



# **PyOhio 2020**

## **Online Lightning & Thunder Talk Edition**

# **Python in Education for Generation Z**

**Gajendra Deshpande**

**KLS Gogte Institute of Technology, India**

**25-26 July 2020**

# Agenda

- ▣ The Generation Z, BYOD and Education
- ▣ BYOD: Advantages and Disadvantages
- ▣ QPython
- ▣ Blockly
- ▣ Flowgorithm
- ▣ VisuPy
- ▣ TextBook Companion Project

# The Generation Z, BYOD and Education

- ▣ Gen Z : born between 1995 and 2010
- ▣ More Tech Savvy
- ▣ Learn best by doing/creating
- ▣ Teachers need to be equipped with the technology

People born from 1995 to 2010—are true digital natives: from earliest youth, they have been exposed to the internet, to social networks, and to mobile systems.

With BYOD you are creating a 1:1 classroom. Students bring and use their choice of technological devices in the classroom.

# BYOD: Advantages and Disadvantages

CISCO DevNet - Securing and increasing productivity of BYOD in classrooms at schools (AICTE India)

## The benefits of BYOD

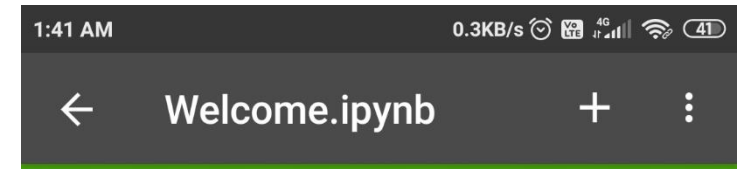
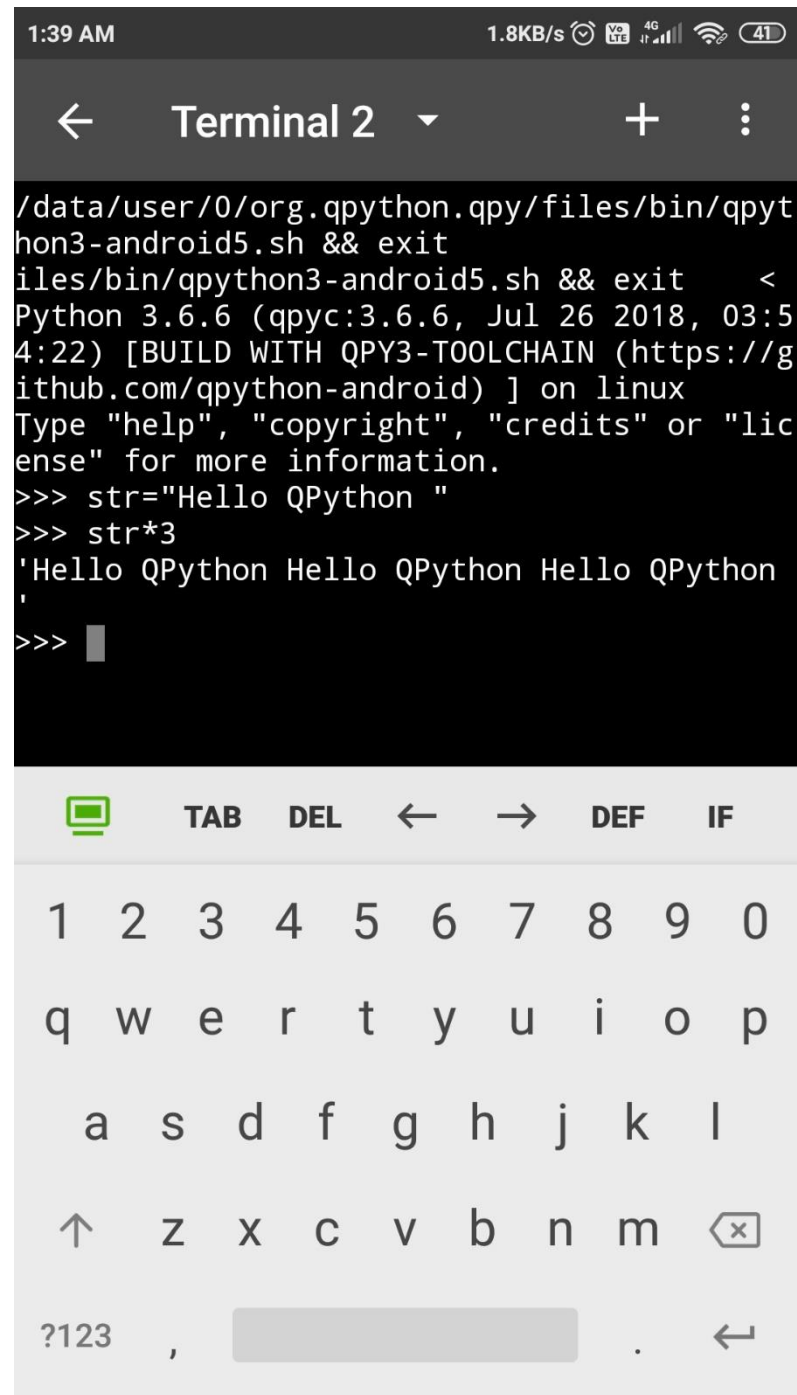
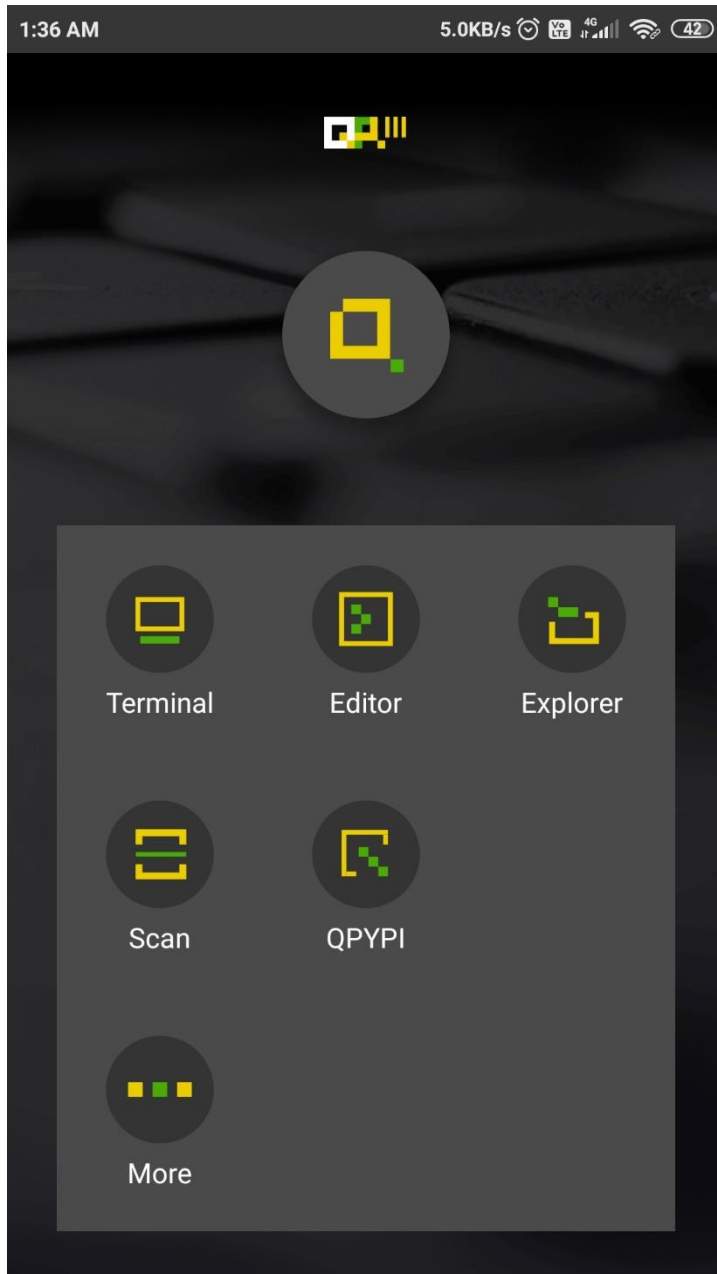
- ▣ Your students know the device
- ▣ Technology has many possibilities
- ▣ Cutting-edge devices
- ▣ Cost Effective
- ▣ Learning outside the school hours
- ▣ Respect for the device
- ▣ Organized students

## Disadvantages of BYOD

- ▣ Students without devices
- ▣ Different devices
- ▣ Distraction
- ▣ Not-responsible student

Austria, USA, Estonia,  
Australia, Finland, Norway,  
Portugal, Switzerland, UK

# QPython



## Welcome to use the QPython Notebook service!

This Notebook Service was **launched just for you**. It's a temporary way for you to try out a recent development version of the IPython/Jupyter notebook.

Thanks to [Jupyter](#), [IPython](#) etc., QPython Notebook is built based on these excellent opensource projects.

## Run some Python code!

To run the code below:

1. Click on the cell to select it.
2. Press the play button (▶) in the toolbar bottom.

A full tutorial for using the QPython Notebook interface is available [here](#).



Website: <https://www.qpython.com/>

# Blockly

The image shows the Blockly web interface. On the left is a category menu with options: Logic, Loops, Math, Text, Lists, Color, Variables, and Functions. The main workspace contains a script with the following blocks: a 'set Count to 1' block, a 'repeat while' loop block with 'Count ≤ 3' as the condition, and a 'do' block containing a 'print "Hello World!"' block and a 'set Count to Count + 1' block. The 'print' block is highlighted with a yellow border. On the right, the 'Language: Python' dropdown is selected, and the corresponding Python code is displayed: 

```
Count = None

Count = 1
while Count <= 3:
    print('Hello World!')
    Count = Count + 1
```

 A large blue play button is located at the bottom right of the interface.

Website: <https://developers.google.com/blockly>

# Flowgorithm

Website: <http://www.flowgorithm.org/>

The screenshot displays the Flowgorithm IDE interface, which includes a menu bar (File, Edit, Program, Tools, Help), a toolbar with icons for file operations and execution, and a main workspace divided into three panels: a flowchart editor, a source code viewer, and a console.

**Flowchart:** The flowchart illustrates an algorithm to calculate the sum of the first  $n$  natural numbers. It begins with a 'Main' terminal, followed by a declaration of integer variables  $i$ ,  $n$ , and  $sum$ . The variable  $sum$  is initialized to 0. An output statement prompts the user to 'Enter value for n:'. An input statement receives the value of  $n$ . A loop is then executed from  $i = 0$  to  $n$ . Inside the loop, the value of  $sum$  is incremented by  $i$  ( $sum = sum + i$ ). After the loop, the program outputs the 'Sum of First  $n$  & numbers is: ' and then the final value of  $sum$ . The process concludes at an 'End' terminal.

```
graph TD
    Main([Main]) --> Decl[Integer i, n, sum]
    Decl --> Init[sum = 0]
    Init --> Out1[/Output "Enter value for n:"/]
    Out1 --> In1[/Input n/]
    In1 --> Loop{i = 0 to n}
    Loop -- Next --> SumAdd[sum = sum + i]
    SumAdd --> Loop
    Loop -- Done --> Out2[/Output "Sum of First "&n "&" numbers is: "/]
    Out2 --> Out3[/Output sum/]
    Out3 --> End([End])
```

**Source Code Viewer:** The Python code corresponding to the flowchart is displayed in the source code viewer. It includes comments for each step of the algorithm.

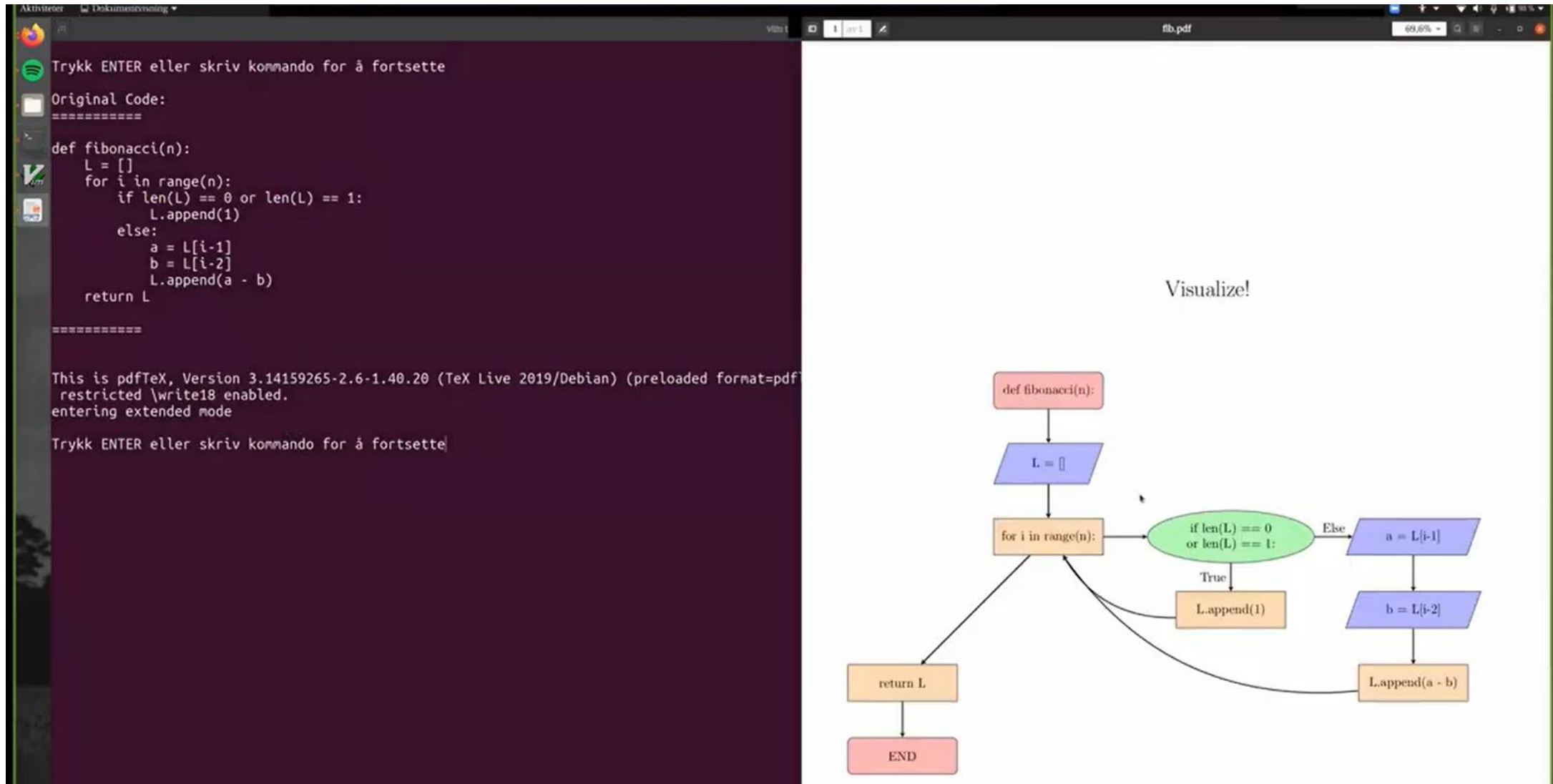
```
0 sum = 0
1 print("Enter value for n:")
2 n = int(input())
3 for i in range(0, n + 1, 1):
4     sum = sum + i
5 print("Sum of First " + str(n) + " numbers is: ")
6 print(sum)
```

**Console:** The console shows the execution of the program. It displays the prompt 'Enter value for n:', the user input '5', the output 'Sum of First 5 numbers is:', and the final result '15'.

```
Enter value for n:
5
Sum of First 5 numbers is:
15
```

Font size set to 9pt. EN

# VisuPy- Code Visualization



# Python Textbook Companion Project by FOSSEE

The screenshot shows a web browser window with the URL [python.fossee.in/textbook-companion-project/](https://python.fossee.in/textbook-companion-project/). The page features the Python logo and a navigation menu with links: Home, Resources, Gallery, About, Testimonials, Contact Us, and Fellowship 2020. On the left, there are two vertical menus. The first menu, under the heading 'Activities', lists: Python Workshops, Python Self Learning Course, Textbook Companion (highlighted), TBC for Academics, Forum, and Spoken Tutorials. The second menu, under the heading 'FOSSEE', lists: Scilab, eSim, OpenFOAM, R, and Osdag. At the bottom of the left sidebar is a yellow button labeled 'SciPy India'. The main content area is titled 'Textbook Companion Activity (TBC)'. It contains a section titled 'Objective' with the following text: 'The Textbook Companion activity aims to create a repository of reference material for Python by coding solved examples of standard engineering textbooks using Python. This activity intends to -'. Below this is a bulleted list: 

- Make individuals learn Python through a practical approach
- Provide a huge database of Companions as a learning resource
- To make it easy for users of such textbooks to start using Python
- To improve the documentation available for Python

Following the list is a paragraph: 'Use the completed Textbook Companions for academics. [Click here](#) to have a look at the same.' Below this are five more sections, each with a title in a grey box: 'Who can Contribute ?', 'How to contribute ?', 'Guidelines for Coding', and 'Goodies'.

Textbook Companion Activity (TBC)

Objective

The Textbook Companion activity aims to create a repository of reference material for Python by coding solved examples of standard engineering textbooks using Python. This activity intends to -

- Make individuals learn Python through a practical approach
- Provide a huge database of Companions as a learning resource
- To make it easy for users of such textbooks to start using Python
- To improve the documentation available for Python

Use the completed Textbook Companions for academics. [Click here](#) to have a look at the same.

Who can Contribute ?

How to contribute ?

Guidelines for Coding

Goodies

Website: <https://tbc-python.fossee.in/>

Thank You!