them pursue their dreams.

We pride ourselves in partnering with entrepreneurs to bring 21st century learning centers to communities across the country.  Most importantly, we partner with families to provide children with a safe and encouraging environment to gain the skills they need for a brighter future. We look forward to earning your trust.

FUTURE KEYS®

# COURSES FOR 3 – 18 YEARS



## Younger Age
3 – 6 Years Old

## Middle Age
7 – 12 Years Old

## Teen Age
13+ Years old

FUTURE KEYS®

# MIDDLE AGE GROUP

Scratch
入门编程

APP Development
App 制作

Processing:
Draw with Code
代码绘图

Lego Ev3
Explorer
EV3 机器人
模型

Minecraft
我的世界

Roblox游戏

Unity Game
Development
游戏开发

# TEEN AGE GROUP

**Website Development**
网页开发

**Python**
语言

**AI Programing**
人工智能

**Unreal 3D Game Development**
虚拟3-D游戏开发

# YOUNGER AGE GROUP

# LEGO MACHINE EXPLORERS

**Short Description** - Students can observe and investigate simple machines. Discover mechanical principles by exploring, investigating, and solving tasks related to mechanical principles as well as learn about energy, balance, buoyancy, and much more.

**Difficulty Level -** Beginner

**Age Requirement –** Student must turn 3 years old

**Class duration –** 1 Hour

**Learning Outcomes –** 1. Understanding and observing simple machine mechanism
2. Improves problem solving skills
3. Increases English vocabulary
4. Creativity
5. Patience
6. Concentration

# OSMO: COLORFUL CODING BLOCKS

**Short Description** - The objective of this course is to introduce coding fundamentals and concepts to young imaginative minds. Students will use Osmo, an interactive and highly-visible iPad application that develops logical thinking and problem solving skill.

**Difficulty Level -** Intermediate

**Age Requirement –** Student must turn 4.5 years old

**Class duration –** 1 Hour

**Learning Outcomes –** 1. Fundamentals of Computer science
2. Improves problem solving skills
3. Increases English vocabulary
4. Team work
5. Individuality

**Class requirement –** iPad

FUTURE KEYS ®

# CODE YOUR OWN STORY

**Short Description** - This course aims to provide a fun and engaging introduction to foundation of computer science and coding with the Foos. Students will start with exploring and solving puzzles with simple, animated block-based commands,  after knowing all the programming blocks they will code their own story.
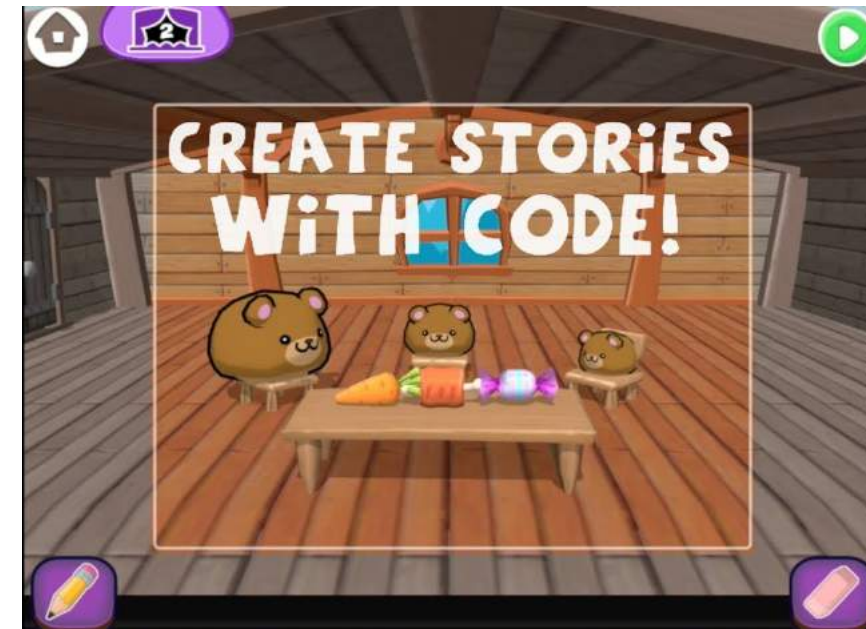
**Difficulty Level -** Advance

**Age Requirement –** Student must turn 5 years old

**Class duration –** 1 Hour

**Learning Outcomes –** 1. Fundamentals of Computer science
2. Improves problem solving skills
3. Increases English vocabulary
4. Logical thinking
5. Creativity

**Class requirement –** iPad/Android Tablet


CREATE STORIES WITH CODE!

FUTURE KEYS ®

# SCRATCH JR. IPAD GAMES

**Short Description** - This course introduces coding fundamentals and concepts to young imaginative minds. Students start with basic commands including moving sprites forward and turning left/right, then building their own fun stories and characters through sequences, loops, conditionals. Students are encouraged to construct creative games and detail the steps to bring their imagination to reality.

**Difficulty Level -** Intermediate

**Age Requirement –** Student must turn 6 years old

**Class duration –** 1 Hour

**Learning Outcomes –** 1. Fundamentals of Computer science
2. Improves problem solving skills
3. Increases English vocabulary
4. Logical and lateral thinking
5. Creativity

**Class requirement –** iPad/Android Tablet



FUTURE KEYS ®

MIDDLE AGE GROUP

# MAKE YOUR OWN MINECRAFT MODS

**Short Description** - This course puts focus on the design aspect of Minecraft Modding - students will be learning how to design a proper image and get it to work as a mod in Minecraft. Students will learn the basics in how to create mods, to draw 2D images using Pixlr and turn them into 3D objects in Minecraft.
Through modding, each student develops a completely unique game world. Students will be able to share their unique worlds with others, creating a fun and educational experience.

**Difficulty Level -** Beginner

**Age Requirement –** Student must turn 6 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Creativity
                       2. Game Modding
                       3. 2D/3D Art Design
                       4. Design basics
                       5. Individuality

**Class requirements –** MacBook/ Laptop and a mouse.

# SCRATCH : 2D COMPUTER GAMES

**Short Description** - Students learn key computer science concepts such as sequences, loops, conditionals and abstraction by building their own fun projects of animated stories and multimedia games. Students are encouraged and supported by professionally trained instructors to design creative games and to think about the programming processes behind the apps and games they see in daily lives.

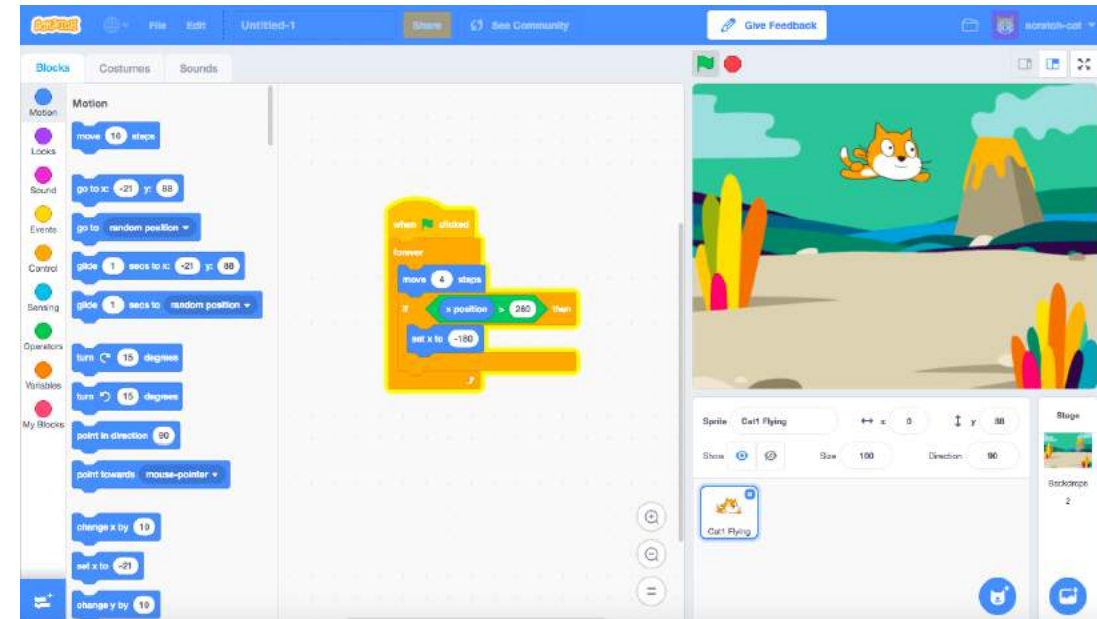**Difficulty Level –** Beginner, Intermediate

**Age Requirement –** Student must turn 7 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Block Base Programing
2. Concepts of Computer Science
3. 2D Game Design and Development
4. Lateral thinking and Creativity
5. Individuality

**Class requirements –** MacBook/ Laptop

# LEGO ROBO EXPLORERS

**Short Description** - This course emphasizes on fundamentals of robot engineering and computer programming. It teaches students the computer programming logic and reasoning skills using a robotics engineering context.

**Difficulty Level –** Beginner, Intermediate and Advance

**Age Requirement –** Student must turn 7.5 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Robotics
2. Concepts of Computer Science
3. Knowledge of servo motors and sensors
4. Lateral thinking
5. Team work
6. Block Base Programing

**Class requirements –** MacBook/ Laptop

# MAKE YOUR OWN MINECRAFT MODS

**Short Description** - This course puts focus on the design aspect of Minecraft Modding - students will be learning how to design a proper image and get it to work as a mod in Minecraft. Students will learn the basics in how to create mods, to draw 2D images using Pixlr and turn them into 3D objects in Minecraft.
Through modding, each student develops a completely unique game world. Students will be able to share their unique worlds with others, creating a fun and educational experience.

**Difficulty Level –** Beginner and Intermediate

**Age Requirement –** Student must turn 8 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Block Base Programing
2. Game Modding
3. 3D Modeling
4. Design thinking
5. Design Process

**Class requirements –** MacBook/ Laptop and a mouse.



FUTURE KEYS ®

# 3D ROBLOX : GAME MOD

**Short Description** - In this course, we'll learn to code Roblox games using the language Lua. Lua is a lightweight programming language which is designed for multiple applications such as games and web applications. Apart from learning to code in Lua, we'll also learn to use other features such as creating constraints to objects and adding GUIs to our games on Roblox Studio.

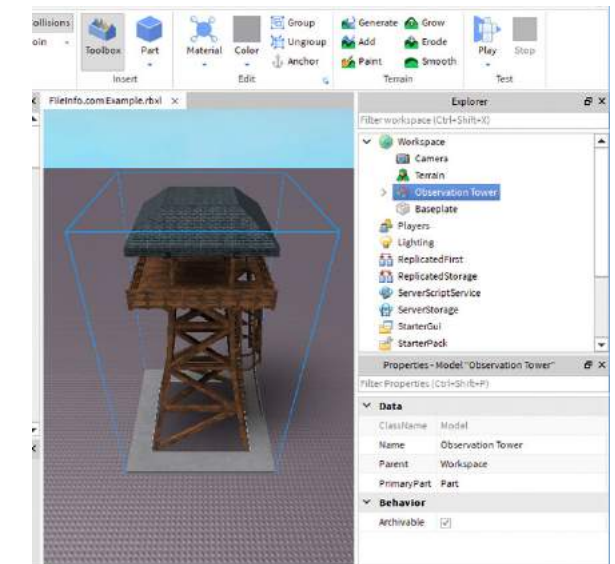**Difficulty Level –** Beginner, Intermediate and Advance

**Age Requirement –** Student must turn 8 years old
- Must know block base programming

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Game Modding
2. Lua (Programing Language)
3. 3D Game Design
4. 3D Modeling Basics
5. Syntax Programing

**Class requirements –** MacBook/ Laptop and a mouse.





FUTURE KEYS ®

# APP MAKERS

**Short Description** - This course aims to empower students through teaching them problem-solving using technology. From introducing fundamental computer science concepts, applying Design Thinking process to utilizing block base programing language, we guide students to apply problem-solving skills to their daily lives..

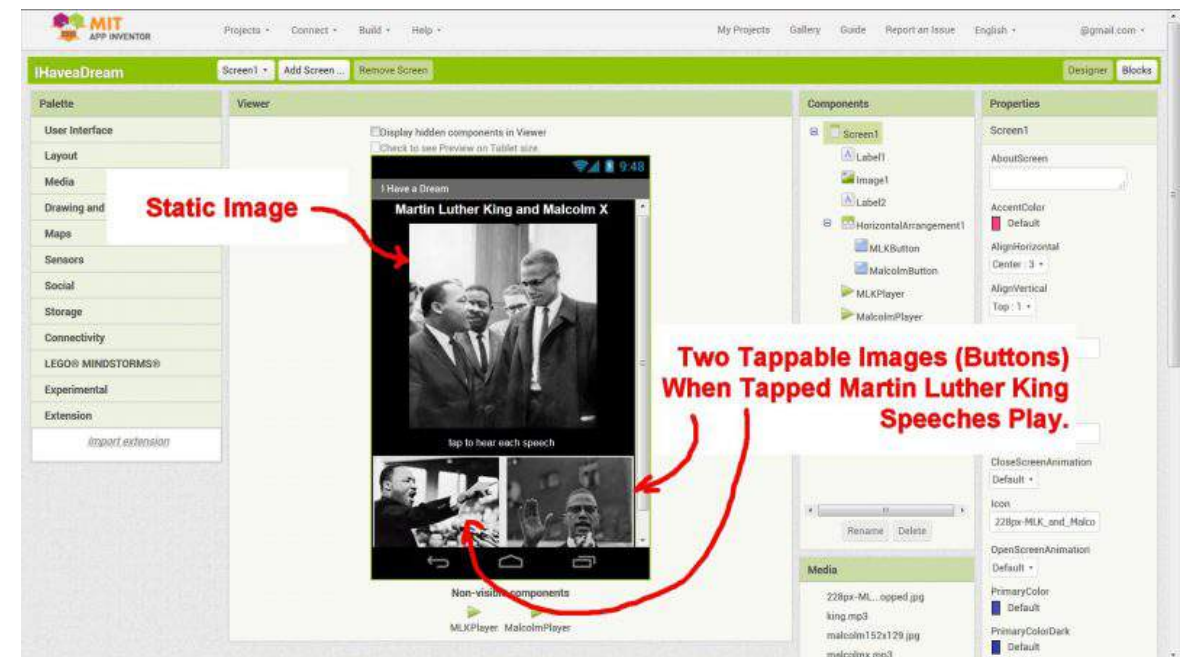**Difficulty Level –** Intermediate and Advance

**Age Requirement –** Student must turn 8 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Mobile App Development
2. UI/UX Design
3. Design thinking
4. Block base programing
5. Prototyping

**Class requirements –** MacBook/ Laptop and a mouse.



FUTURE KEYS®

# DRAW WITH CODE

**Short Description** - This course is designed for the creative minded. Within the context of the visual coding sketches, students will learn how to code in Processing, a Java-based, flexible software tool which serves as a gentle introduction to typing based coding languages. Through programming visual sketches, students will be introduced to key computer science concepts like functions and loops, as well basic design theories such as the golden ratio.
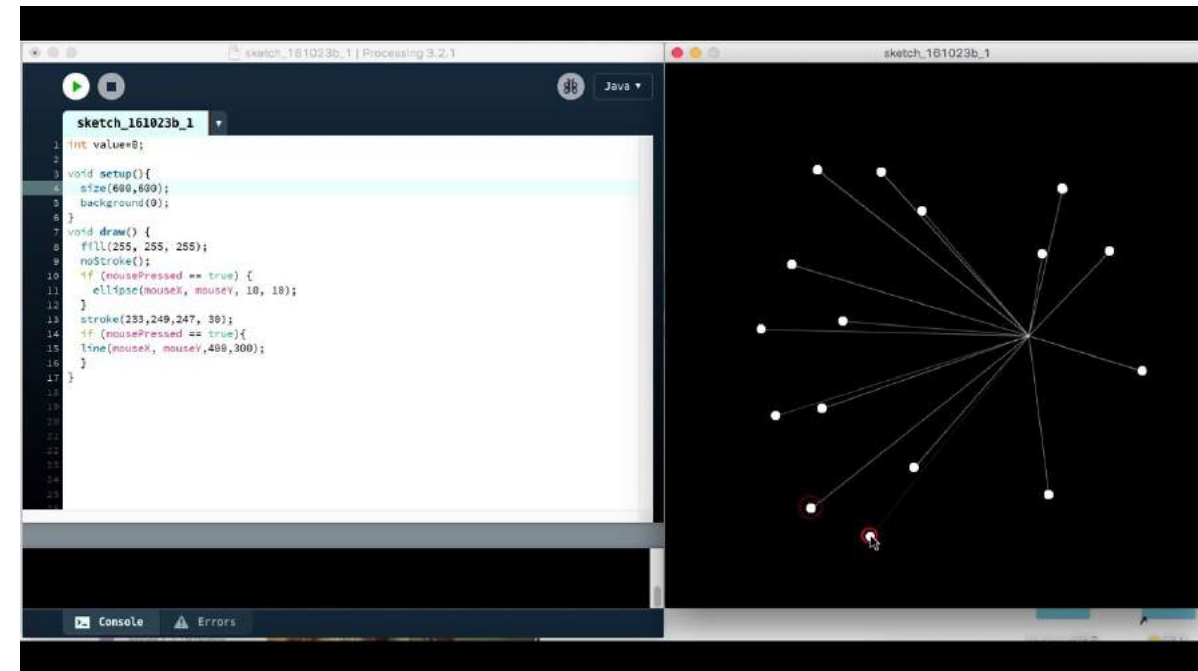
**Difficulty Level –** Intermediate

**Age Requirement –** Student must turn 10 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Java Script (Programing Language)
2. Design Basics
3. Color theory
4. 2D/3D Art
5. Syntax Programing

**Class requirements –** MacBook/ Laptop and a mouse.

# UNITY 3D GAME MAKING

**Short Description** - In this course, we first provide a gentle introduction to 3D modeling using Blender, a professional 3D design tool. Students gain hands-on experience in 3D geometry and create fun 3D objects. Using these custom objects created in Blender, students are then introduced to the programming language C# to program their own 3D world in Unity.
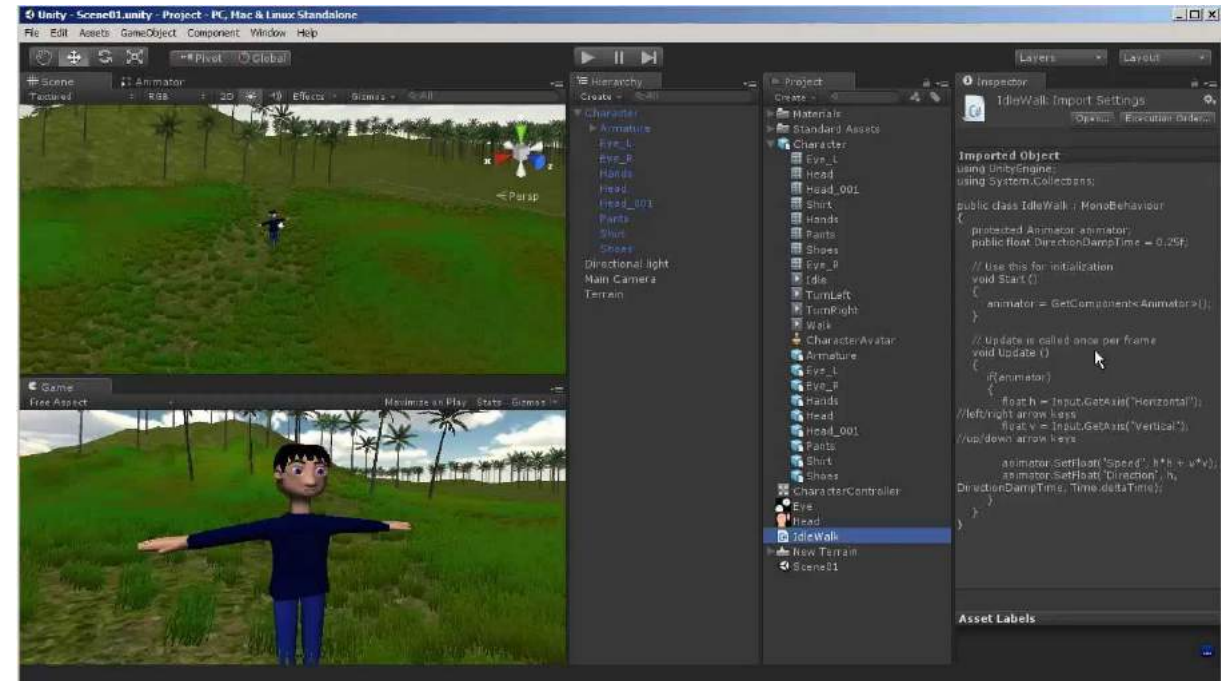
**Difficulty Level –** Advance

**Age Requirement –** Student must turn 12 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. C#(Programing Language)
                              2. 3D Modelling
                              3. Game Development
                              4. Design Thinking
                              5. Syntax Programing



**Class requirements –** MacBook/ Laptop and a mouse.

# TEEN AGE GROUP

# PYTHON

**Short Description** - Designed for any student curious about the power of computers and technology, the Python course introduces the basics of Python and computer science by programming a series of fun games and rewarding code challenges. In addition to Python syntax and computer science concepts, students also learn to think about problems logically and solve them systematically. The course also introduces Python libraries to students, laying a solid foundation towards many more powerful applications of Python at higher levels.

**Difficulty Level –** Intermediate and Advance

**Age Requirement –** Student must turn 14 years old

**Class duration –** 1.5 Hour

**Learning Outcomes –** 1. Python Programing Language)
2. Algorithm Design
3. Logical reasoning
4. Advance concepts from Computer Science
5. Syntax Programing

**Class requirements –** MacBook/ Laptop

# AI PROGRAMING

**Short Description** - This course will be taught using Python - one of the most trending languages in the field of AI. Students are required to have basic knowledge in Python which includes the basics syntaxes, loops, conditionals, and functions. Students who have not learned Python before are required to possess an intermediate level of programming knowledge in at least one programming languages (e.g. Java, Ruby, C#, Swift). Basic knowledge in elementary geometry and algebra is assumed. Knowledge in math topics such as system of linear equation, quadratic equation, matrix and calculus is a huge plus (not required).

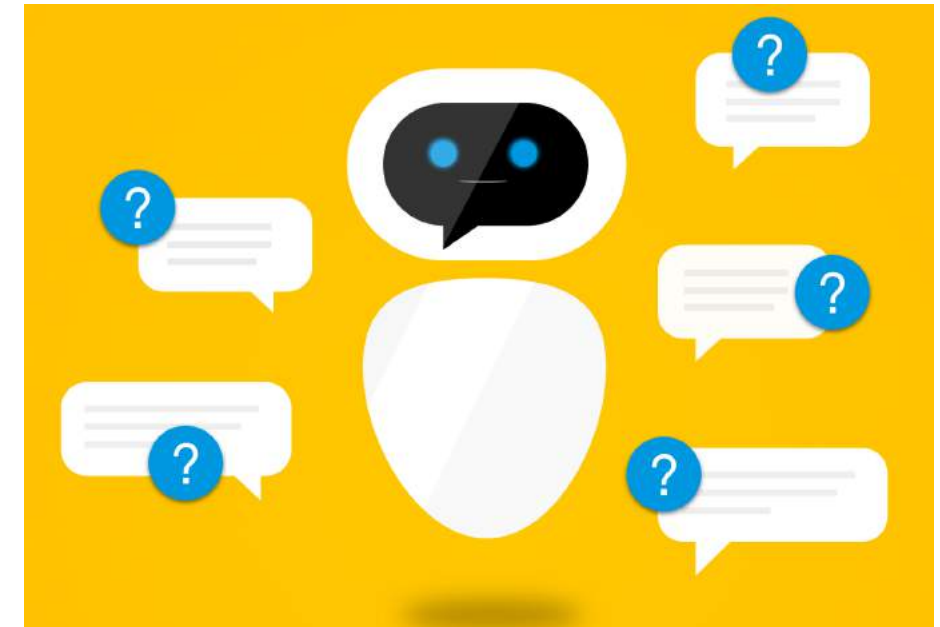**Difficulty Level –** Intermediate and Advance

**Age Requirement –** Student must turn 15 years old

**Class duration –** 1.5 Hour
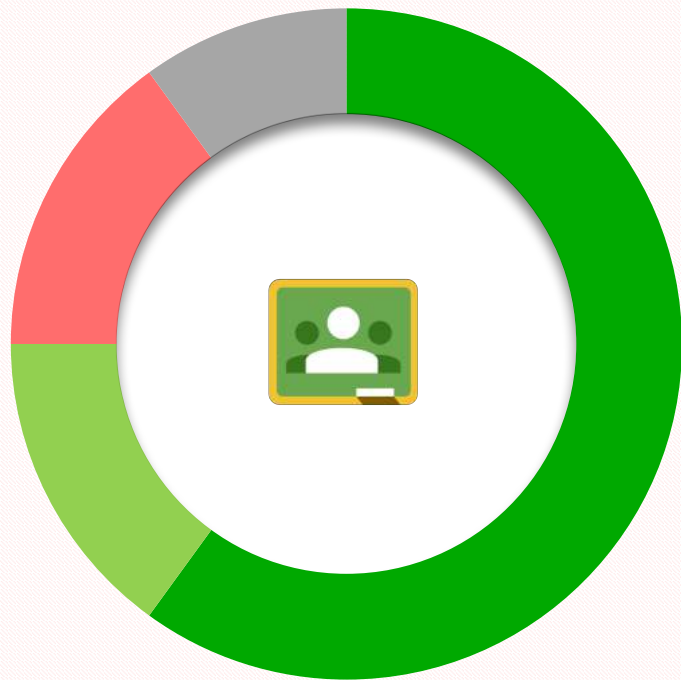
**Learning Outcomes –** 1. Python Programing Language)
2. Algorithm Design
3. Complex math
4. Advance concepts from Computer Science
5. Machine Learning

**Class requirements –** MacBook/ Laptop

# OUR CLASSROOM STANDARDS

Project Base Learning : Hands On

Lesson - Understanding Key terms

Unplugged Activity

Recap

FUTURE KEYS ®

# STANDARDS

**Unplugged Activities**
Activities designed to aid the teaching and learning of technology through engaging games and puzzles using cards, string, crayons and lots of running around.
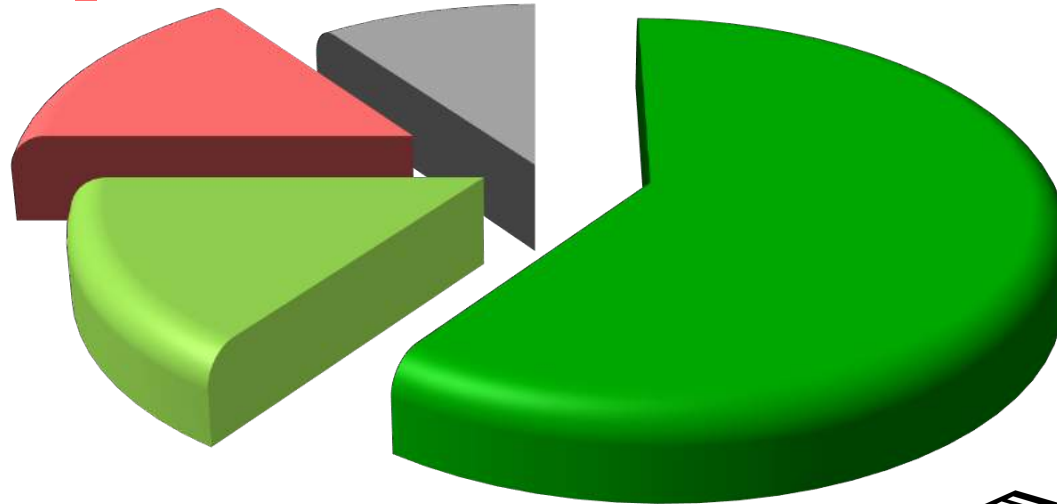
**15%**

**60%**

**Project Based Learning**
- Active learning.
- Teachers present problems and students solve together through group work.
- Teachers be flexible, supportive and engaged in the learning process
- Expect and forgive mistakes
- Interest -based selection of projects, accommodate to different students
- Help students develop real-world skills

**Recap**
- Reviewing previous project
- Improves observation skills
- Helps in memory retention

**10%**

**15%**

**Lesson– Understanding Key Terms**
- Improves Vocabulary
- Relating to daily life
- Tons of Examples
- Class Activities

FUTURE KEYS ®

# CLASSROOM RULES

### Done is Better than Perfect

Motto from Facebook's headquarters. Mark Zuckerberg reminds engineers that software will never be perfect. Why is it important for us?



### Do things tell people

Motto From Tesla's CEO Elon Musk. To make sure we need to convey our idea to other people in order to ensure its strength.



### Respect Everyone & Everything

This is to remind kids that please be nice and maintain the classroom environment.

FUTURE KEYS ®

# MOTIVE BEHIND CLASSROOM RULES

- **Inspiration** –  Inspiration is really important in transforming the way we perceive our own capabilities.

- **Positivity** - If you're doing what you love, work doesn't feel like work. That's why positivity and fun is baked into every aspect of what we do, from our game-based curriculum, to the joy of bringing Future Keys to your community.

- **Efficiency** - By providing examples from big entrepreneurs like a quote from Facebook CEO Mark Zuckerberg "Done is better than prefect" that means maxim for all of us prone to endlessly tweaking our projects and never finishing them. The key to moving ahead is following through.

- **Communication Skills**– Its very important to convey your ideas, making students speak up will help in their personal development. So each course end with a presentation by the students.

- **Discipline** - Discipline brings stability and structure into classroom environment. And has a very important role in learning process.

FUTURE KEYS ®