

THE IMPORTANCE OF VISUALS FOR TEACHING PYTHON

TADEH HAKOPIAN



LIGHTNING TALKS

ABOUT ME

- Architect who likes to code
- Uses Python in day to day work
- Tries to encourage others to use Python and open source tools
- Struggles to teach people coding because they're not used to it

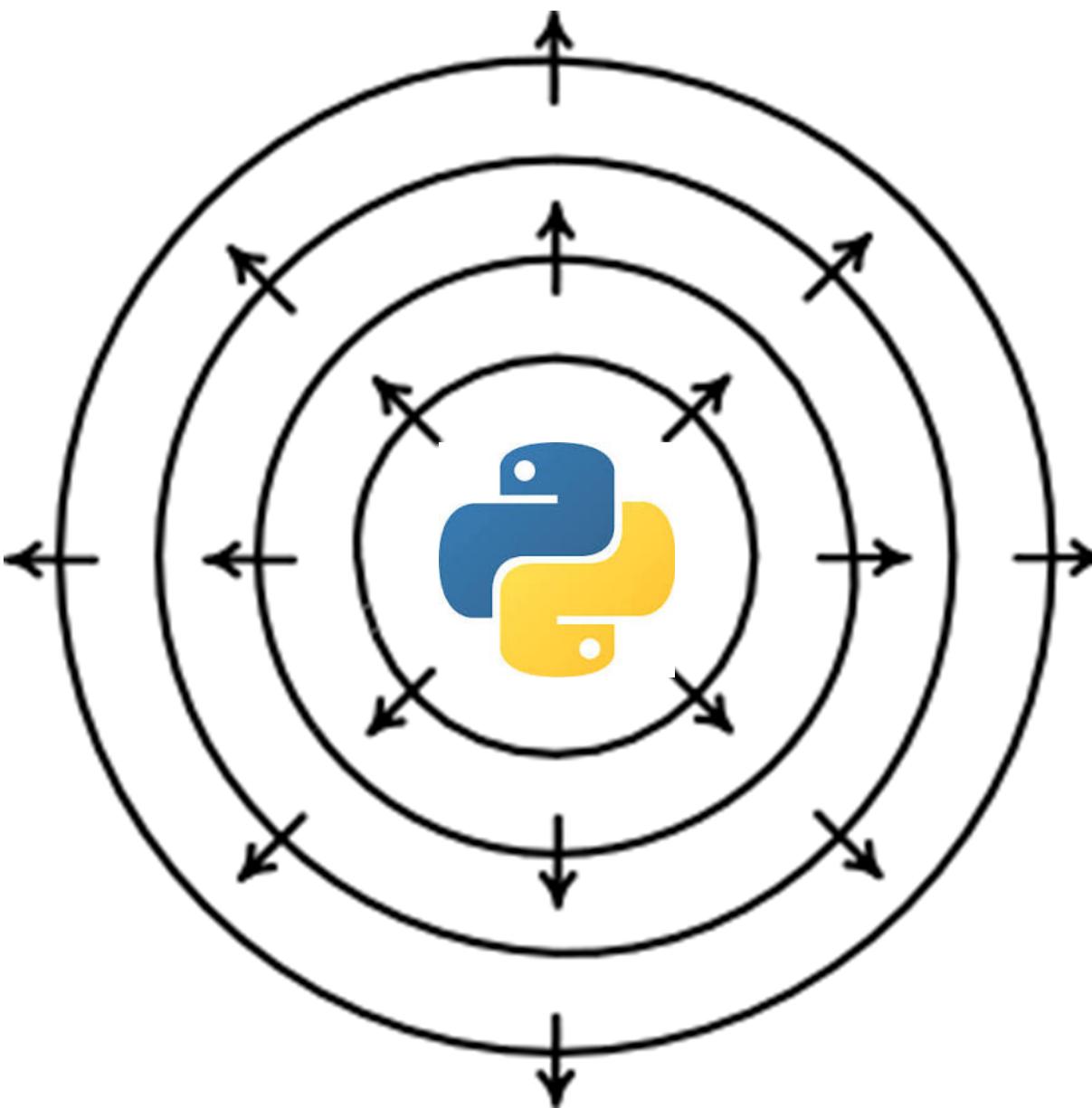


THE MESSAGE

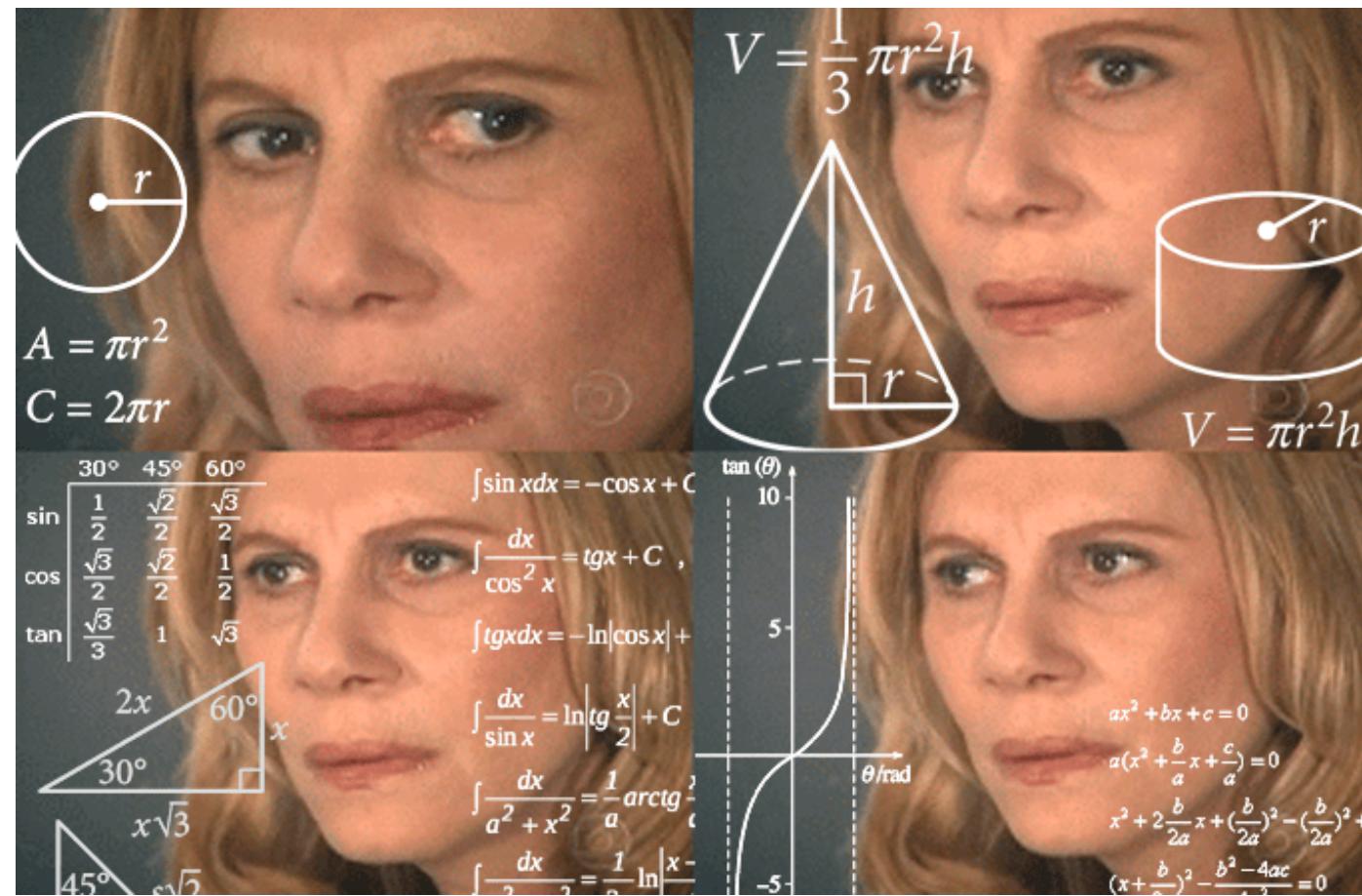


- Make learning code accessible for anyone

EXPAND THE CIRCLE



INTIMIDATION FACTOR

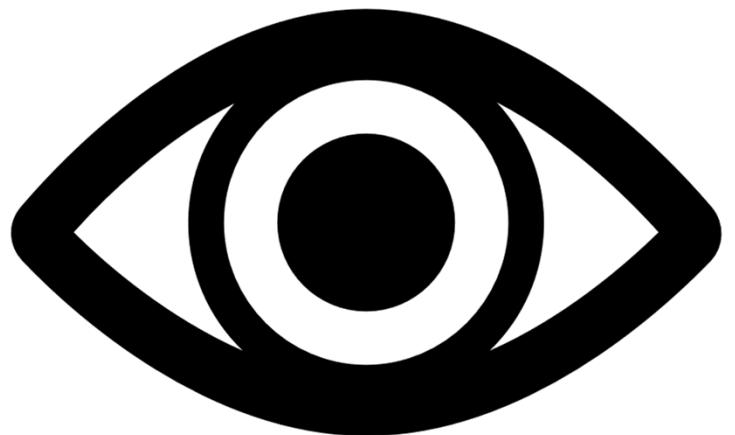


REMOVE BARRIERS

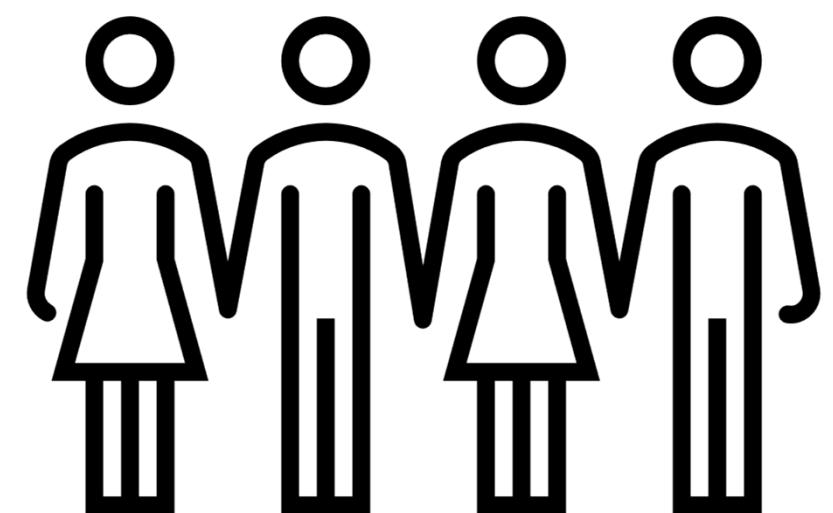
Make the first step non-intimidating



Emphasize visuals and graphics for creatives who are comfortable in that format

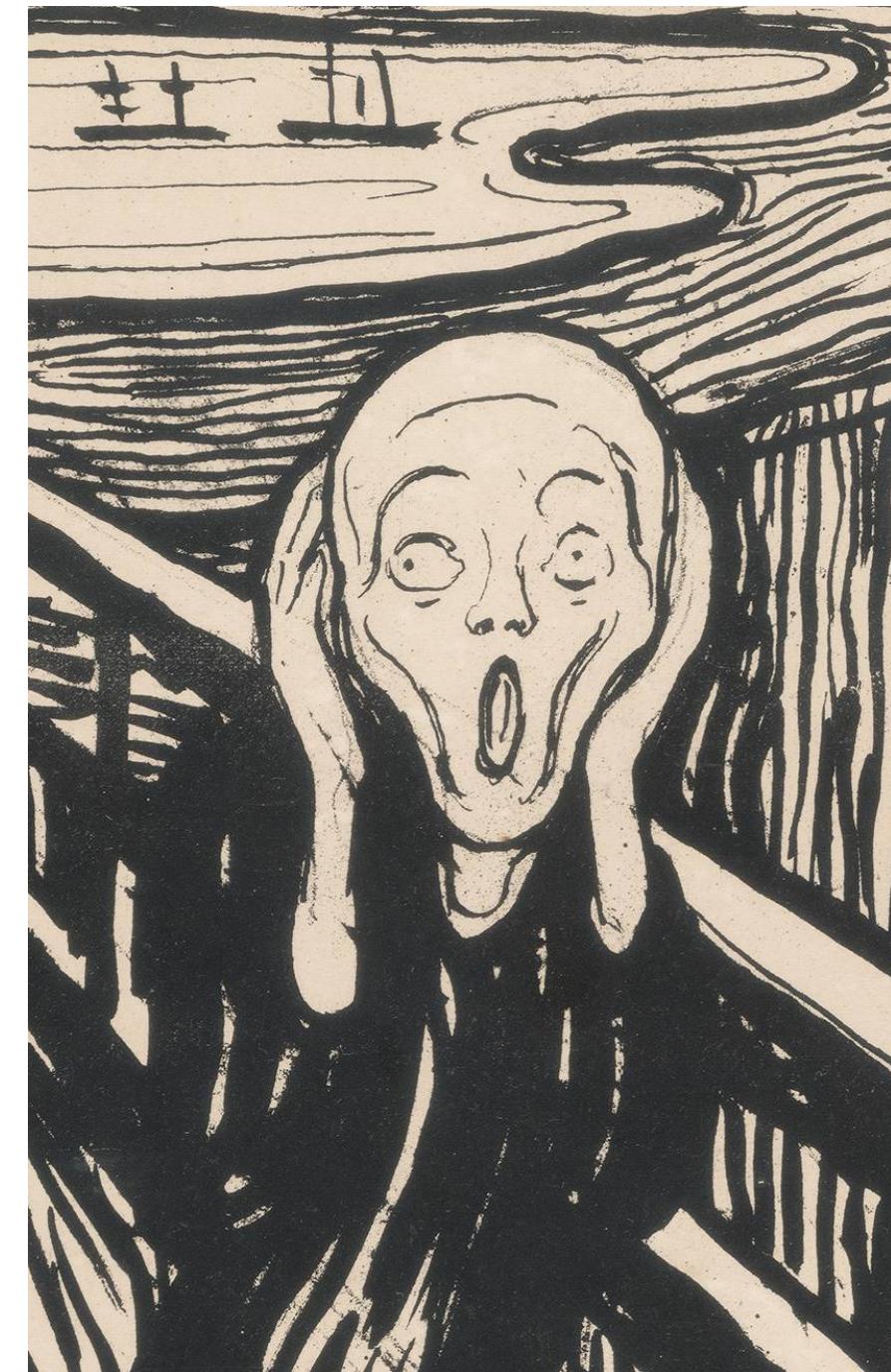


The more methods of training people the better – find your fit



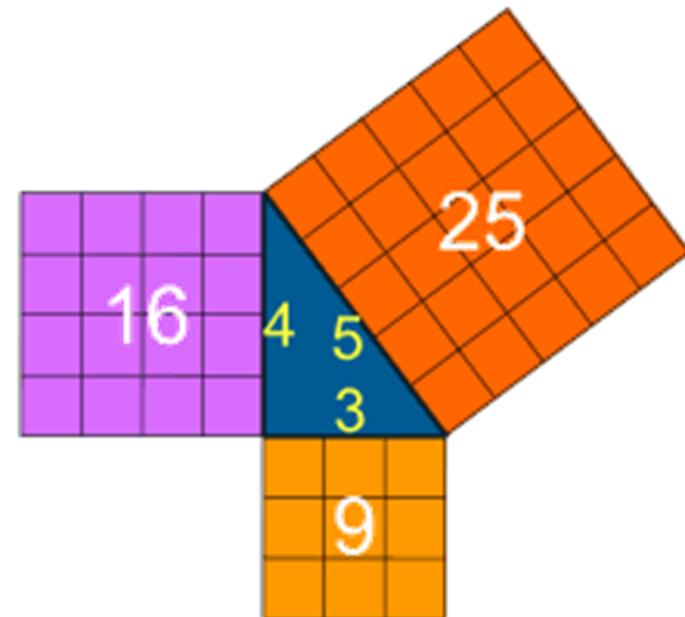
TEXT EDITOR CODE

```
1 import random
2 from flask import Flask, render_template, \
3     abort, url_for, flash
4 from flask.ext.sqlalchemy import SQLAlchemy
5
6 app = Flask(__name__)
7 app.config.from_object('config')
8 db = SQLAlchemy(app)
9
10 from models import Answer, Question
11
12 # routes
13 @app.route('/', methods=['GET'])
14 def home():
15     question = get_question()
16     options = get_options(question)
17     return render_template('home.html', question=question, options=options)
18
19
20 @app.route('/answer/<int:answer_id>/<string:question>')
21 def answer(answer_id, question):
22     updated_question = str(question)
23     answer_query = db.session.query(Answer).filter(Answer.answer_id == answer_id).first()
24     question_query = db.session.query(Question).filter(Question.description == updated_question).first()
25     right_answer = db.session.query(Question).filter(Question.description == updated_question).first()
26     answer = get_correct_answer(right_answer)
27     if answer_query.question_id == question_query.question_id:
28         correct = flash("Correct!")
29         return render_template('check.html', correct=correct)
30     else:
31         incorrect = flash(("Wrong. The correct answer is \"{}\"").format(right_answer.description))
32         return render_template('check.html', incorrect=incorrect)
33
34
35 # helper functions
36 def get_question():
37     rand = random.randrange(0, db.session.query(Question).count())
38     question = db.session.query(Question)[rand]
39     return question
40
41
42 def get_options(question):
43     ...
```



VISUAL LEARNING – PYTHAGOREAN THEOREM

```
1 import math  
2  
3 def pathag(sidea, sideb):  
4     csquare = (sidea ** 2) + (sideb ** 2)  
5     sidec = math.sqrt(csquare)  
6     return sidec  
7  
8 a = float(input('Side A: '))  
9 b = float(input('Side B: '))  
10  
11  
12 test1 = pathag(a, b)  
13 print "Side C is", test1,  
14  
15 #Written by Ben
```



ANIMATE

Python 3.6

```
→ 1 word = "oxygen"  
  2 for char in word:  
  3     print(char)
```

[Edit this code](#)

→ line that has just executed

→ next line to execute

Click a line of code to set a breakpoint; use the Back and Forward buttons to jump there.

Print output (drag lower right corner to resize)

Frames

Objects

<< First

< Back

Step 1 of 14

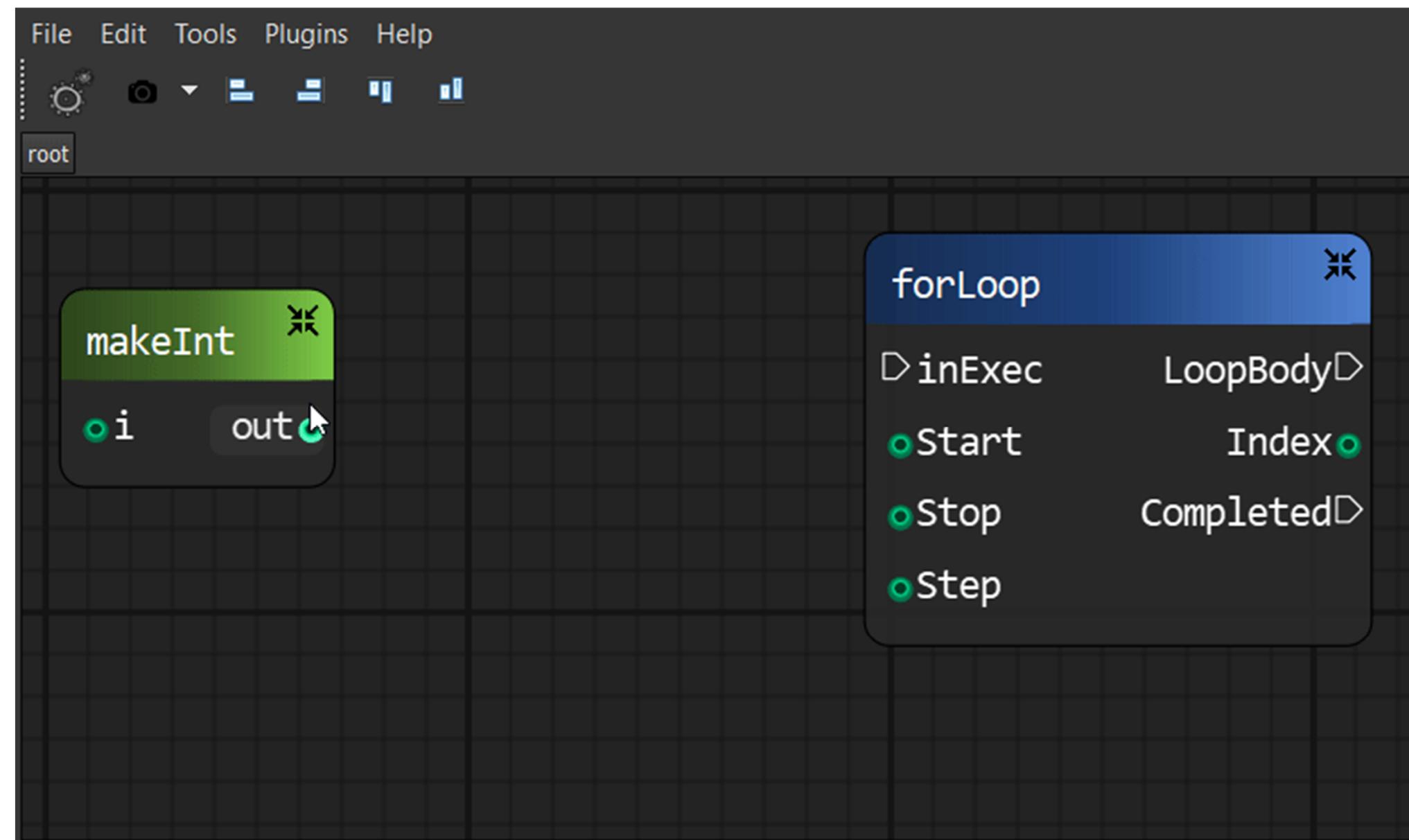
Forward >

Last >>

CODE BLOCKS



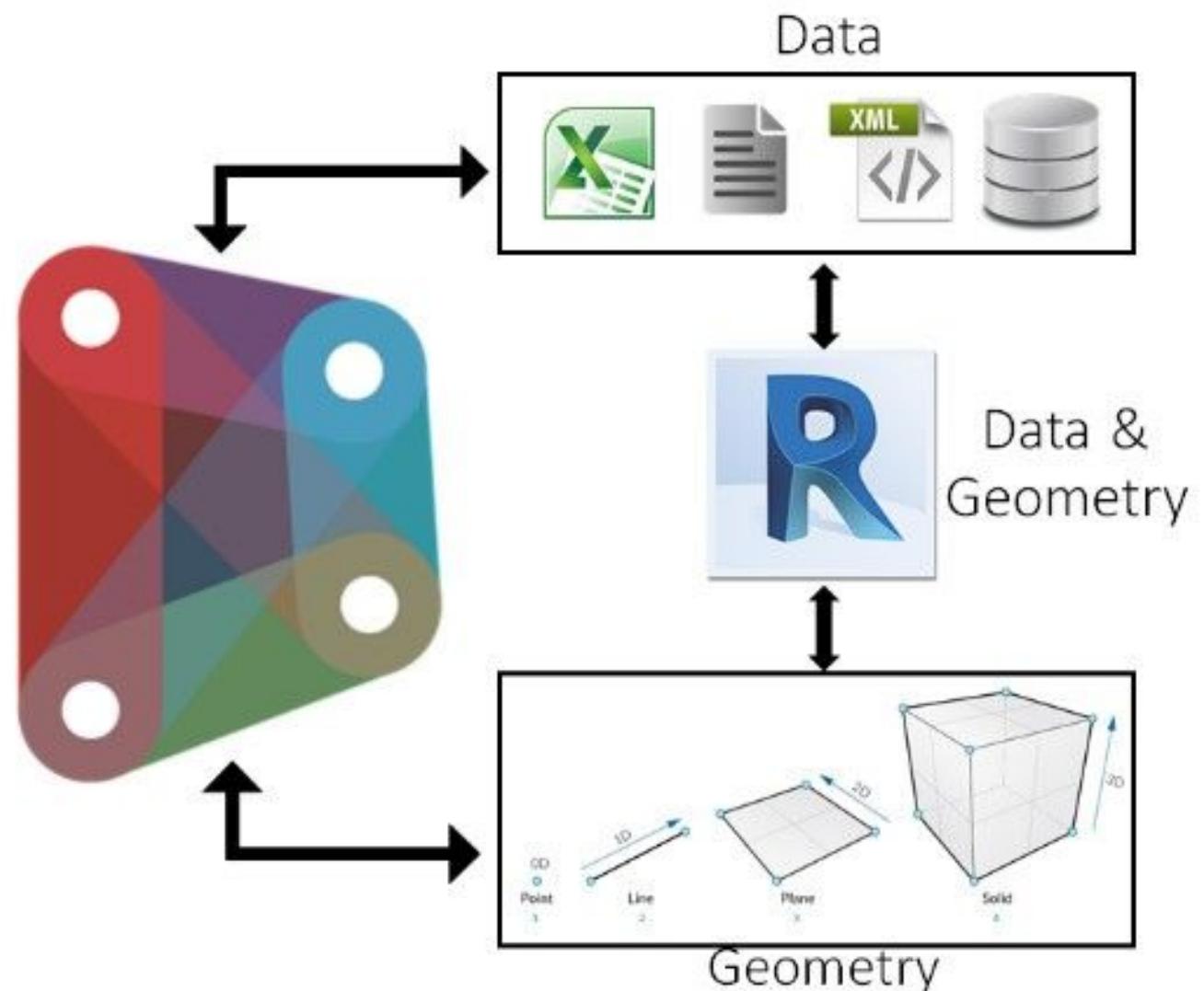
PYFLOW



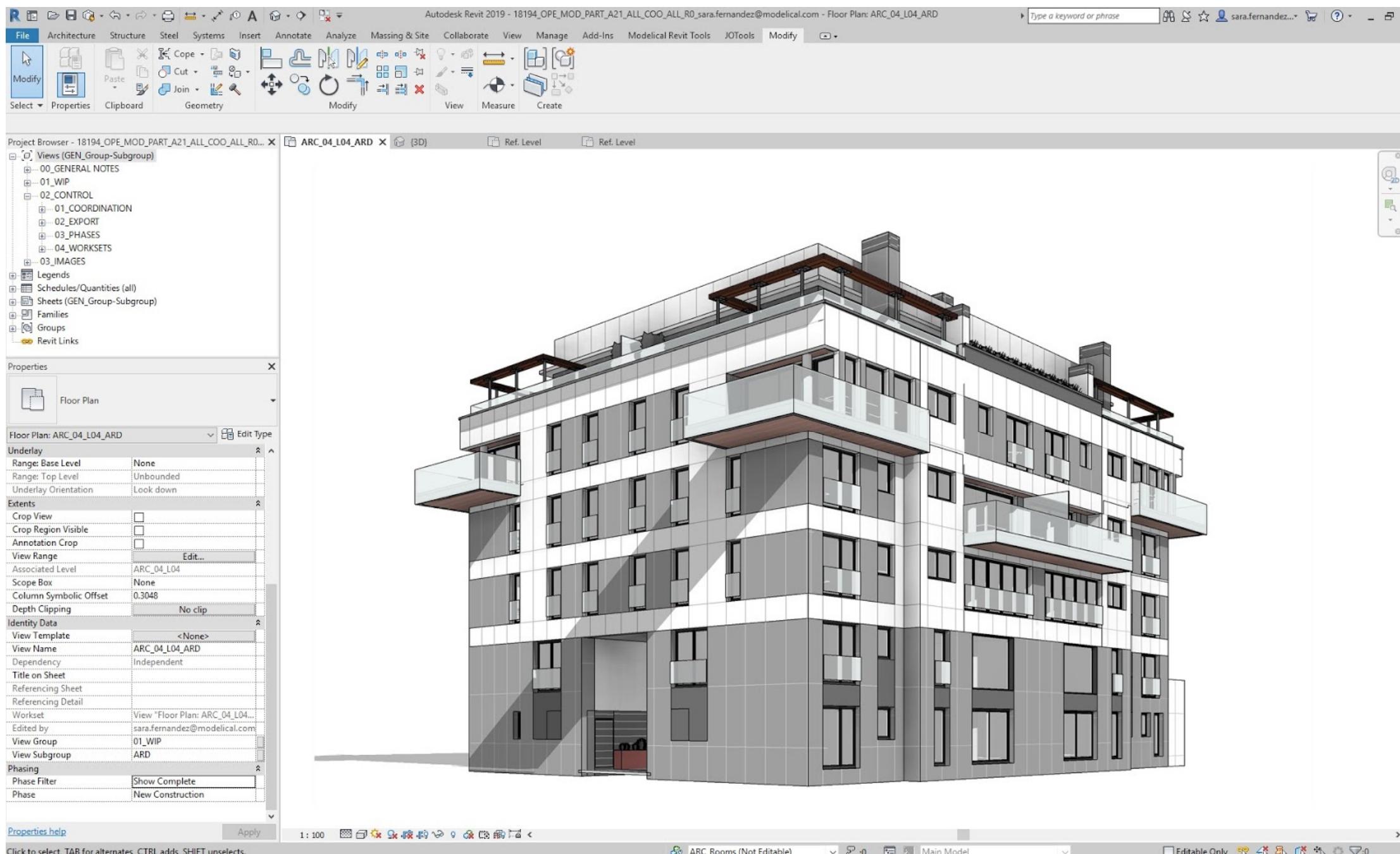
THE BREAKTHROUGH



DYNAMO



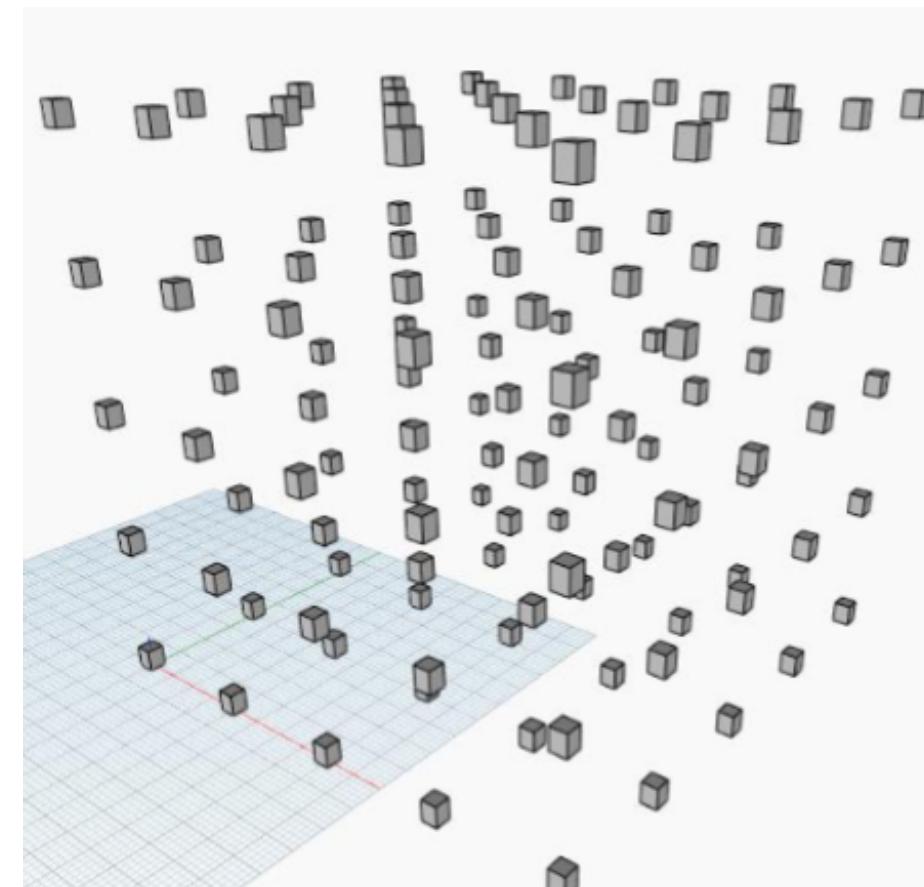
3D MODELING



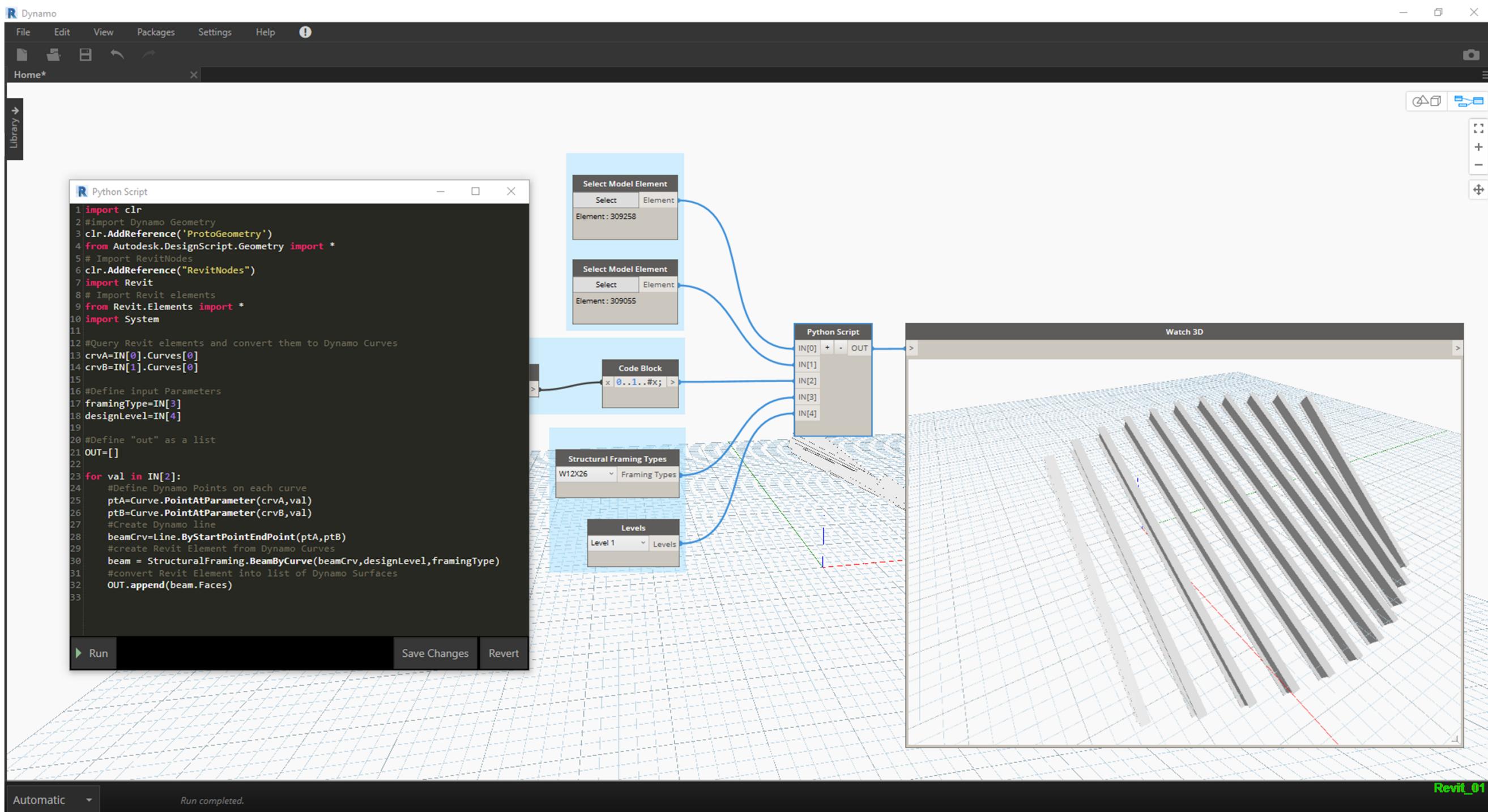
DYNAMO

R PS - XYZ matrix Family PY

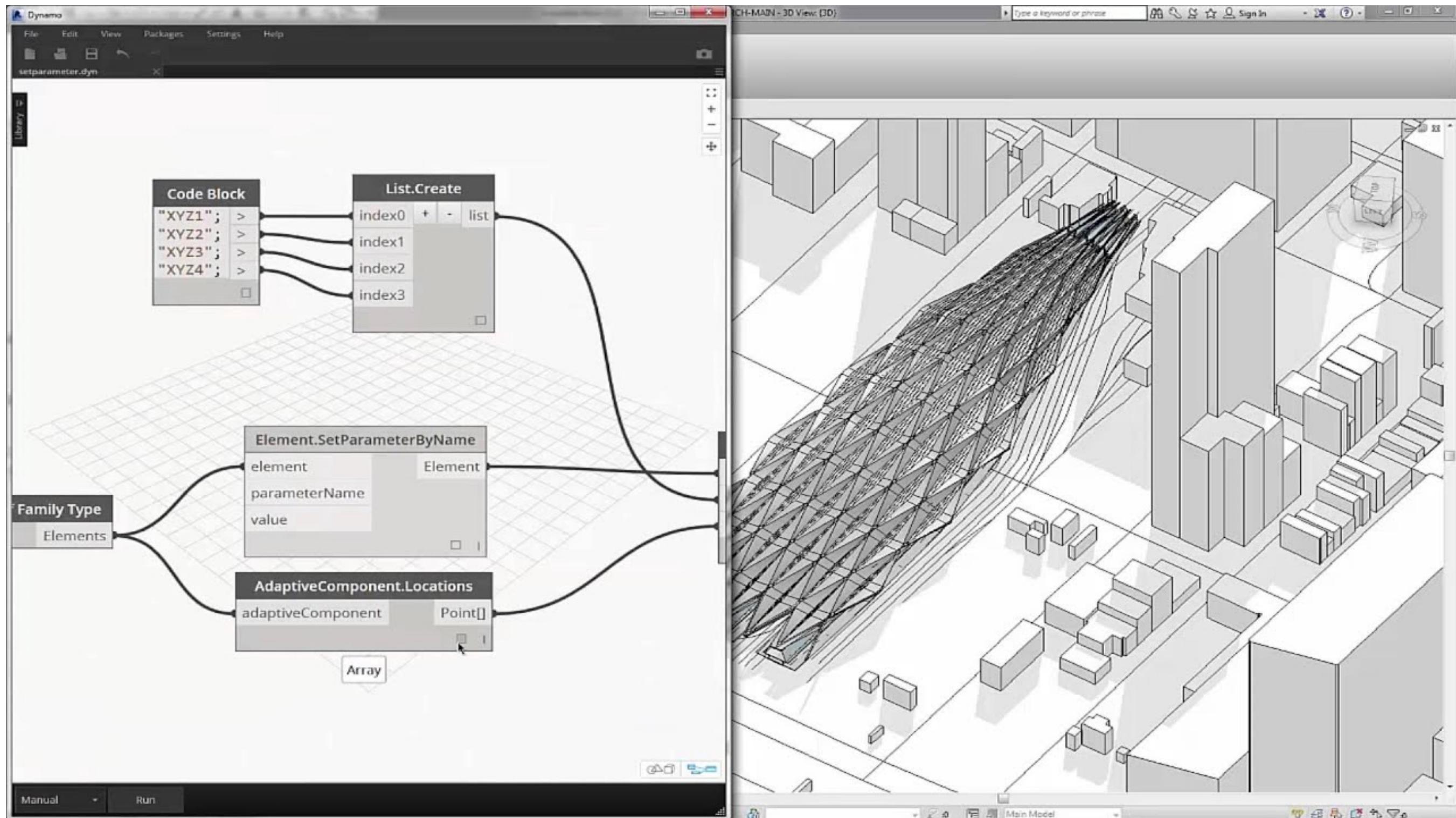
```
1 # Enable Python support and load DesignScript library
2 import clr
3 clr.AddReference('ProtoGeometry')
4 from Autodesk.DesignScript.Geometry import *
5 clr.AddReference('RevitNodes')
6 from Revit.Elements import *
7
8 famtype = IN[0]
9 pbc = Point.ByCoordinates(0,0,0)
10 output = []
11
12 for x in range(0, 100, 20):
13     for y in range(0, 100, 20):
14         for z in range(0, 100, 20):
15             pbc = Point.ByCoordinates(x,y,z)
16             col = FamilyInstance.ByPoint(famtype,pbc)
17             output.append(col)
18
19 OUT = output
```



DYNAMO



DYNAMO



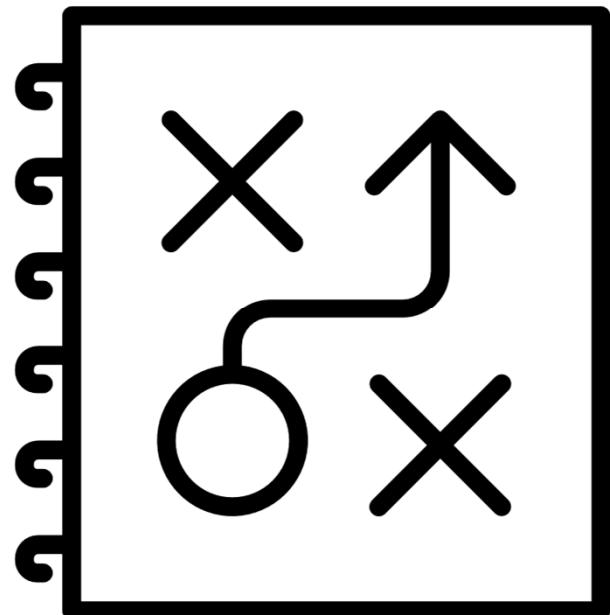
PROGRAMMING SHOULD BE FUN



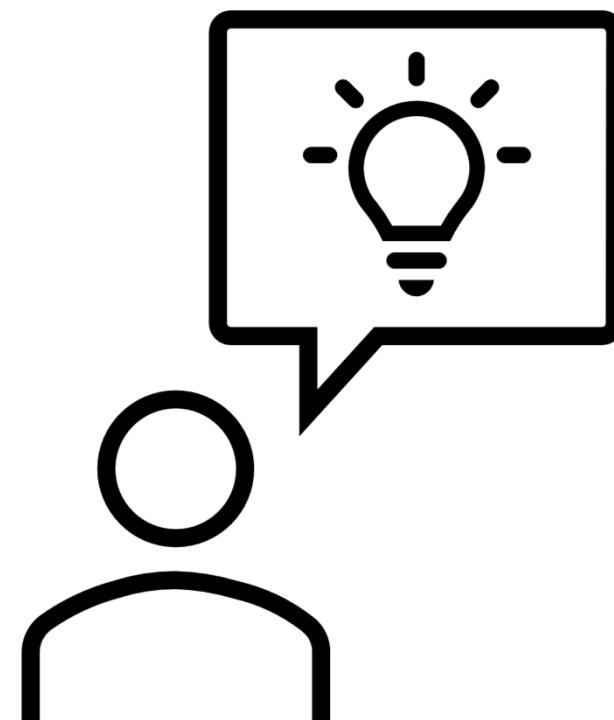
Python

WHAT YOU CAN DO

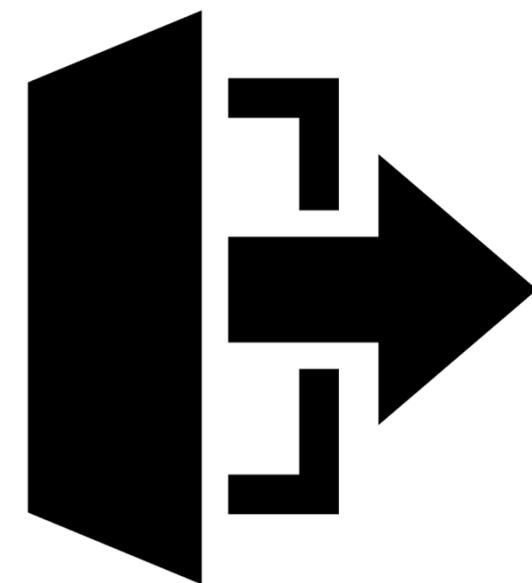
Use Diagrams in your documentation and training content



Be comfortable stepping out of text code and using graphics and other visuals to communicate code



Use and Contribute to open source projects that promote graphical code learning



RESOURCES

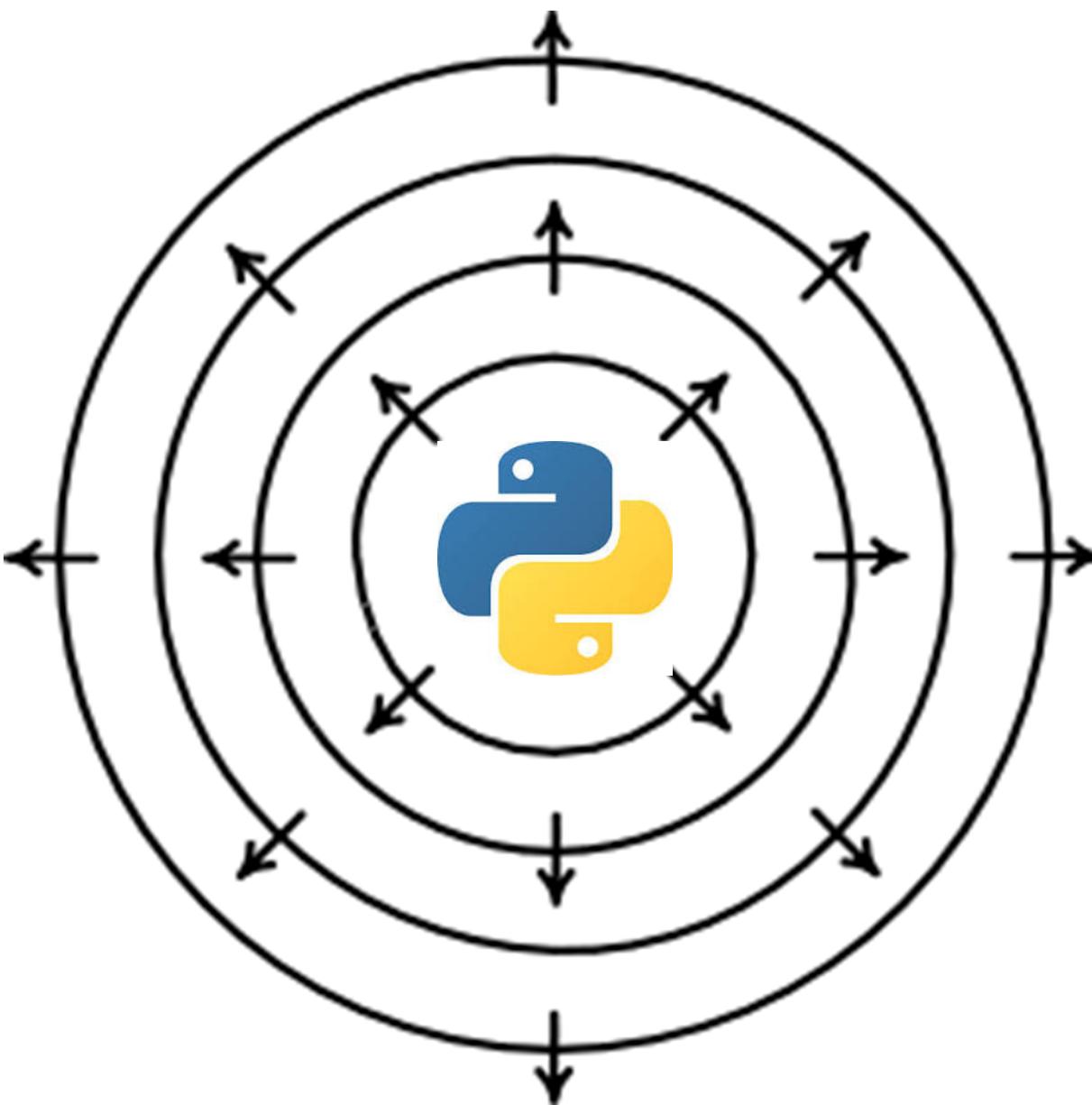
SCRATCH - https://en.scratch-wiki.info/wiki/Scratch_Source_Code

PYTHON TUTOR - <http://www.pythontutor.com/>

PYFLOW – <https://pyflow.readthedocs.io/en/latest/index.html>

DYNAMO - <https://dynamobim.org/>

EXPAND THE CIRCLE



Thanks everyone!



@tadeh_Hakopian
thakopian@gmail.com