

introduction.

Week 1 | Lecture 1 (1.1)

if nothing else, write `#cleancode`

This Week's Content

- **Lecture 1.1**
 - **Introduction**
- **Lecture 1.2**
 - Variables, Expressions, and Operators
 - Reading: Chapters 1, 2, 3
- **Lecture 1.3**
 - The Programming Process

Teaching Team



Ben
Instructor



Seb
Instructor



Joseph
TA



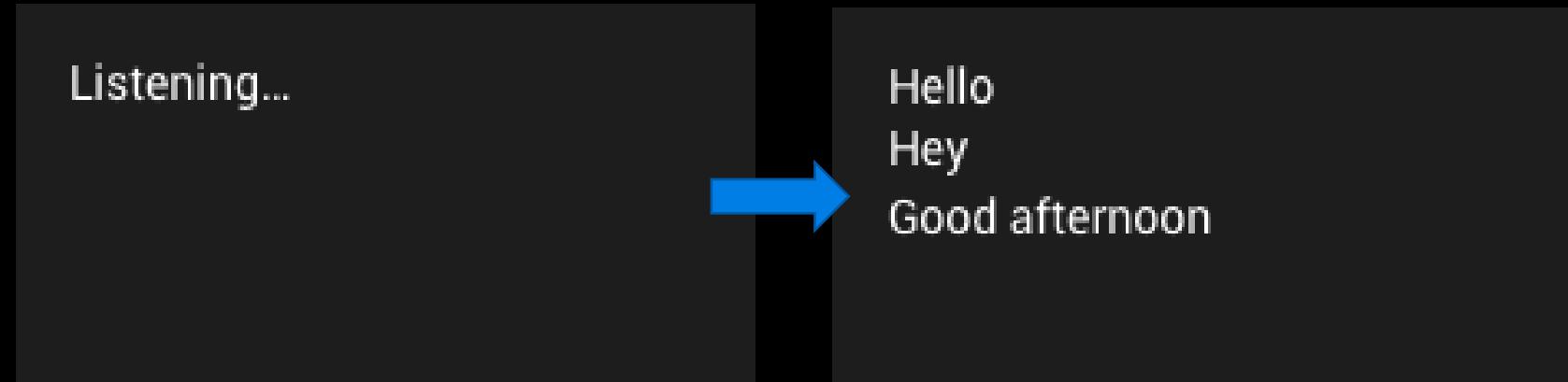
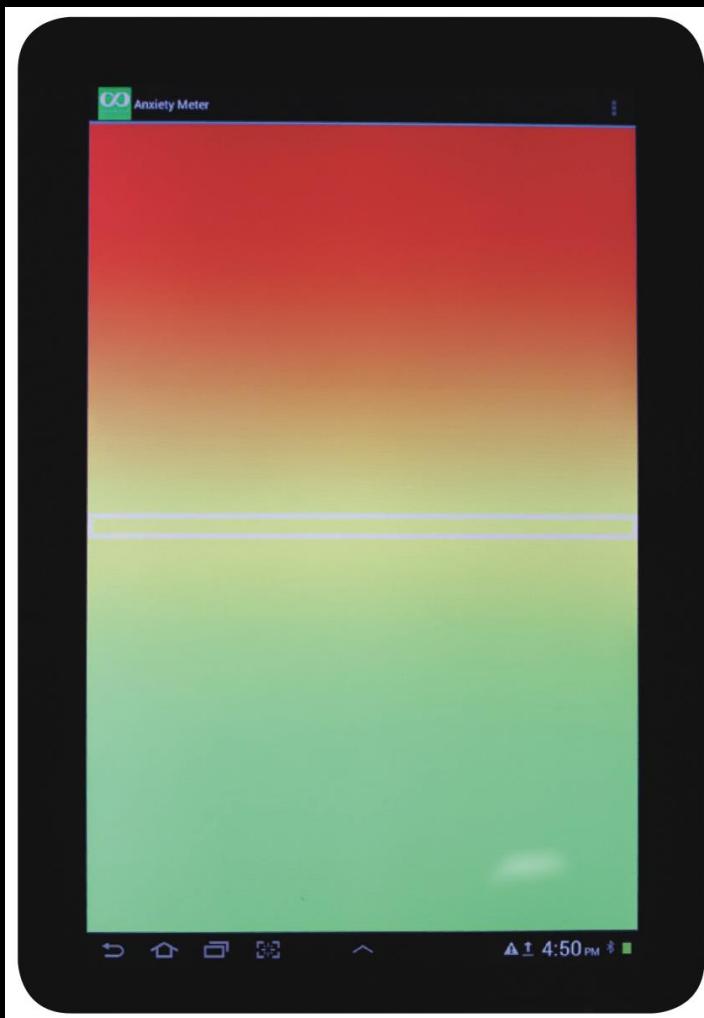
Katia
TA

Ben

- Bachelor of Electrical & Biomedical Engineering
@ McMaster University 2015
 - Realized the power of programming (and how it can be used for good, not evil)
- Master of Biomedical Engineering
@ University of Toronto 2020
 - Hold up... people will pay me for 'playing' on my computer?
- In between obtaining degrees I spent time programming a few different projects...

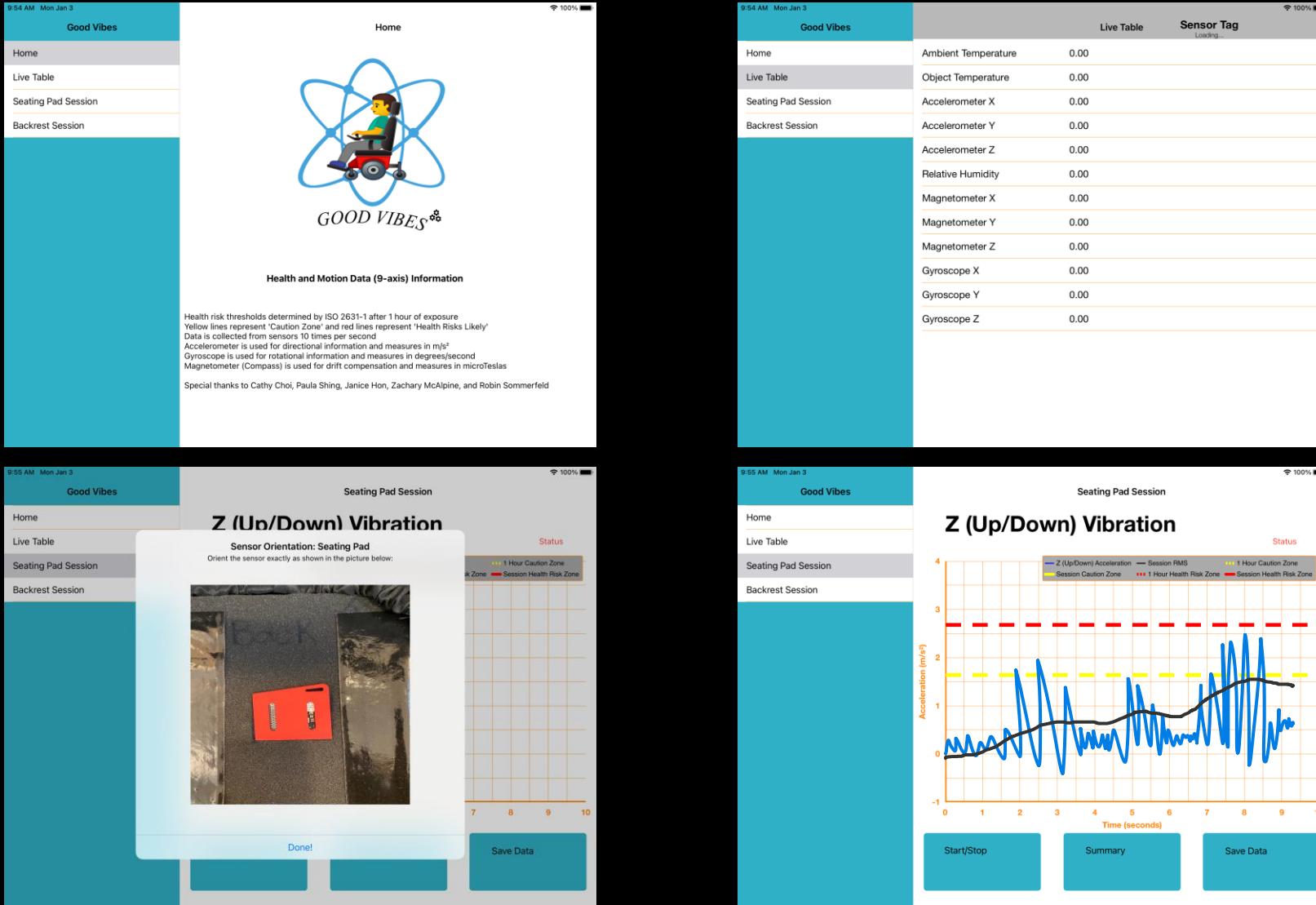


Projects for children with autism spectrum disorder



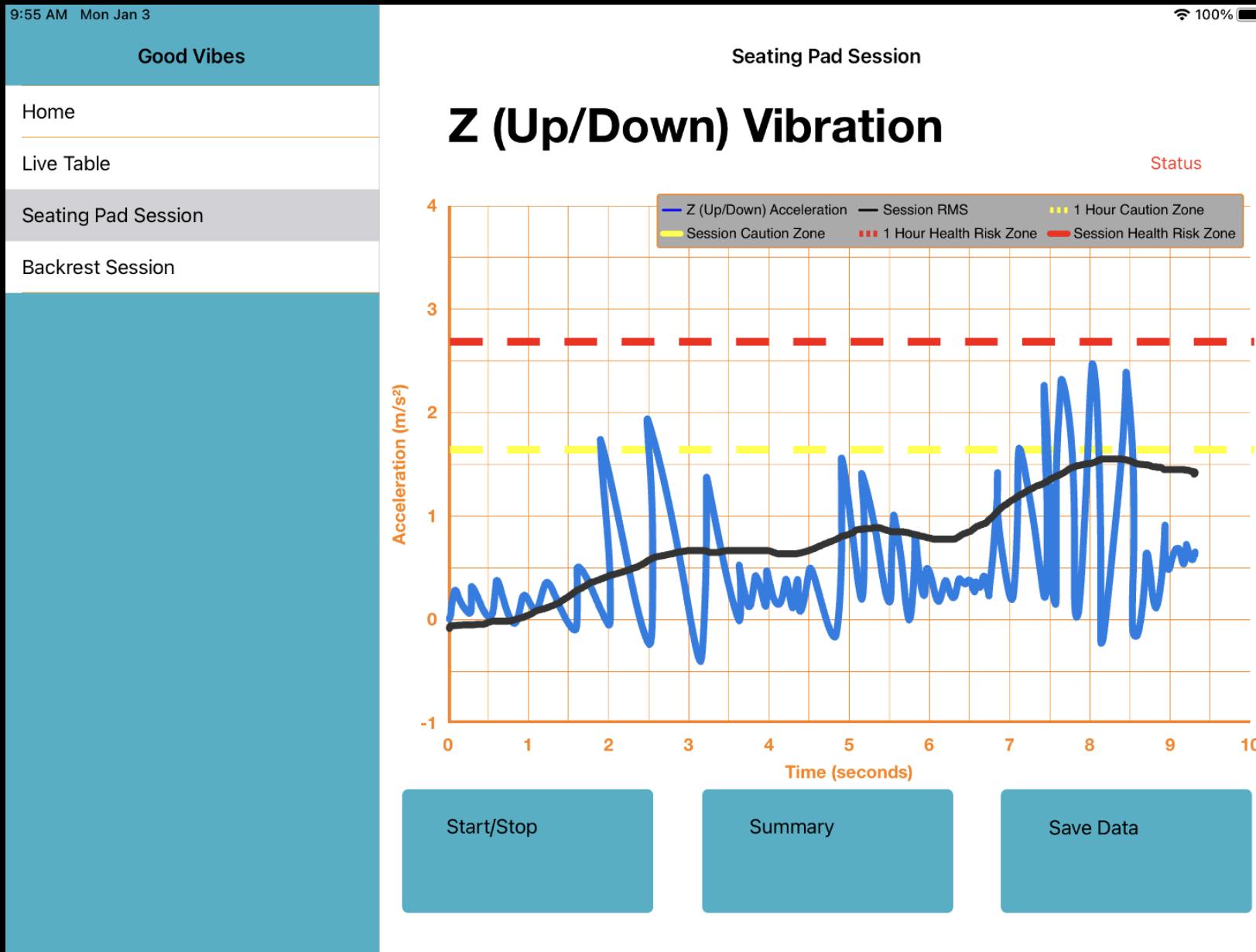
Good Vibes

A wheelchair vibration measurement tool



The image displays four screenshots of the Good Vibes mobile application, showing its user interface and data collection features.

- Home Screen:** Shows a sidebar with "Good Vibes", "Home", "Live Table", "Seating Pad Session", and "Backrest Session". The main area features a stylized atom icon with a person in a wheelchair, and the text "GOOD VIBES" below it. A note at the bottom explains the 9-axis motion data and sensor thresholds.
- Live Table Screen:** Shows a table with columns for "Good Vibes", "Live Table", and "Sensor Tag". The "Sensor Tag" column shows "Loading...". Data rows include "Ambient Temperature", "Object Temperature", "Accelerometer X", "Accelerometer Y", "Accelerometer Z", "Relative Humidity", "Magnetometer X", "Magnetometer Y", "Magnetometer Z", "Gyroscope X", "Gyroscope Y", and "Gyroscope Z", all with values of 0.00.
- Seating Pad Session Setup:** Shows a "Seating Pad Session" screen with a "Z (Up/Down) Vibration" section. It prompts to "Orient the sensor exactly as shown in the picture below:" and shows a photo of a red rectangular sensor placed on a dark surface. A legend indicates "1 Hour Caution Zone" (yellow dashed line) and "Session Health Risk Zone" (red dashed line). Buttons for "Done!" and "Save Data" are at the bottom.
- Seating Pad Session Data View:** Shows a graph of "Z (Up/Down) Vibration" over time (0 to 10 seconds). The y-axis is Acceleration (m/s²) ranging from -1 to 4. The x-axis is Time (seconds). The graph shows a blue line for "Z (Up/Down) Acceleration" with several sharp peaks, a black line for "Session RMS", and horizontal lines for "1 Hour Caution Zone" and "Session Health Risk Zone". Buttons for "Start/Stop", "Summary", and "Save Data" are at the bottom.



More about Ben...

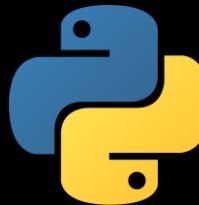


Seb

- Completed CivMin PhD in 2014 where I studies rock fracture and seismology (**RFDF**).
- AI Lead at KORE Geosystems, a mining tech startup.
- Senior Research Scientist at SickKids.  **SickKids®**
- Joined UofT CivMin in January 2020.
- Research topics: rock mechanics, ultrasonics, signal processing, computer vision, applied machine learning in mining.



AI 4 Mining



At KORE,
I use code
to do this!

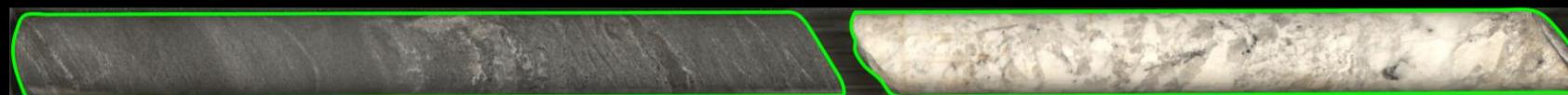


- Mining companies drill thousands of meters every year and the recovered rock (core) needs to be visually described by humans.

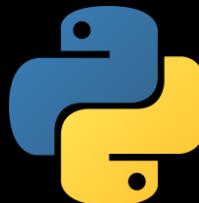
AI 4 Mining



At KORE,
I use code
to do this!



AI 4 Mining



At KORE,
I use code
to do this!



Alpha and Beta Angles

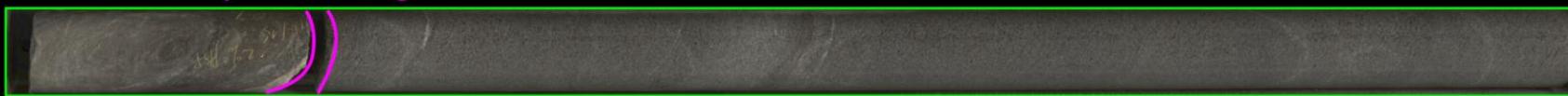


Alpha and Beta Angles

Alpha and Beta Angles



Alpha and Beta Angles



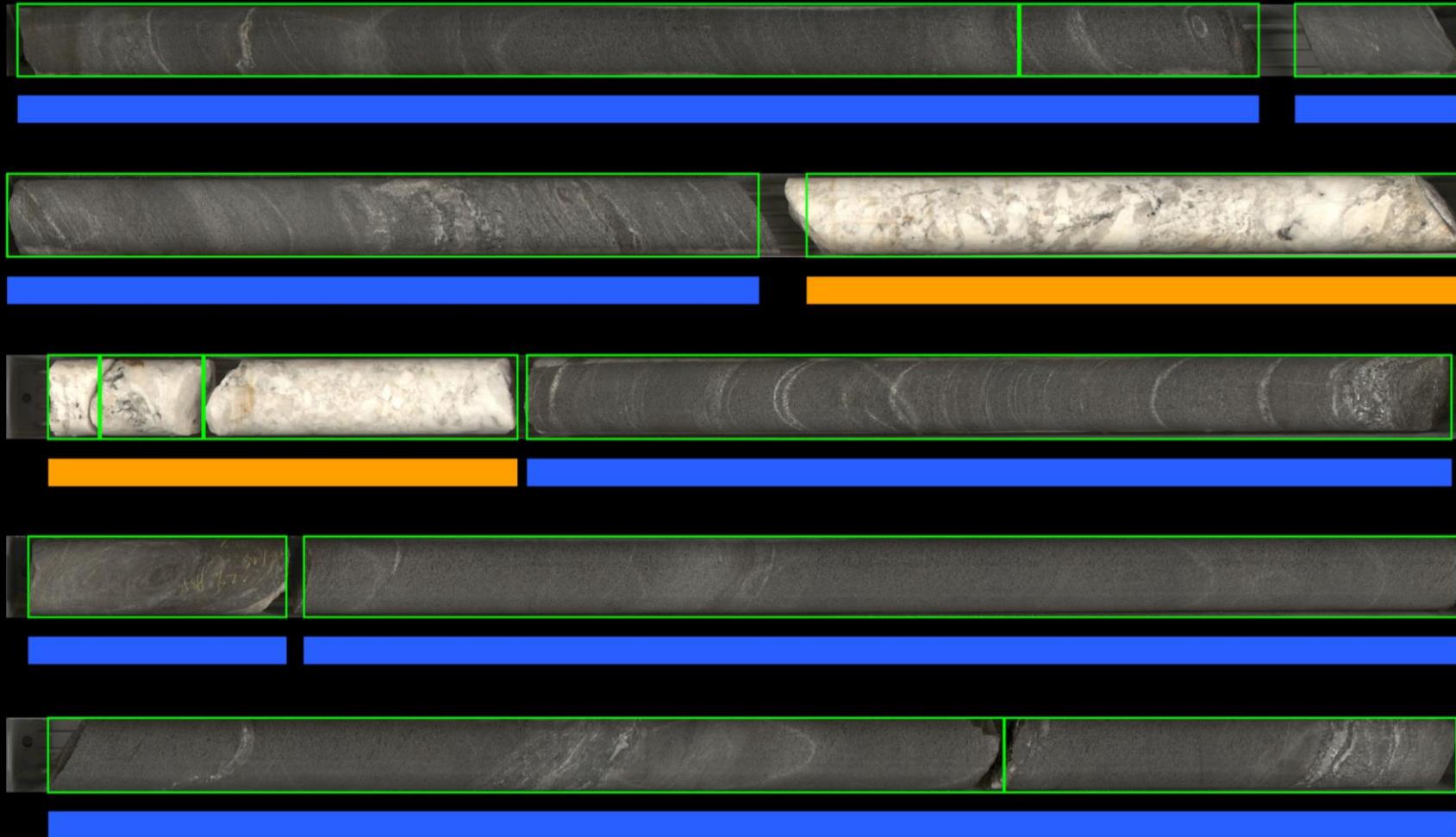
Alpha and Beta Angles



AI 4 Mining



At KORE,
I use code
to do this!



AI 4 Healthcare

AI Detects JET Pages Mjaye



Staff Physician



JET Onset

Nurse



5 minutes later

Treatment begins



At SickKids,
I use code
to do this!



Confirm 12-Lead



SickKids®

AI  JET: 5 %

Seb

- I love travelling and have visited over 20 countries including France, Switzerland, Italy, Austria, Croatia, Costa Rica, Nicaragua, Panama, Peru, Bolivia, United States, Israel, Jordan, West Bank, Egypt, Turkey, Morocco, South Africa, and Namibia.
- I was a founding member of Ottawa's premiere Beatles cover band.
- I love camping, hiking, and surfing.
- I have a wonderful baby boy named Avery Goodfellow and baby girl name August.
- I love building things with code ( Github: Seb-Good).



Katia

- Doing a Master's in Civil Engineering.
- Worked as a software application engineer-in-training (EIT) for 3.5 years.
- Working with Python to analyze and clean data!
- Using Python to work with images from digital and optical microscopes of concrete and other building materials. As well, analyzing fracture behavior from Finite Discrete Element simulation models (FDEM).
- Martial arts fanatic :) involved in both Muay Thai and karate.
- Also play piano, working towards Royal Conservatory certification.



Joseph

- Doing a PhD in biomedical engineering.
- A marathon runner and currently training for a triathlon.
- Interweaves Python and MATLAB to analyze ultrasound wave propagation in artificial hearts made from stem cells through image and signal analysis.



In Person (hopefully ☹)

- Talking
- Cell Phones
- Eating
- Being Late for Lectures



In Person (hopefully ☹)

- Talking
- Cell Phones
- Eating
- Being Late for Lectures



Online

- Ask questions in the chat or raise your hand if you'd like to unmute yourself to ask a question.
- Answer other students' questions if you think you know the answer
- Never forget - if you have a question, there's always other people in the room wondering the same thing. They will be happy you asked it!

Course Information

- **Quercus:**

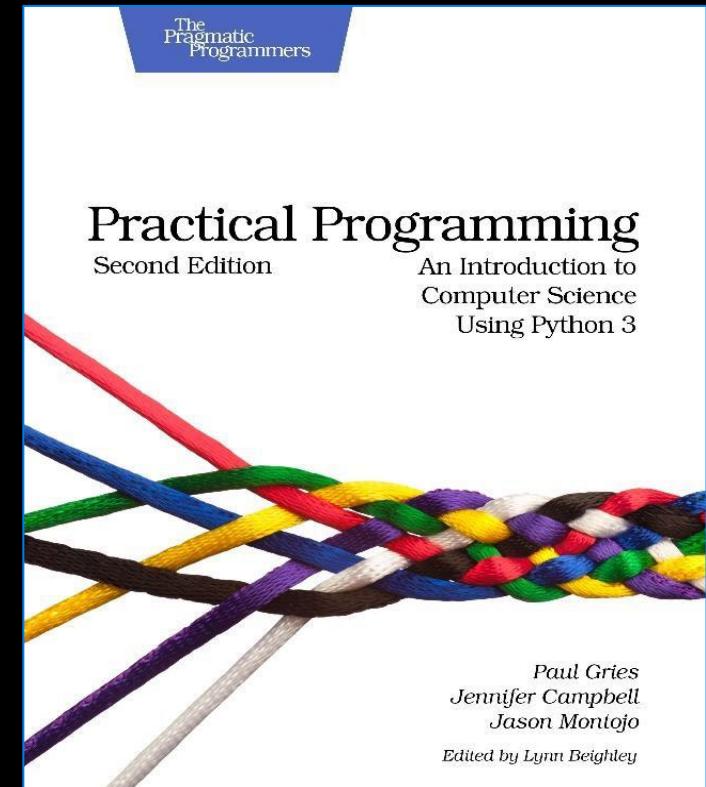
- APS106H1 S LEC0101 (One site contains all information).

- **Textbook:**

- ***Practical Programming: An Introduction to Computer Science Using Python 3***, 2nd edition, Paul Gries, Jennifer Campbell, Jason Montojo, Pragmatic Programmers.

- **Piazza:**

- piazza.com/utoronto.ca/winter2019/aps106



Print \$38 (e-book \$25)

Grading

Petitions: First-year Office.

Reflections (12 in total)

total: 2%

Labs (9 in total)

total: 10%

Term Test 1 (Feb 9)

total: 15%

Term Test 2 (March 14)

total: 23%

Final Exam

total: 50%

Lecture Schedule

Section	Monday	Tuesday	Wednesday	Thursday	Friday
LEC0101		12:00-13:00		12:00-13:00	14:00-15:00

- Lectures will be in person in **MYHAL 150**.
- Lecture content can be found on Quercus using the [Kinsella & Goodfellow](#) links.

Tutorial Schedule

One online tutorial TDB.

Week	Tut Content	Monday	Tuesday	Wednesday	Thursday	Friday	
Week 1 (Jan 9 - Jan 13)	No Tutorial						
Week 2 (Jan 16 - Jan 20)	Tutorial 1	TUT0103 15:00-16:00 WB 130 TUT0104 15:00-16:00 MY 380 TUT0105 12:00-13:00 MY 360 TUT0106 13:00-14:00 MY 330 TUT0108 15:00-16:00 GB 303	TUT0107 11:00-12:00 SF 2202	TUT0101 15:00-16:00 SF 2202			TUT0102 15:00-16:00 WB 130

Lab Schedule

Two online labs TDB.

Week	Lab Content	Monday	Tuesday	Wednesday	Thursday	Friday
Week 2 (Jan 16 - Jan 20)	Lab 1  Click here for content.				Lab 1 Released 18:00	PR0101 10:00-12:00 GB 144 PR0104 15:00-17:00 WB 255
Week 3 (Jan 23 - Jan 27)		PR0106 09:00-11:00 MY 030 PR0107 12:00-14:00 MY 030 PR0108 09:00-11:00 GB 144	PR0102 10:00-12:00 MY 030 PR0103 15:00-17:00 GB 144	PR0105 09:00-11:00 GB 144	Lab 2 Released 18:00	PR0101 10:00-12:00 GB 144 PR0104 15:00-17:00 WB 255 Lab 1 Due 23:00

Reflections

- There are 12 reflections throughout the term.
- Reflections are worth a total of 2% of your final grade.
- Full marks for successful completion of all reflections.
- Reflections are used to take the pulse of students in the course.
- Insights from reflections are used to iteratively improve the course week-by-week.
- Reflections are there to help you, so please answer honestly.

Reflections

- Reflection include question like:
 - Did you attend lecture this week?
 - Do you understand the topics covered this week?
 - Did you attend tutorial this week?
 - Which lecture section did you attend?
 - What will this code output?
 - Write one word to describe how you feel about the course.

Reflections Schedule

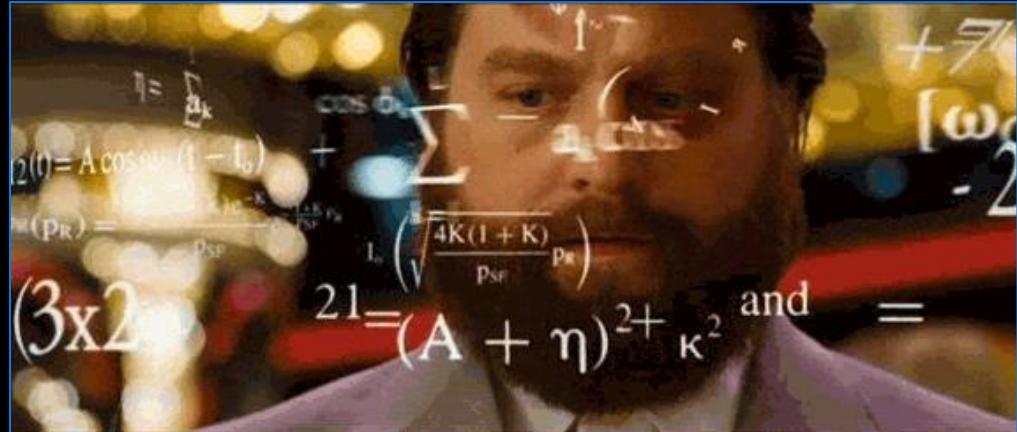
Week	Reflection	Monday	Tuesday	Wednesday	Thursday	Friday
Week 1 (Jan 9 - Jan 13)	Reflection 1				Reflection 1 Released 5:00 pm	
Week 3 (Jan 16 - Jan 20)	Reflection 2	Reflection 1 Due 11:00 am			Reflection 2 Released 5:00 pm	
Week 3 (Jan 16 - Jan 20)	Reflection 3	Reflection 2 Due 11:00 am			Reflection 3 Released 5:00 pm	
Week 3 (Jan 16 - Jan 20)	Reflection 4	Reflection 3 Due 11:00 am			Reflection 4 Released 5:00 pm	

Click here
for content.



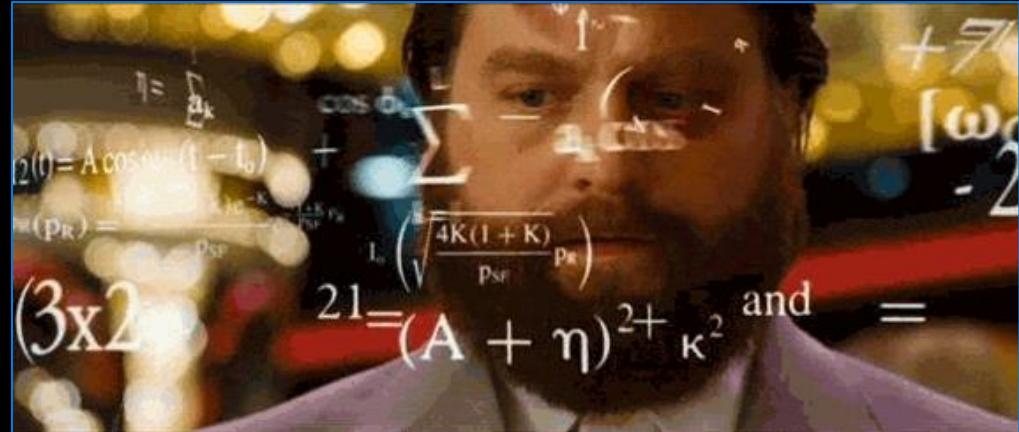
Tips for Success

- Do not wait until the last minute.
- Work Together (except during the test/exam):
 - Synthesis
 - Application
 - Evaluation
 - Explanation
- Work Smart (effort doesn't necessarily equal progress).
- Time Management.
- Know when (and how) to ask for help.



Tips for Success

- **88%** of your final grade is based on two term tests and an exam.
- For these tests and exams, you'll be to write code.
- You will not be able to achieve a high mark by reading the textbook and lecture note. You need to **PRACTICE, PRACTICE, PRACTICE!**
- How can you practice writing code?
 - **Lecture Breakout Sessions.**
 - **Labs (this of this as studying).**
 - **Practice Problems.**



Writing A Lot Of Code During The Term

=

Higher Grade

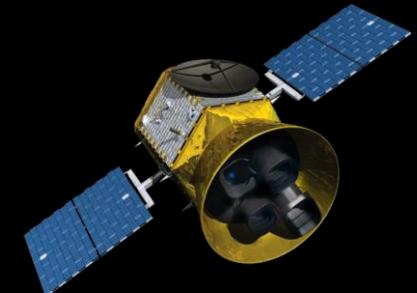
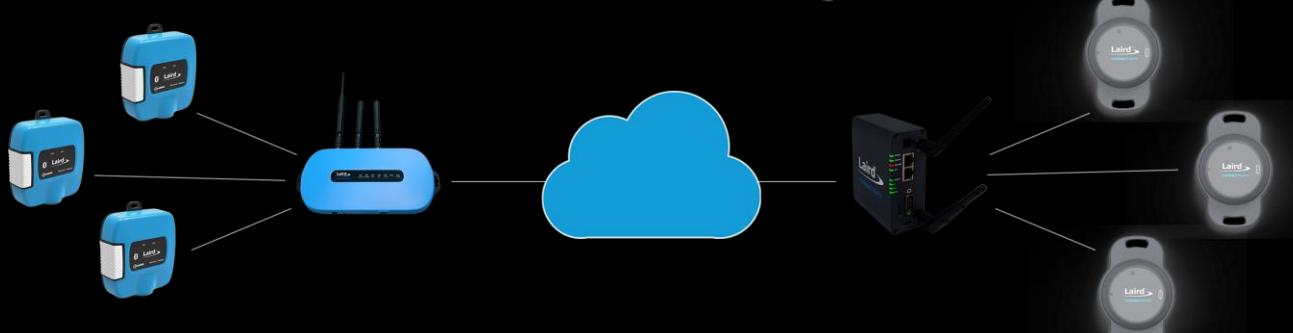
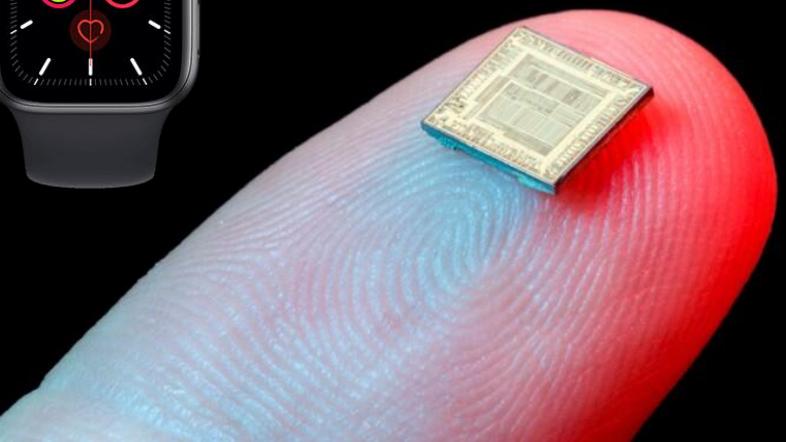
```
31     def __init__(self, path=None, debug=False):
32         self.file = None
33         self.fingerprints = set()
34         self.logduplicates = True
35         self.debug = debug
36         self.logger = logging.getLogger(__name__)
37         if path:
38             self.file = open(os.path.join(path, 'fingerprint.log'), 'a')
39             self.file.seek(0)
40             self.fingerprints.update([line.strip() for line in self.file])
41
42     @classmethod
43     def from_settings(cls, settings):
44         debug = settings.getbool('SUPERVISOR_DEBUG')
45         return cls(job_dir(settings), debug)
46
47     def request_seen(self, request):
48         fp = self.request_fingerprint(request)
49         if fp in self.fingerprints:
50             return True
51         self.fingerprints.add(fp)
52         if self.file:
53             self.file.write(fp + os.linesep)
54
55     def request_fingerprint(self, request):
56         return request_fingerprint(request)
```

What is this course all about?

- Learn the **fundamentals** of programming.
- **Programming as a tool** used in engineering.
- **Engineering design** through programming.

Why Programming?

StartUp **HERE**



Why Programming?

- Computers are everywhere:
 - Learn to interact with them.
 - Simplifies our jobs.
- Entrepreneurial opportunities (\$\$\$).
- Necessary for engineering jobs.
- Research and development.
- Because it's fun!



 **KORE**
GEOSYSTEMS

\$1,000,000 Investment.

Owner of **McEwen Mining Inc.**

CEO of **Franco-Nevada Corporation**

President of **Cisco Systems Canada**

COO of **Goldcorp**

SHARK on ABC's **Shark Tank**



What is Programming?

- A way of telling a computer what to do.
- A computer can't infer (...yet).
 - Need to tell a computer every single step it needs to do in a language it can understand.
 - How would you request an egg for breakfast to a chef and to a computer/robot?

- **To a Chef**

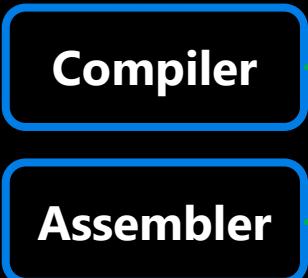
1. Sunny-side up, please!

- **To a Computer**

1. "Turn on stove"
2. "Take out pan"
3. "Take one egg out of fridge"
4. "Crack egg"
5. "Pour egg into pan"
6. "Wait 5 minutes"

How to Program a Computer.

Programmer



English

Pseudocode

Programming Language

Assembly Code

Machine Code



 Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.

 Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.



```
public class HelloWorld
{
    public static void main(String[ ] args)
    {
        system.out.println("Hello World!");
    }
}
```



```
print("Hello World!")
```

Lower-Level

Higher-Level



Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.

Increasing
Program Size

Matlab, SQL

C++, Java, Python

C, Fortran, Pascal

Assembly Code

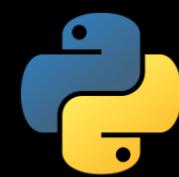
Machine Code



Increasing
Ease-of-Use

 Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.

 python

```
if x > 10:  
    print("x is greater than 10")
```

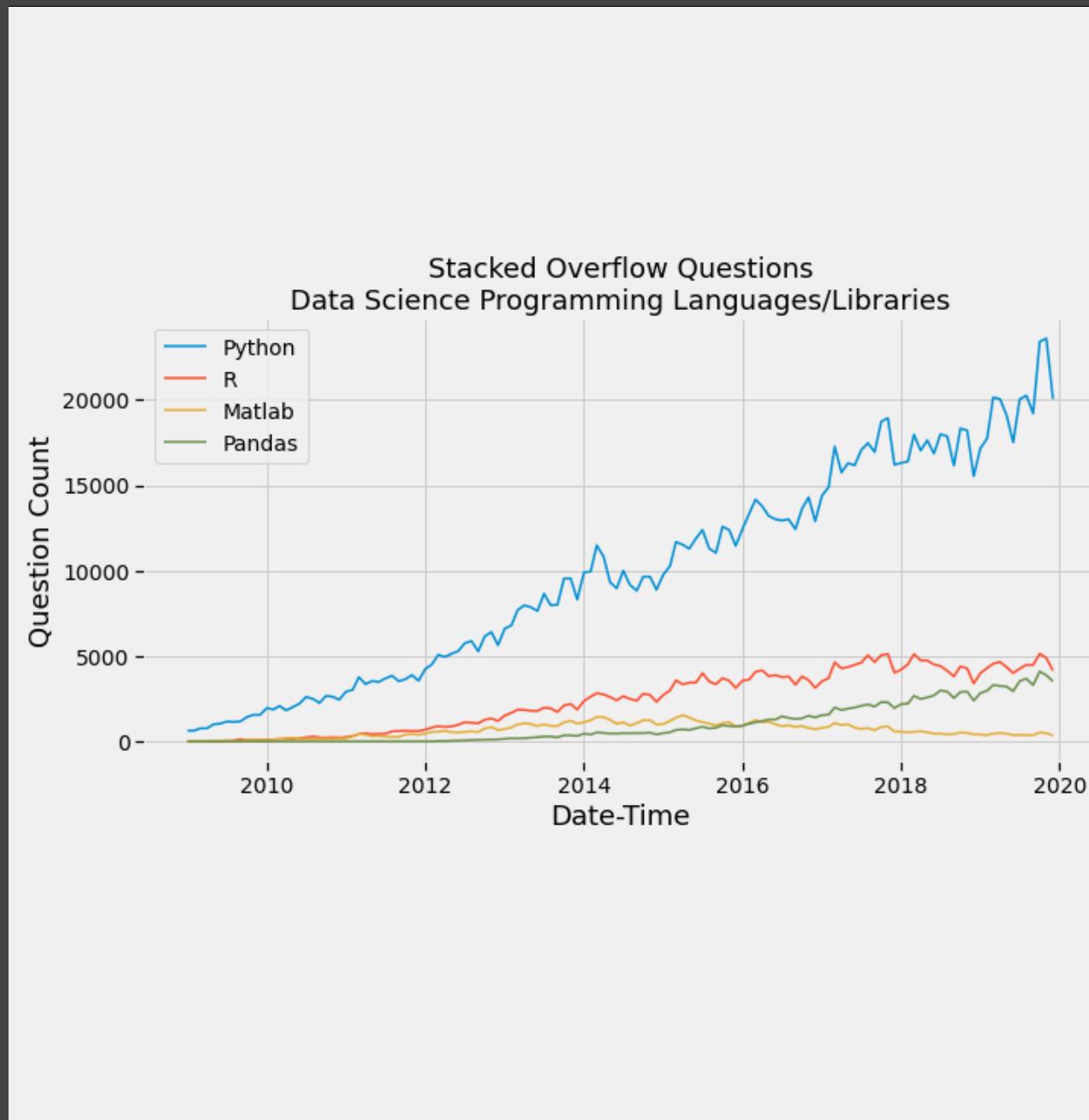


You don't know Python.
Can you figure out what this code does?



Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.



 Why Python?

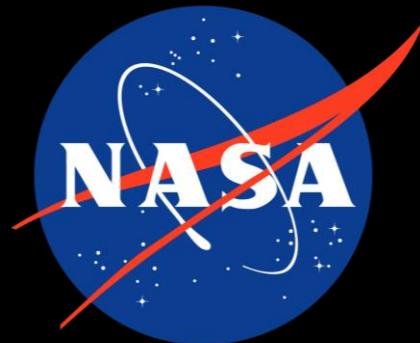
- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.

SPACEX

yahoo!



Google

KORE
GEOSYSTEMS

Spotify®



TESLA

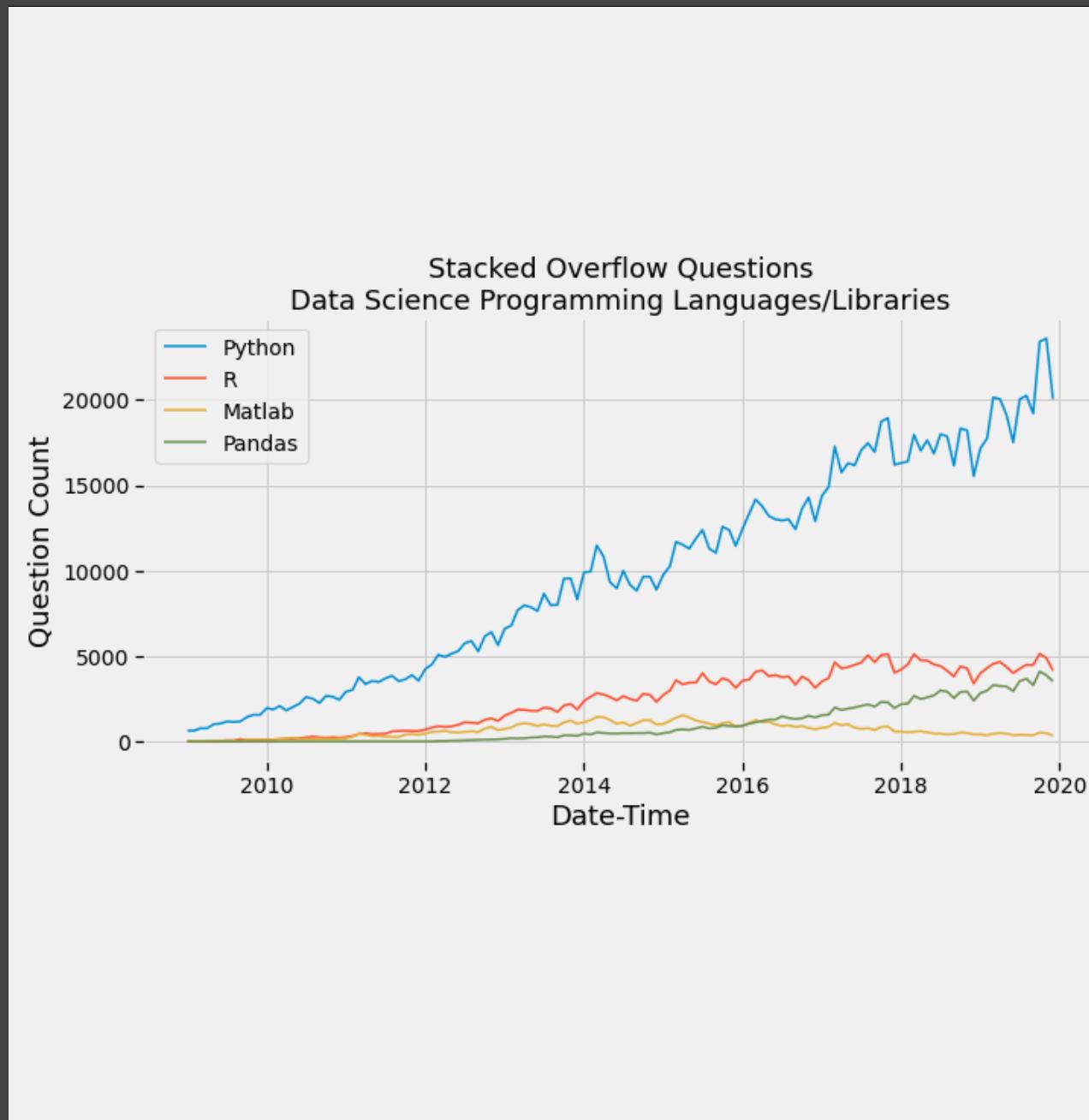
 Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.

 PyTorch LANDING AI Azure
Machine Learning XGBoost TensorFlow scikit
learn DeepMind OpenAI

 Why Python?

- High-level language making it more like a readable, human language than other low-level languages.
- Simple and clear syntax.
- Large open-source community.
- Used by Google, Firefox, Dropbox, Youtube, Instagram, Yelp, NASA, CIA, etc.
- The Artificial Intelligence (AI) community overwhelmingly uses Python.



Why Engineers?

- Ok, so we've explained why coding is important and why we're learning to code in Python.
- **BUT**, you're all engineers, right?
 - Mechanical
 - Materials
 - Civil
 - Chemical
 - Mineral
- Why do these kinds of engineers need to know how to code?



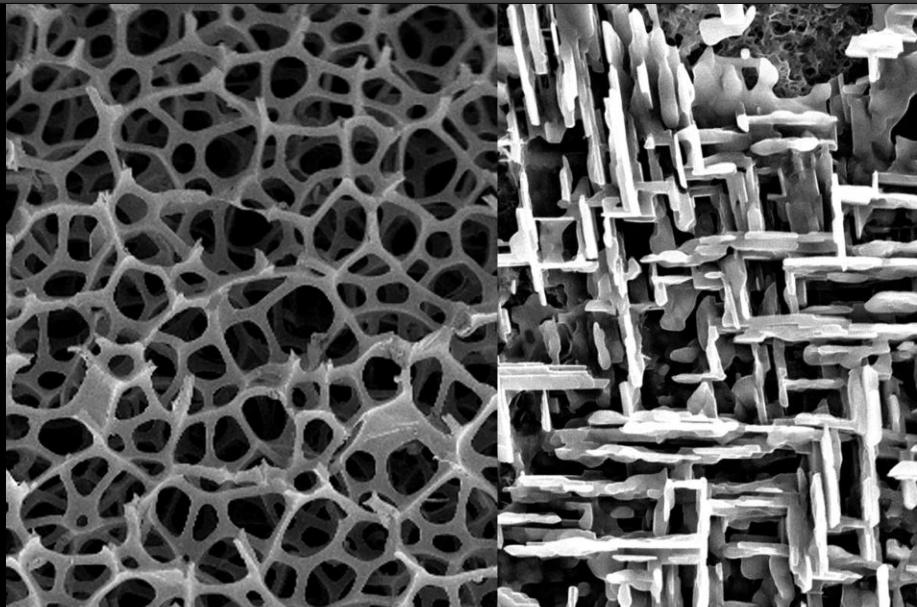
Mechanical Engineers

- Design Airplanes
- Micro Fluid Systems
- Automotive Engines
- Hydraulic Dams



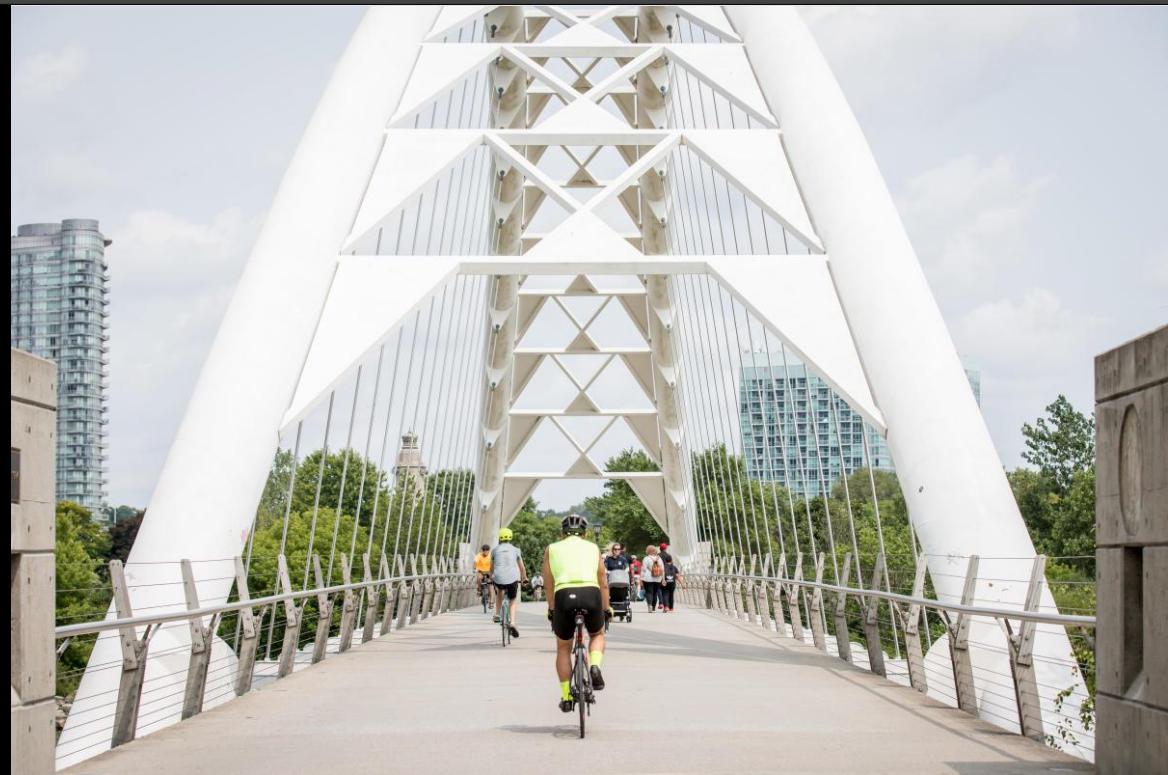
Materials Engineers

- Biomaterials
- Design of Materials
- Manufacturing with Materials
- Sustainable Materials Processing



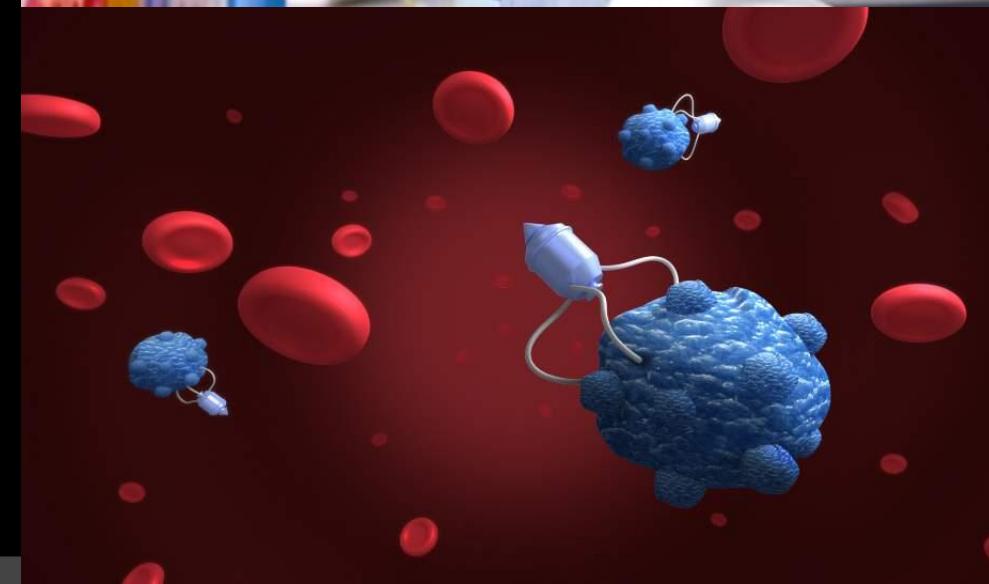
Civil Engineers

- Design Buildings
- Test Concrete
- Manage Water Supply System
- Design Bridges



Chemical Engineers

- Chemical Manufacturing Processes
- Petroleum Extraction
- Nanotechnology
- Manufacturing Computer Components



Mineral Engineers

- Mine Design
- Mineral Processing
- Mineral Exploration
- Mining Finance
- Mineral Extraction



Why Engineers?

- Again, why do engineers need to know how to code?
- In the 20th century, engineers didn't need to know how to code.
- In the 21th century, coding is an **essential skill** and any engineer without solid coding abilities will be at a major disadvantage.

```
31     self.file = None
32     self.fingerprints = set()
33     self.logduplicates = True
34     self.debug = debug
35     self.logger = logging.getLogger()
36
37     if path:
38         self.file = open(os.path.expanduser(path))
39         self.file.seek(0)
40         self.fingerprints.update(fingerprint for fingerprint in
41             self._read_file())
42
43     @classmethod
44     def from_settings(cls, settings):
45         debug = settings.getboolean('debug')
46         return cls(job_dir=settings['job_dir'],
47                    request_fingerprint=settings['request_fingerprint'],
48                    logduplicates=settings.getboolean('logduplicates'),
49                    file=settings.get('file'))
50
51     def request_seen(self, request):
52         fp = self.request_fingerprint(request)
53         if fp in self.fingerprints:
54             return True
55         self.fingerprints.add(fp)
56         if self.file:
57             self.file.write(fp + '\n')
58
59     def request_fingerprint(self, request):
60         return request_fingerprint(request)
```

Why Engineers?

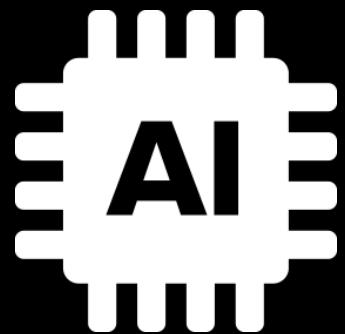
- Five technology trends have changed the game for engineers.



The Cloud



Cheap Sensors



Artificial
Intelligence



Big Data



The Internet
Of Things

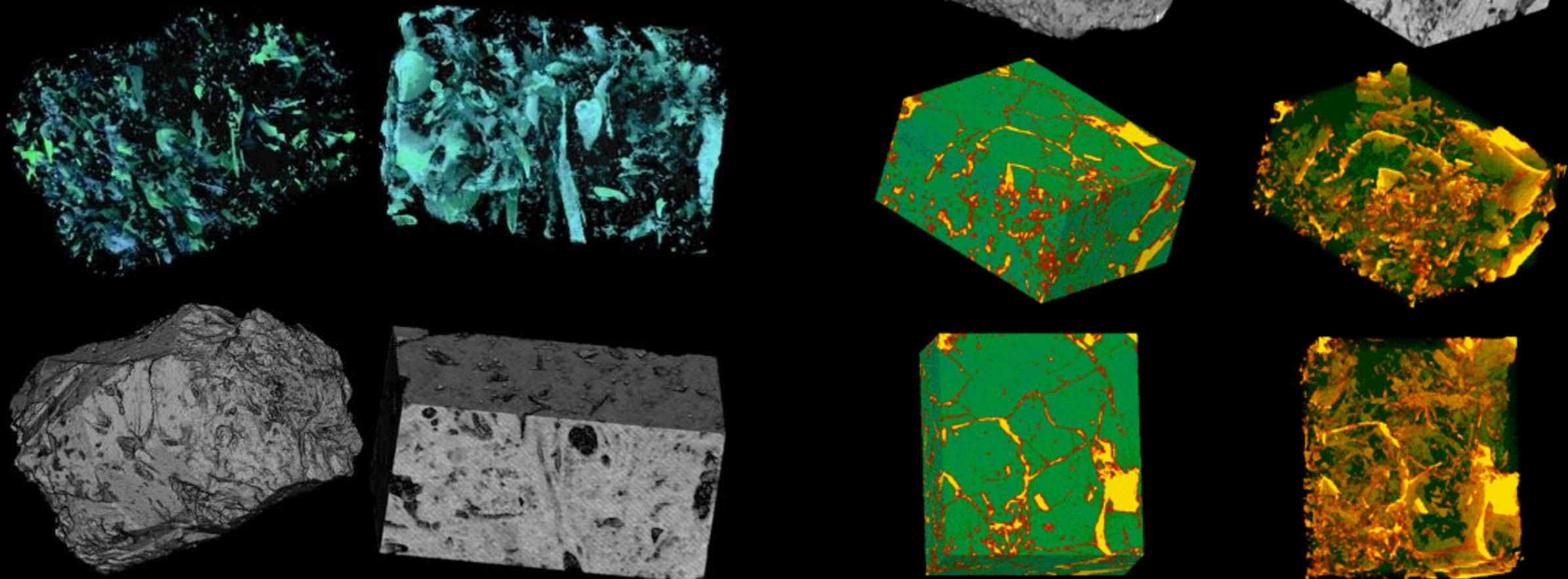
Mechanical Engineers

Realtime Diagnostics & Tuning



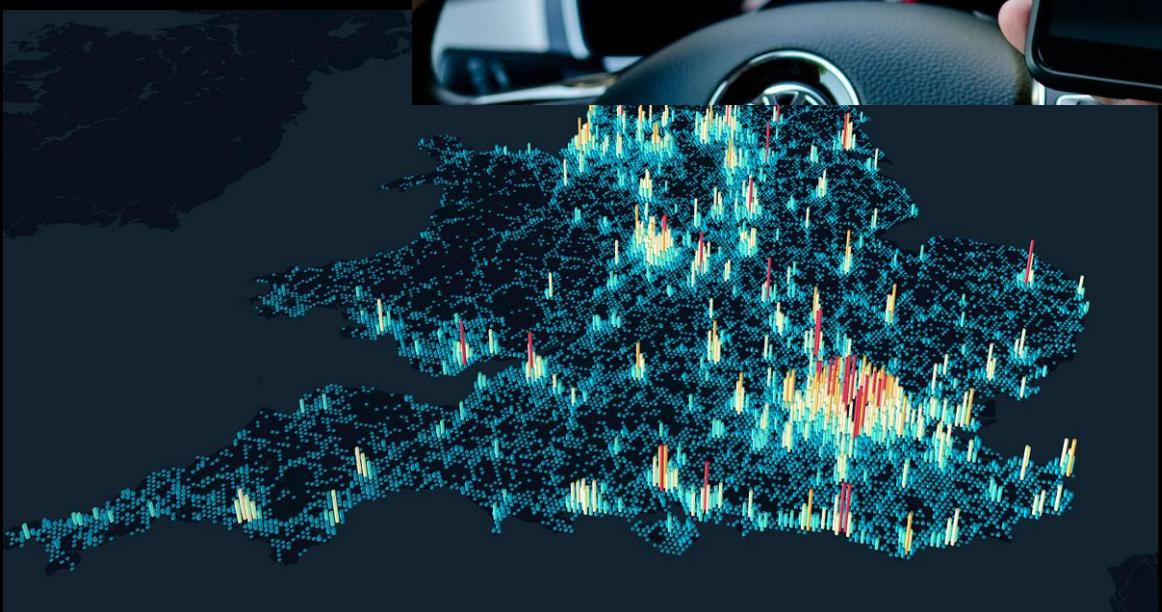
Materials Engineers

Digital Image Analysis of Material Microstructure



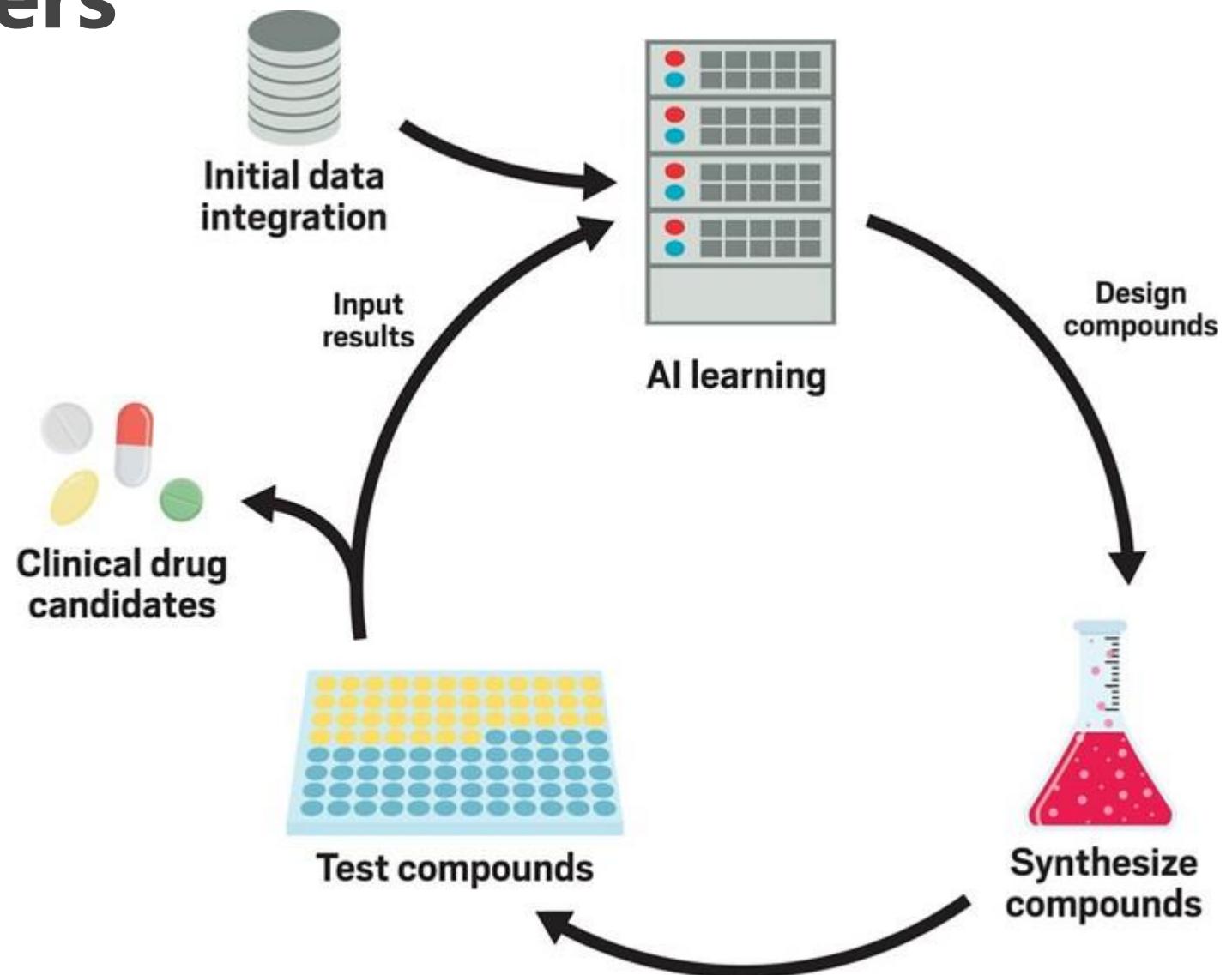
Civil Engineers

Urban Transportation Disrupted

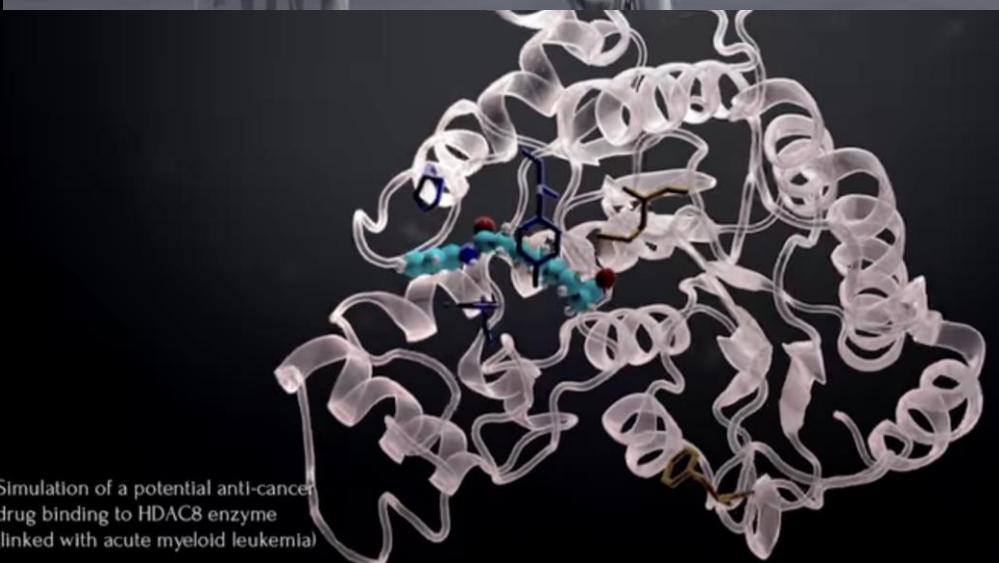
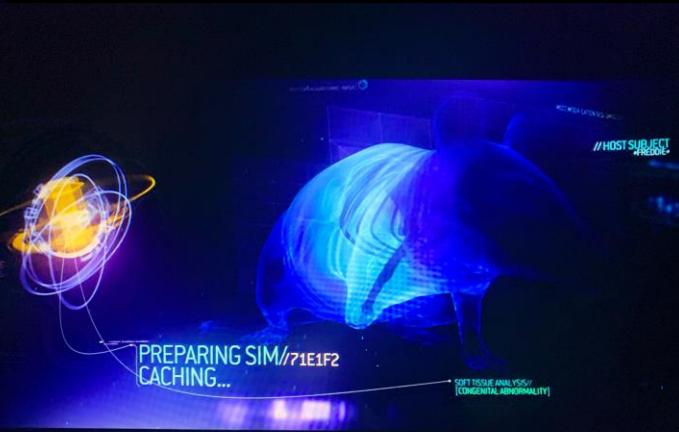


Chemical Engineers

Synthesizing new compounds



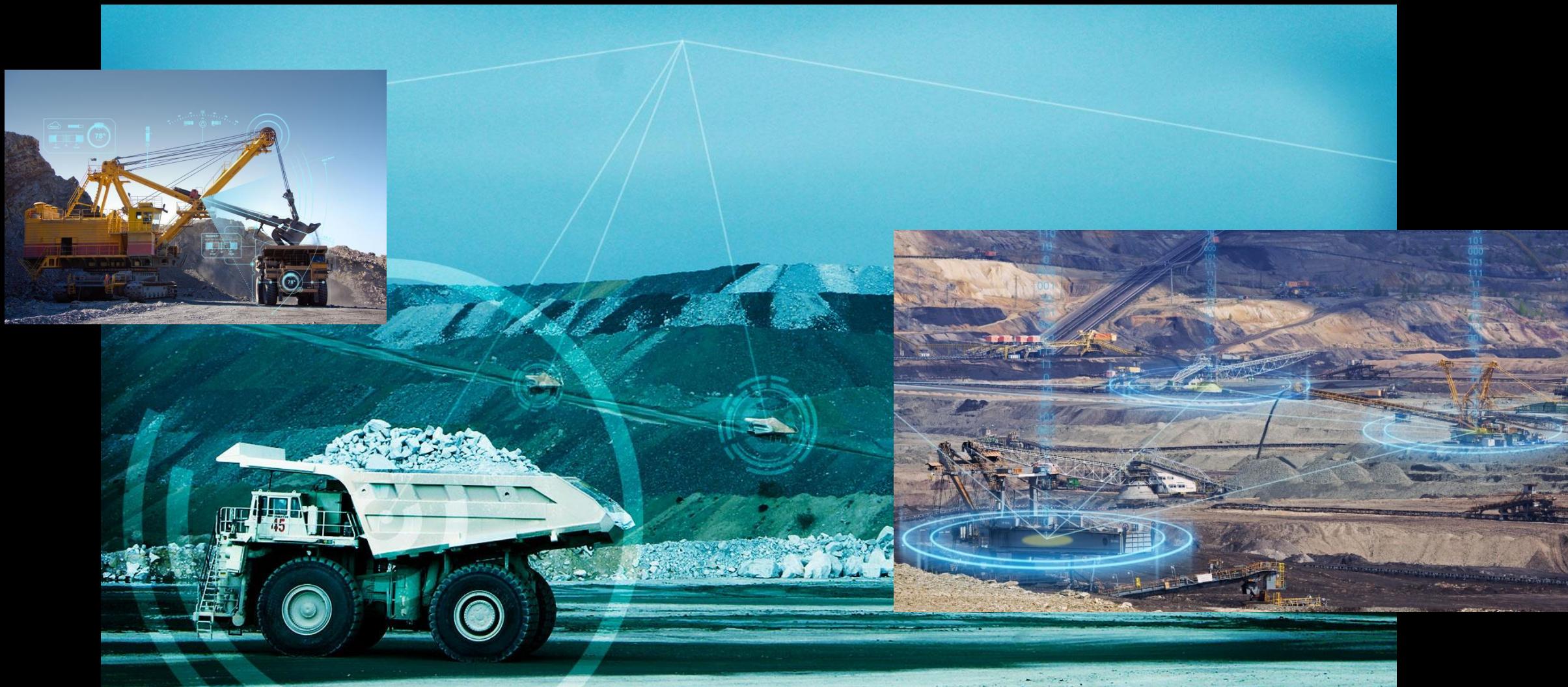
Sci-fi Inspiration



Simulation of a potential anti-cancer drug binding to HDAC8 enzyme (linked with acute myeloid leukemia)

Mineral Engineers

Semi-Autonomous Mines



Programming Skills Are In Demand

- According to the *Developer Survey by StackOverflow*, Python was one of the most in-demand technologies of 2018, 2019, 2020, and 2021.
- As of 2021, it is ranked as the world's 2nd most popular programming language among professional software developers as well as the first most wanted programming language.

```
31     def __init__(self, job_dir, settings):
32         self.job_dir = job_dir
33         self.settings = settings
34         self._init_file()
35
36     def _init_file(self):
37         self.file = None
38         self.fingerprints = set()
39         self.logduplicates = True
40         self.debug = False
41         self.logger = logging.getLogger(__name__)
42
43     @classmethod
44     def from_settings(cls, settings):
45         debug = settings.getboolean("debug")
46         return cls(job_dir=settings["job_dir"],
47                    settings=settings,
48                    debug=debug)
49
50     def request_seen(self, request):
51         fp = self.request_fingerprint(request)
52         if fp in self.fingerprints:
53             return True
54         self.fingerprints.add(fp)
55         if self.file:
56             self.file.write(fp + "\n")
57
58     def request_fingerprint(self, request):
59         return request_fingerprint(request)
```

One of the reasons for choosing Engineering!

- Finally! A course that isn't Chem, Bio, Physics, or Math
- This (as well as your engineering design course) is what separates you from many other sciences
 - And makes you **MUCH** more valuable....

Programmer



What my friends think I do



What my mom thinks I do



What society thinks I do



What my boss thinks I do



What I think I do



What I actually do

introduction.

Week 1 | Lecture 1 (1.1)

if nothing else, write `#cleancode`