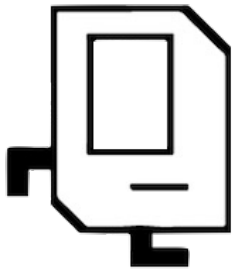
Lesson 1
Welcome to Code in Place

In this first video

- Meet the Team
- What is Code in Place?

After: Learn to Code!



Meet the team!

Mehran Sahami

- Childhood: Iran
- High School: San Diego
- Stanford University Ph.D. in Machine Learning
  (Before Machine Learning was cool)
- Spent a decade in tech industry before coming back as professor
  - Love of teaching is why I came back

Mehran Sahami

Piech and Sahami, Code in Place

Chris Piech

Teaching at Stanford

8,000+ students over 10 years

CS106A	CS106B	CS109	CS221
Programming Methodologies	Programming Abstractions	Probability for Computer Scientists	Intro to Artificial Intelligence
CURRENT	LAST: FALL 2016	LAST: FALL 2018	LAST: SUM 2013

Invented the algorithm behind bird brain

Lead a research lab in:
Future of Education

Grew up in Nairobi, Kuala Lumpur before Stanford!

Piech and Sahami, Code in Place

What An Awesome Team


Chris Piech


Mehran Sahami

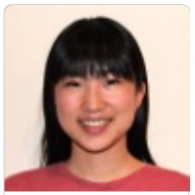

Ali Malik


Brahm Capoor


Juliette Woodrow


Julie Zelenski


Thomas Jefferson


Miranda Li


Patricia Wei


Cameron Mohne


Sierra Wang

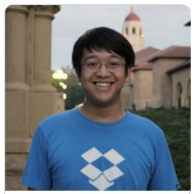

Hannah Cussen


Jennifer Xu


Joseph Tey


Julia Markel


Sarah Jade Yao


Allen Nie


Yunsung Kim

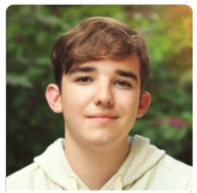

Dora Demszky


Jason Ford


Jim Malamut


Jennifer Langer-Osuna


Hannah Clay


Justin Blumencranz

It Takes a Village!

Aaron Broder, Abdu Mohamdy, Abdulwahab Omira, Adam Mosharrafa, Ahmet Uysal, Akinbowale Akin-Taylor, Akshaya Dinesh, Alan Cheng, Alejo, Alexandre Bucquet, Ali Malik, Amelie Byun, Ana Saavedra, Anand Shankar, Andrea Collins, Andrew Tierno, Areeba Khan, Arun Kulshreshtha, Ashley Taylor, Baker Sharp, Baris Bozkurt, Ben Barnett, Ben Holtz, Ben Ulmer, Cameron Mohne, Cameron Schaeffer, Caroline Tran, Cathy Zhang, Daniel Chao, Daniela Gonzalez, Dora Demszky, Elena Mosse, Eric Chew, Erin McCoy, Ethan Chi, Ethan Roy, Francisco Pernice, Frank Chen, Gonzalo Revuelta, Grace Hu, Griffin Dietz, Griffin Koontz, Guy Blanc, Ha Tran, Hannah Zhang, Haven Whitney, Heidi Chen, Inessa Roy, Irene Han, Isaac Pohl-Zaretsky, Jackson Eilers, Jacob Sharf, Jason Chen, Jason Ford, Jason Ma, Je-Mé Kruger-Baartjes, Jennie Yang, Jerry Chen, Jessie Duan, Jillian Tang, John Cho, Joseph Lee, Julia Chin, Julia Daniel, Julia Lee, Julia Truitt, Juliette Woodrow, Kaili Wang, Kartik Chandra, Kat Gregory, Kate Rydberg, Katherine Erdman, Kayla Patterson, Kevin Shin, Kurt Berglund, Kylie Jue, Lee Alpert, Lekan Wang, Lena Blackmon, Linda Tong, Lisa Einstein, Lisa Wang, Lisa Yan, Logan Marquis, Lucía M, Maneesh Apte, Matt King, Matt Peng, Maxwell Bigman, Maya Ziv, Michael Chang, Mike Wu, Miranda Li, Miroslav Suzara, Moussa Doumbouya, Nadin Tamer, Nancy Hoang, Natalie Cygan, Nathan Dalal, Neel Kishnani, Nick Bowman, Nick McKeown, Nikhil Raghuraman, Nishant, Parth Sarin, Patricia Wei, Rich Davis, Ryan Eberhardt, Ryan Lian, Sabri Eyuboglu, Sam King, Samantha Kim, Sasha Ronaghi, Sathya Edamadaka, Sauren Khosla, Scott Maxwell, Segev Tsfati, Sejal Jain, Shreya Shankar, Sierra Kaplan-Nelson, Sonja Johnson-Yu, Sophie Andrews, Stephanie Palocz, Swati Daub, Tague Griffith, Tara Balakrishnan, Tara Iyer, Toby Bell, Tori Qiu, Troy Shen, Tyler Yep, Tyrell Baker, Varun Tandon, Will Crichton, William Ellsworth, Yifei Men, Yosefa Gilon, Zachary Birnholz, Zack Cinquini, Zheng Lian

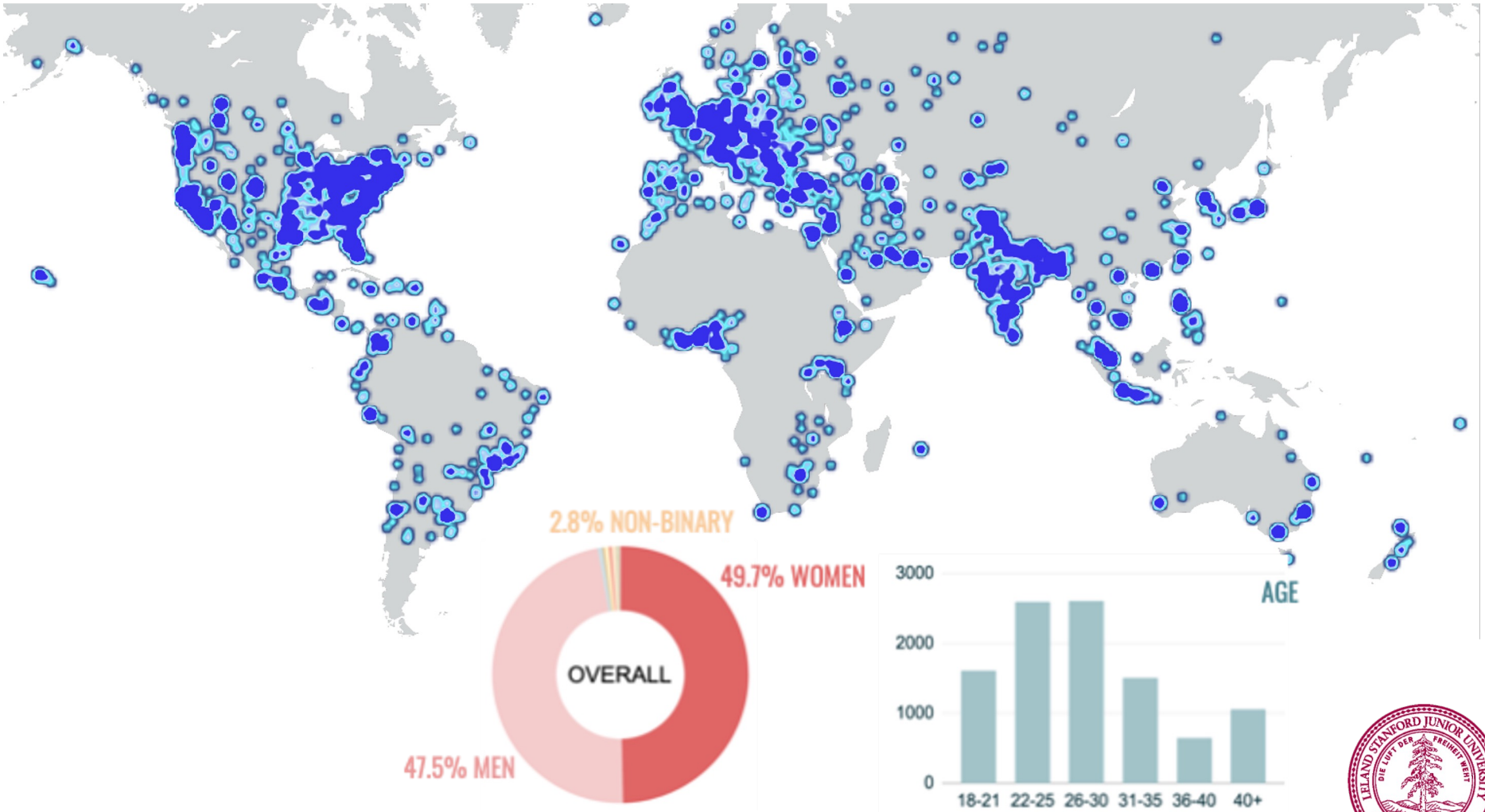
What is Code in Place?

Code in Place:

1/2 of CS106A
As Community Service

Learners from over 150 countries

and all walks of life have completed this course



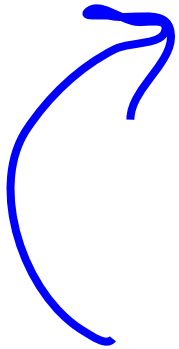
This class is just for **fun**

(what could be more fun than learning)

We are doing this as **volunteer** work. You get a class for free!

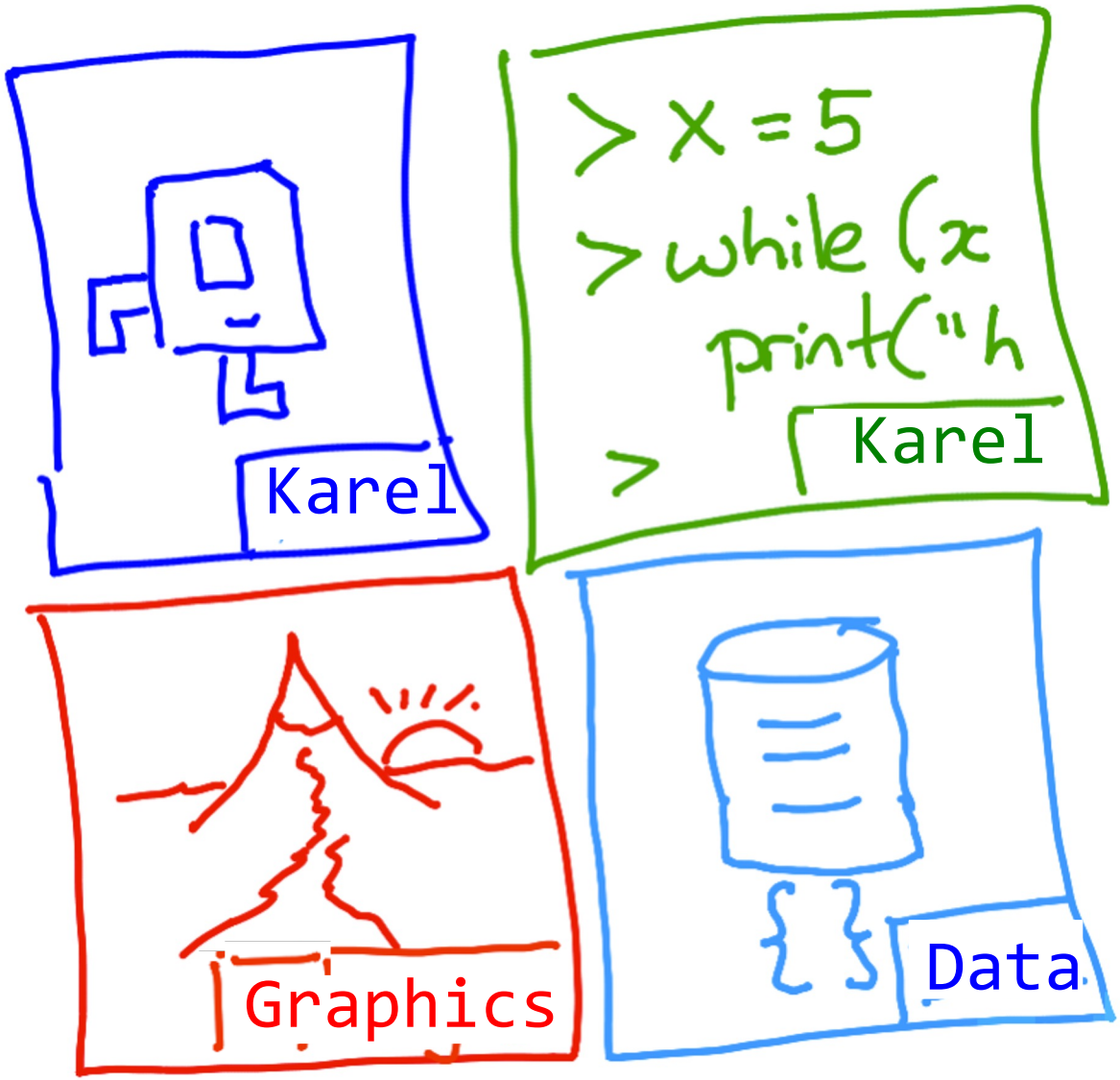
Learning Goal

Learn how to solve problems with python

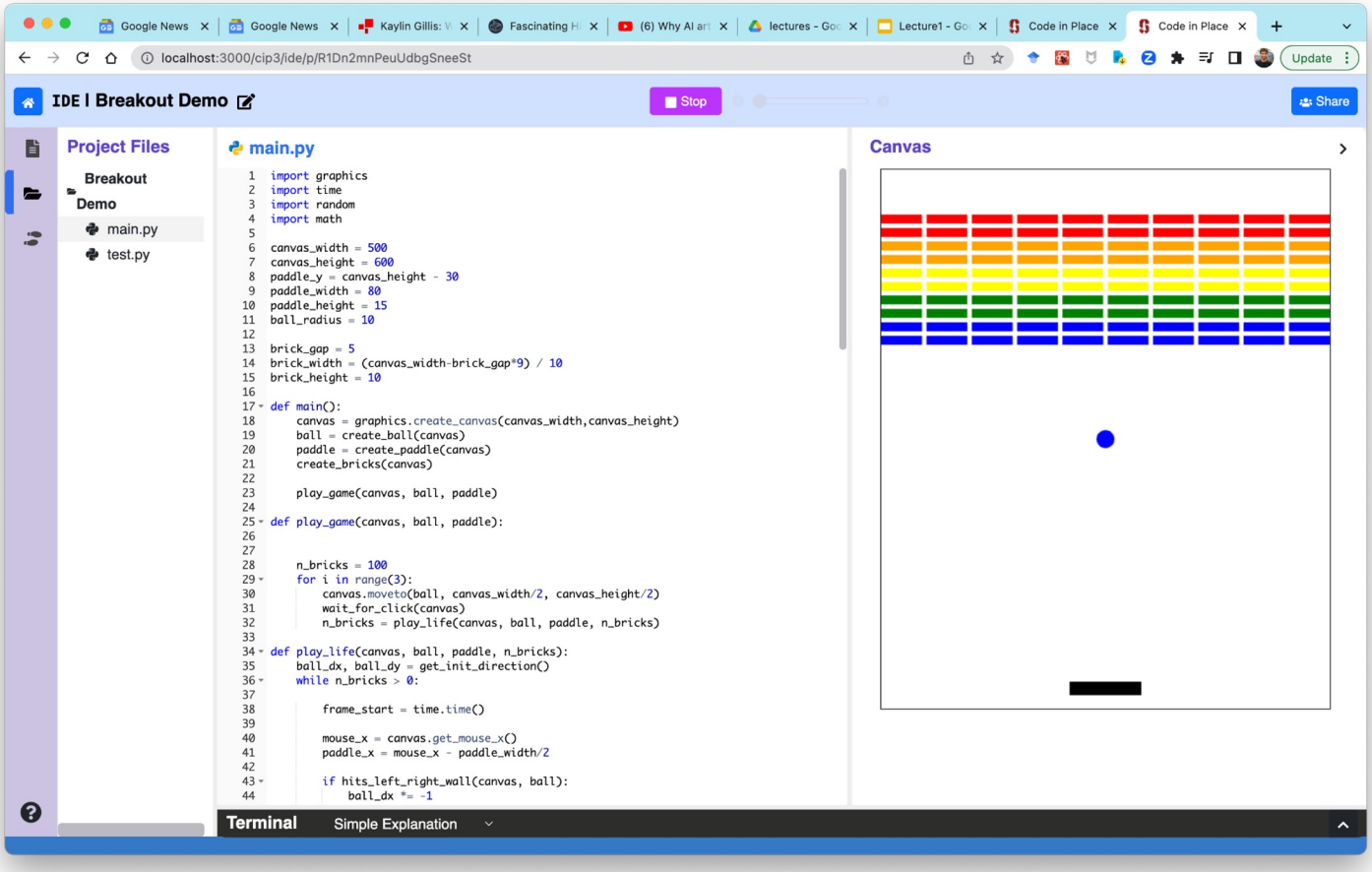


Find the joy of a new skill

Course Plan



Programming Along the Way



Piech and Sahami, Code in Place

Course Values

Art of Computer *Science*

Learn by Doing

Gratitude

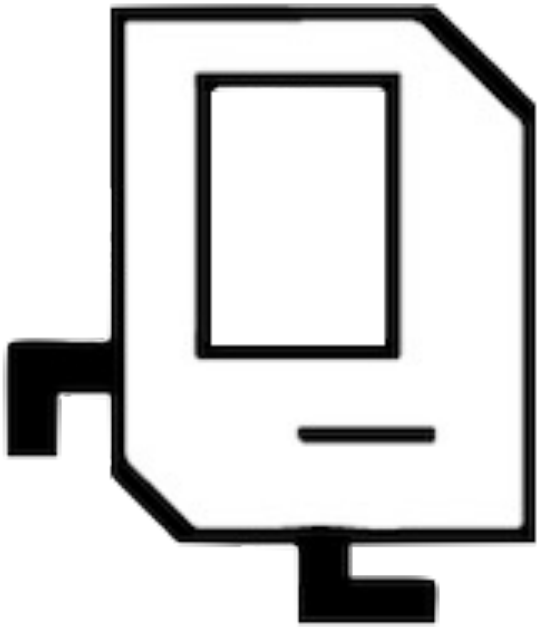
Everyone is Welcome

We are all learners, teachers

This is a collective community service project. You are part of our community.

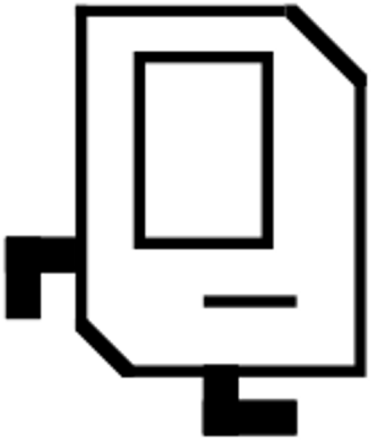
Help us make this work.

Let's learn!



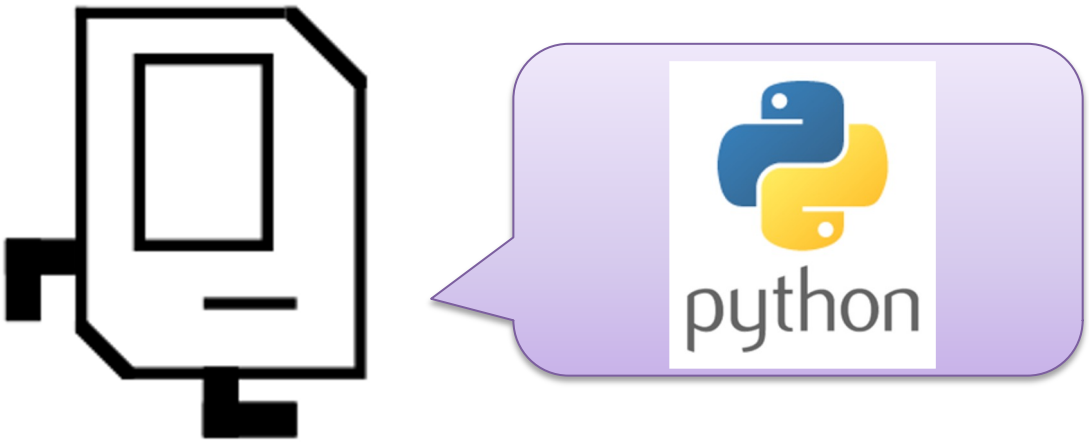
Meet Karel the Robot

Rich Pattis



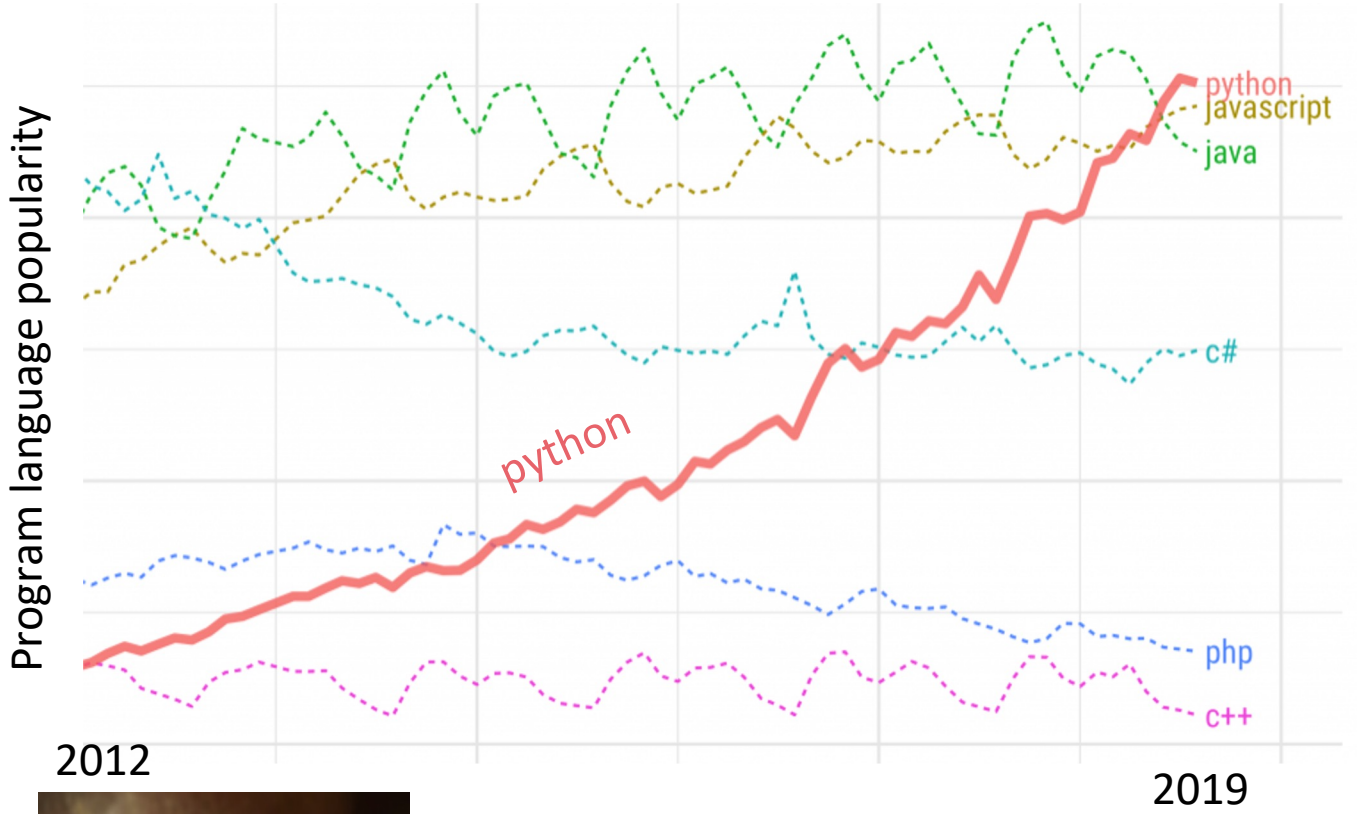
Good morning

Karel Speaks Python



Why Python?

1



2

https://stackoverflow.blog/2017/09/06/incredible-growth-python/

Guido van Rossum

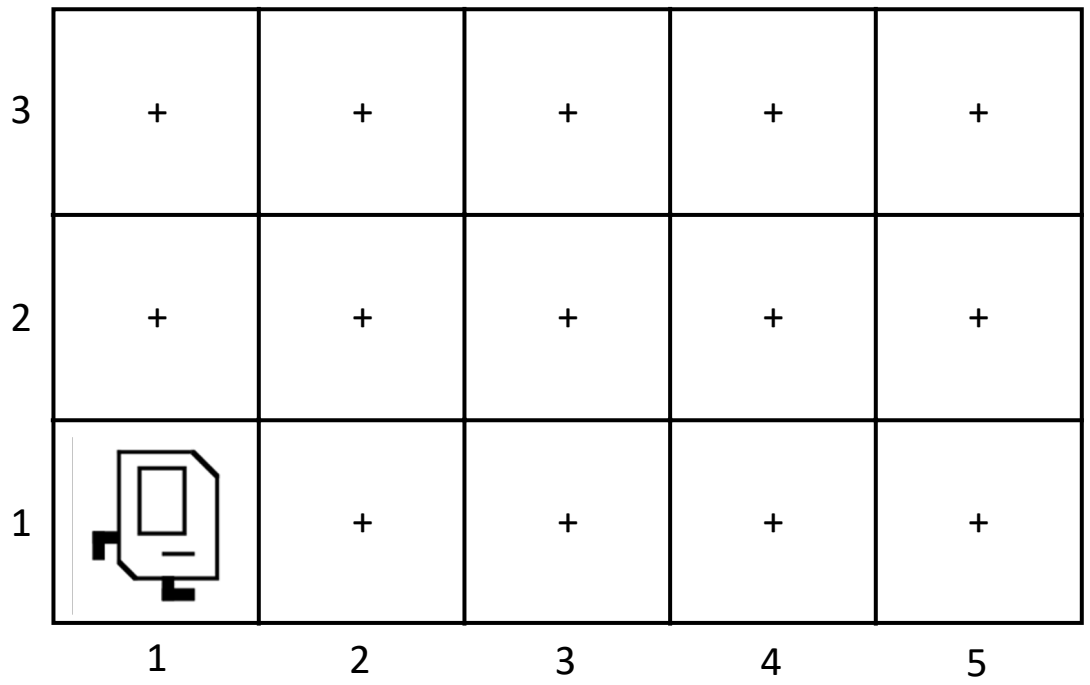
Karel's World

North

West East

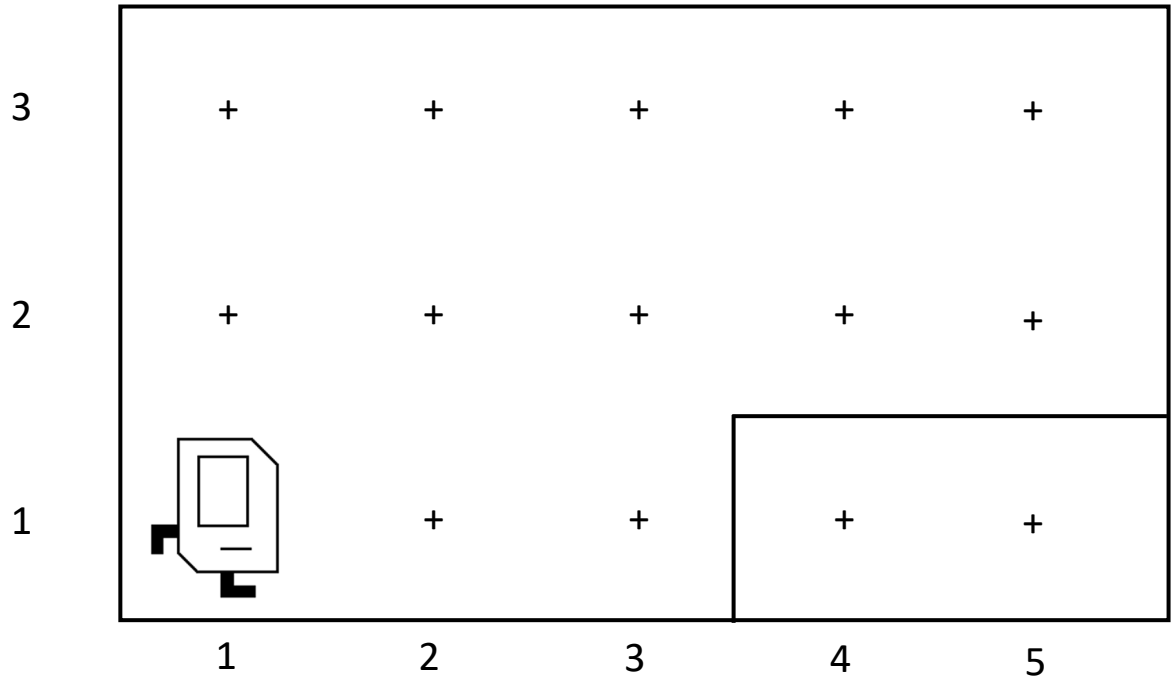
South

"columns" run Left/Right

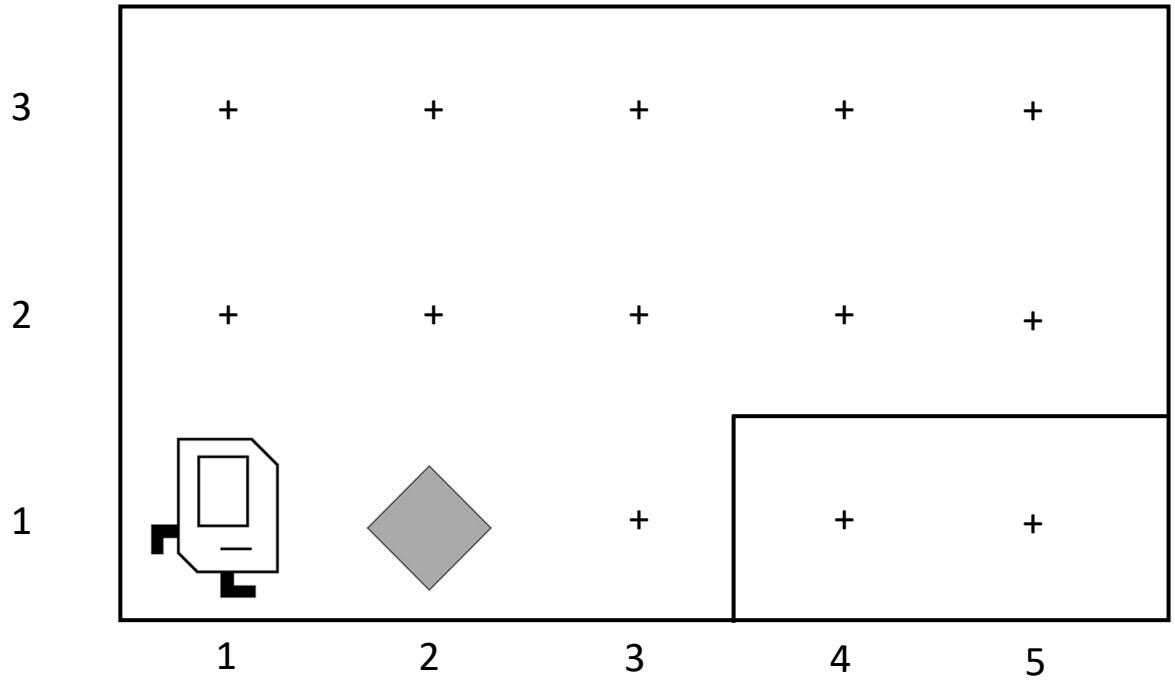


"rows" run Bottom/Top

Walls



Beepers

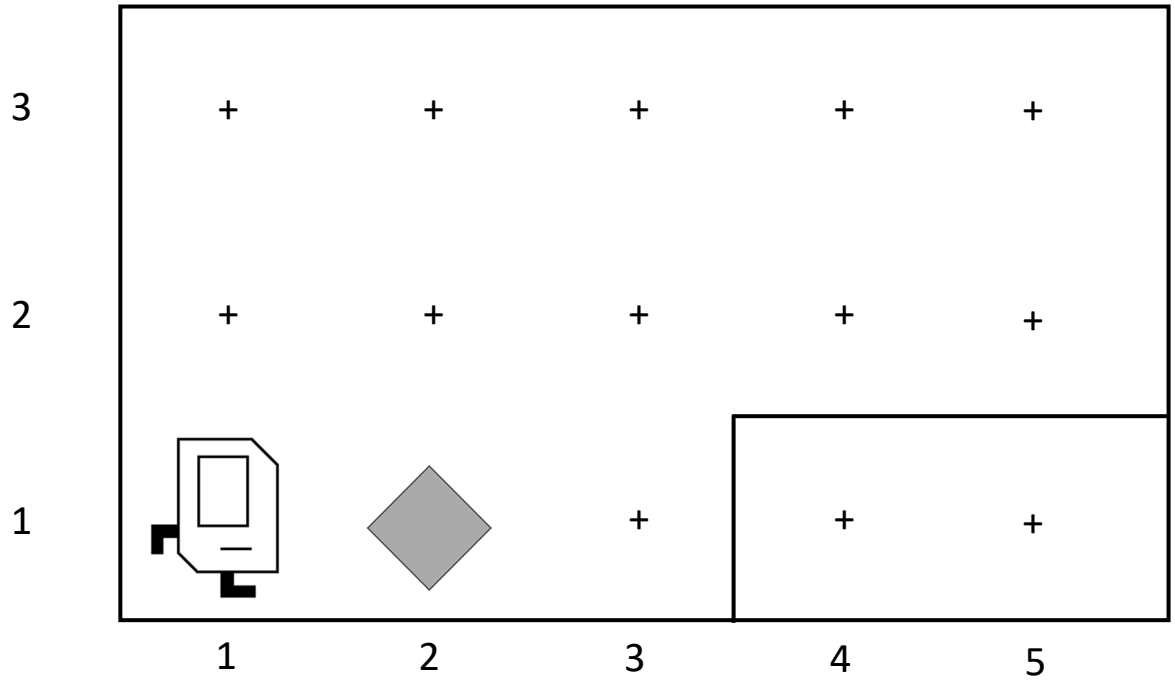


Knows Four Commands

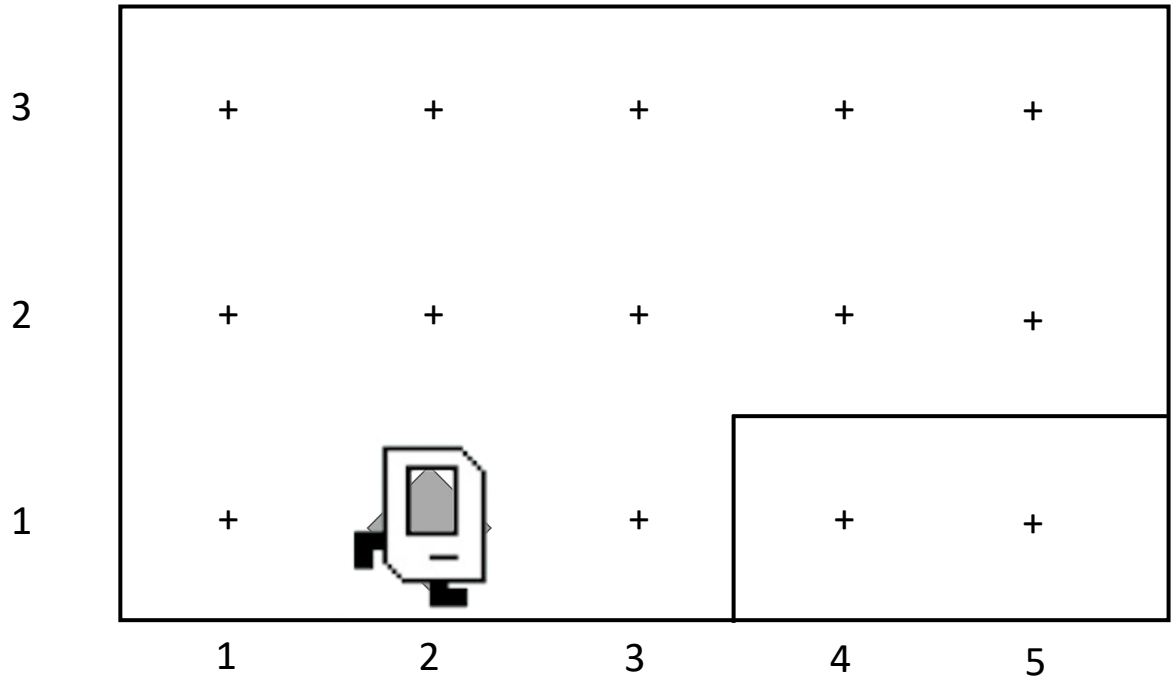
```
move()

turn_left()

put_beeper()

pick_beeper()
```

move()

move()

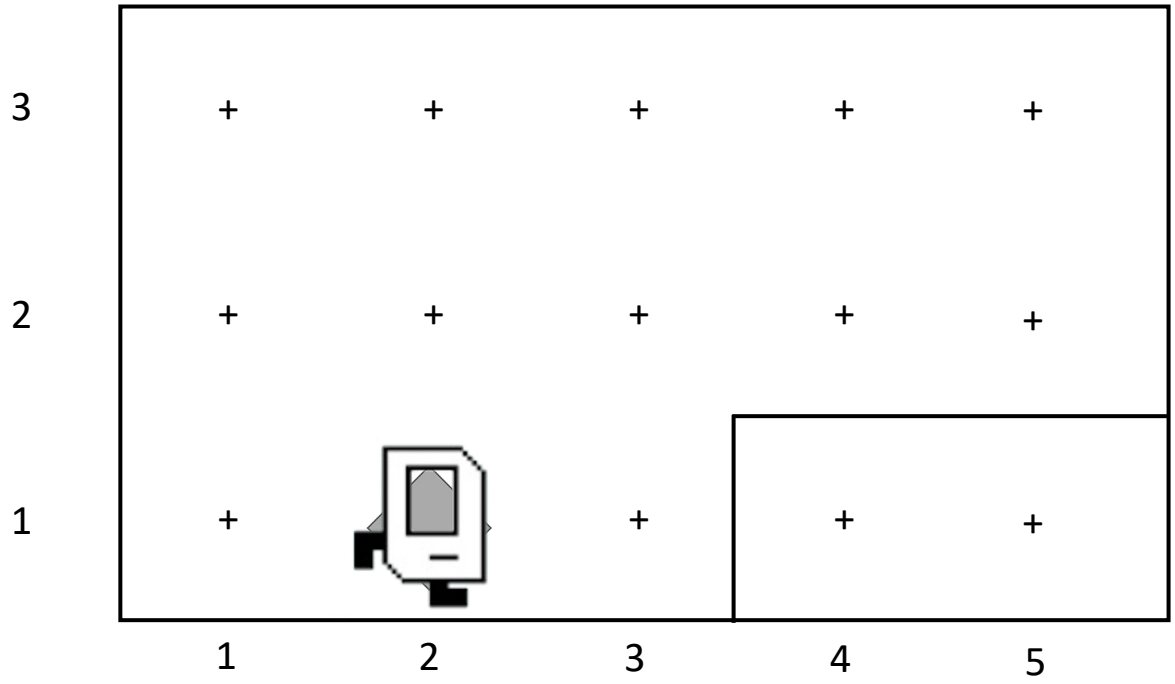


move()

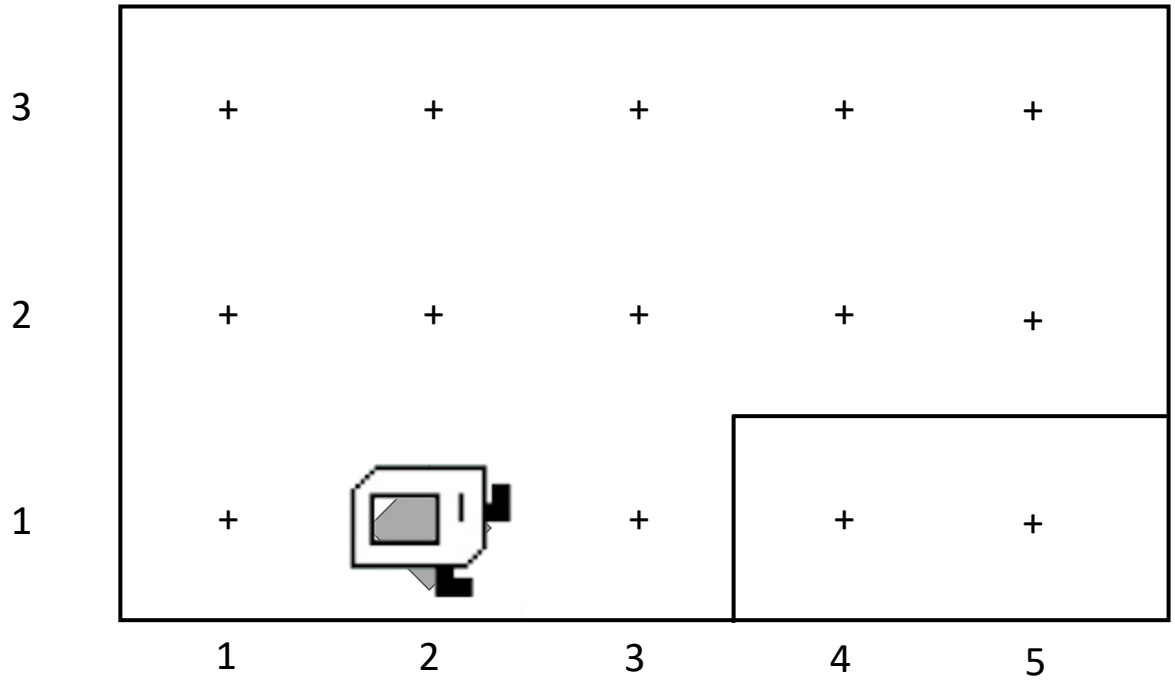


```
turn_left()
```

turn_left()

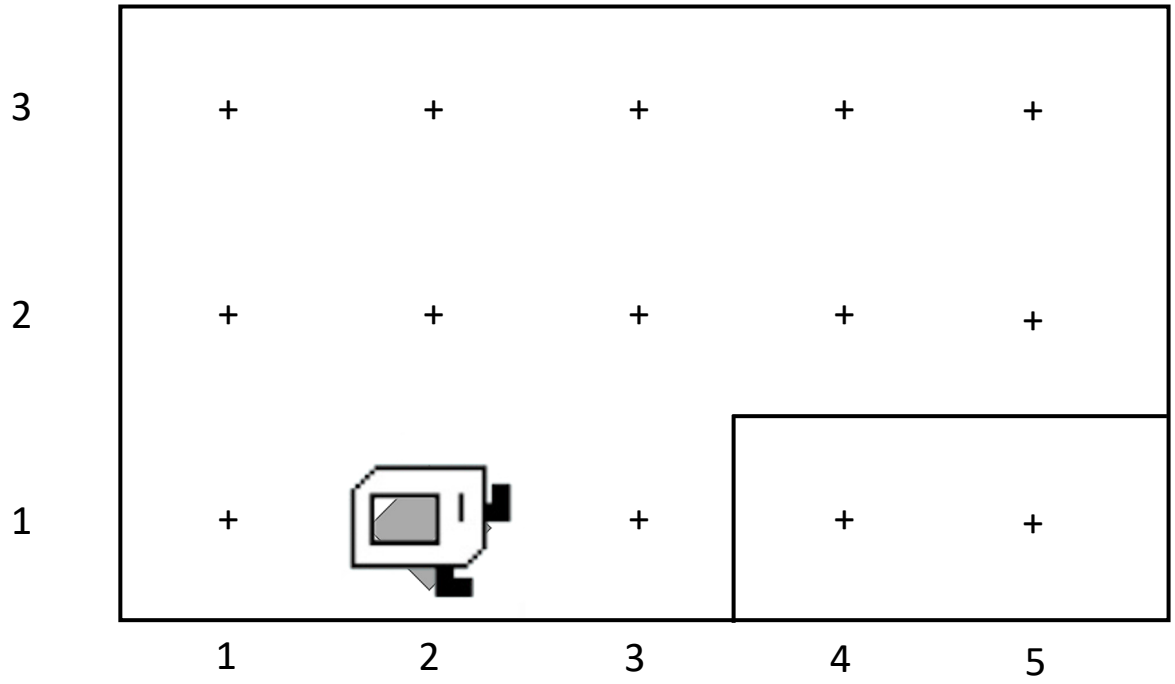


turn_left()

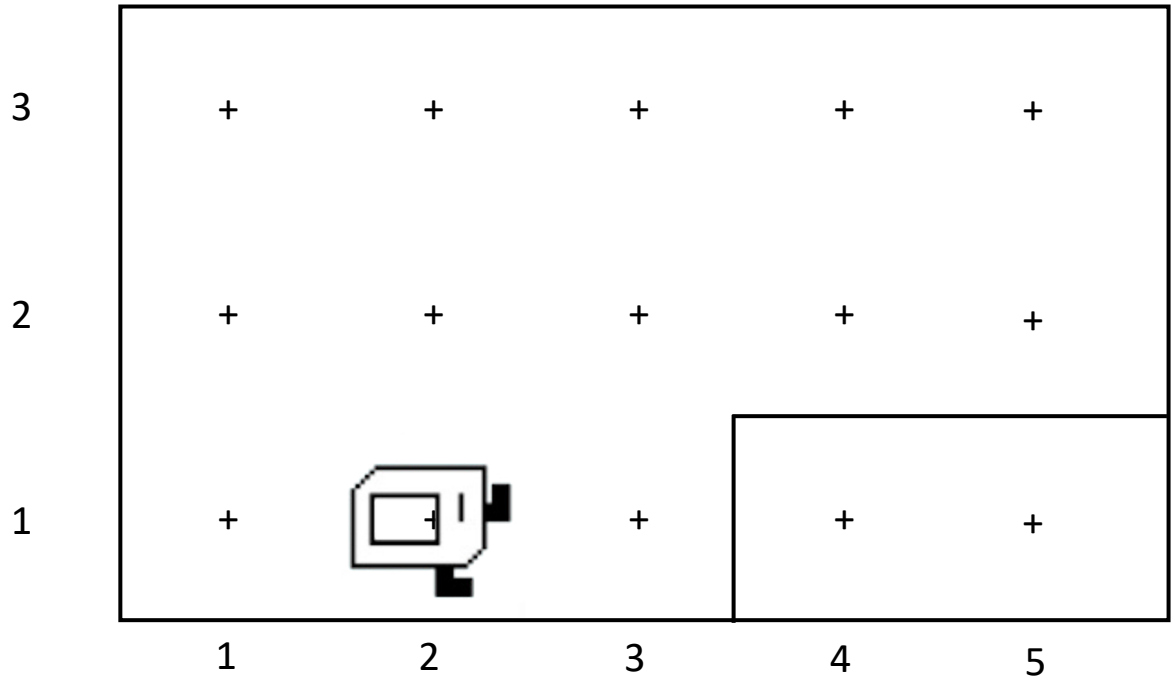


```
pick_beeper()
```

turn_left()

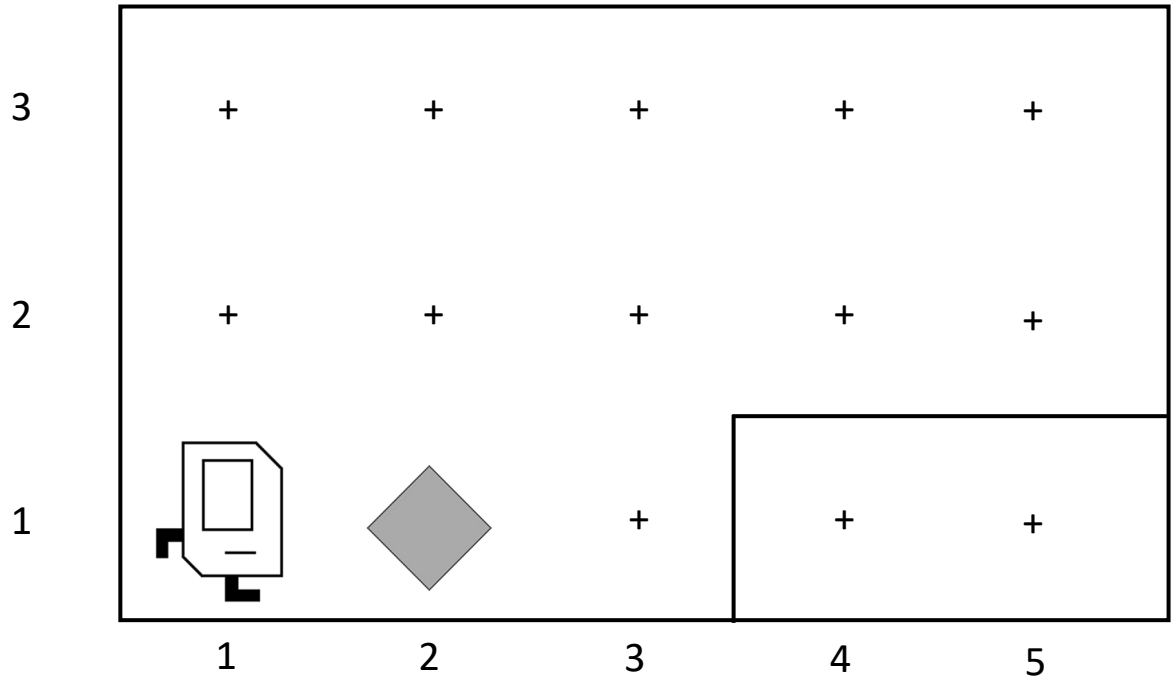


turn_left()

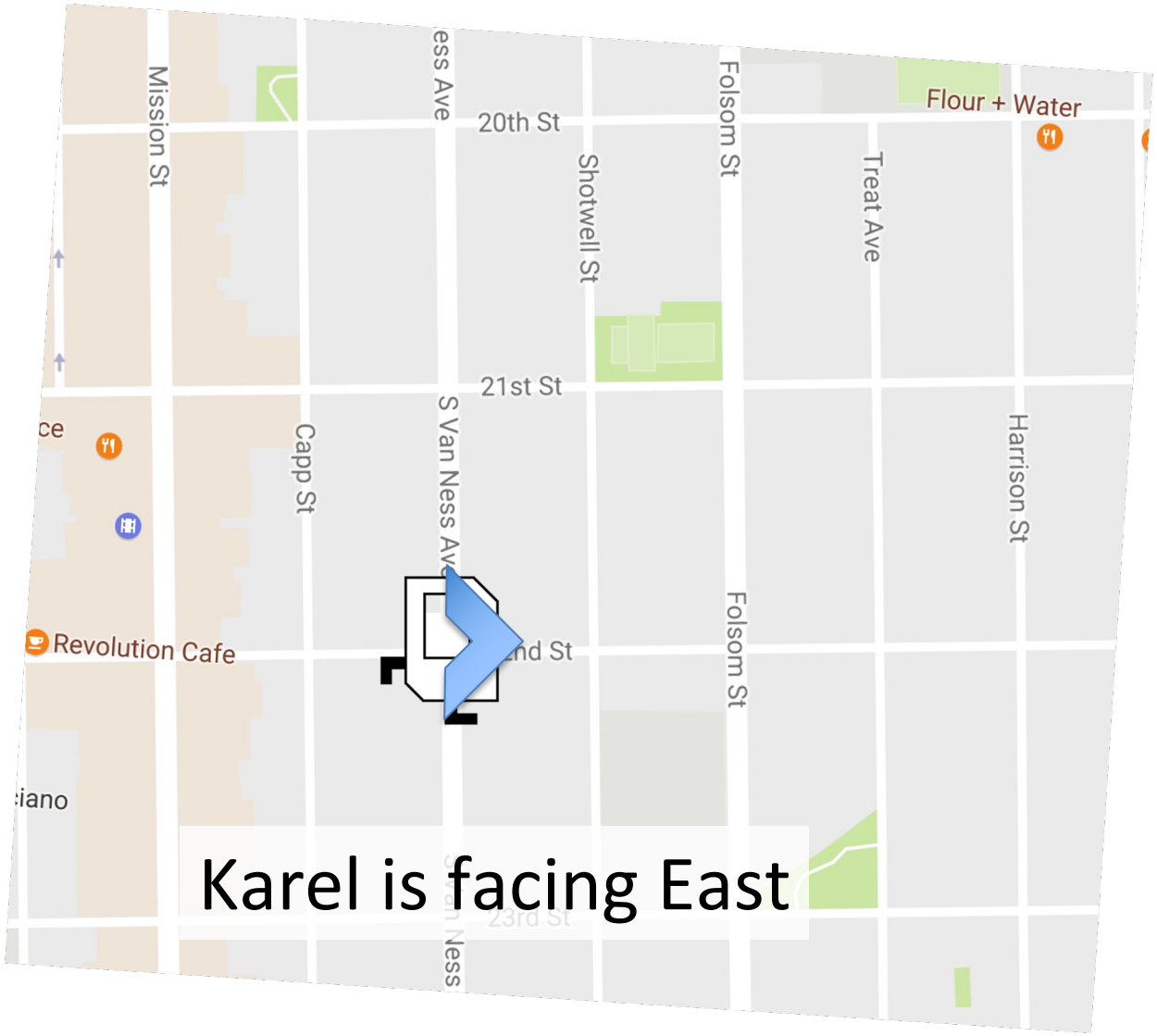


Make Sense?

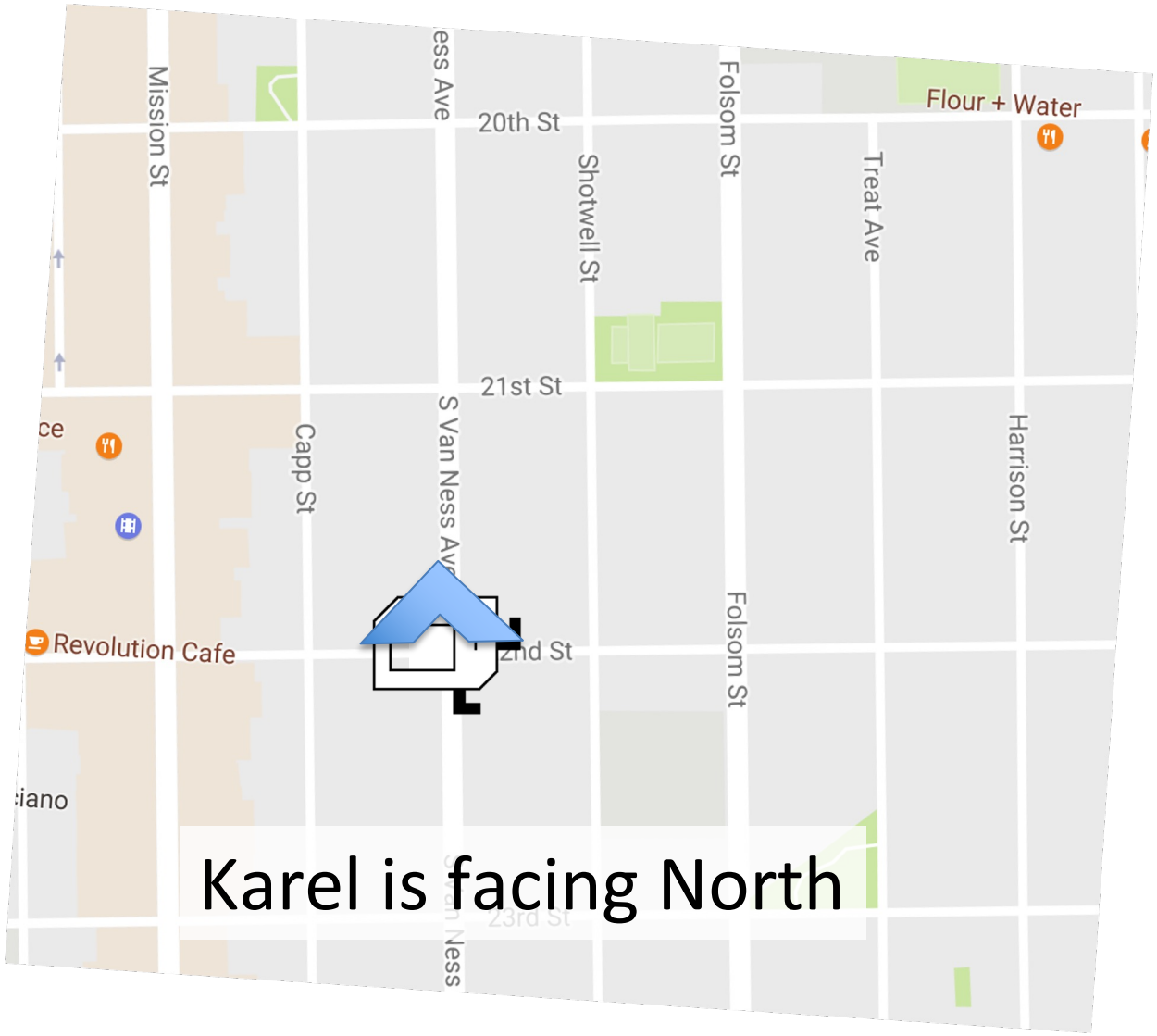
Bird's Eye View



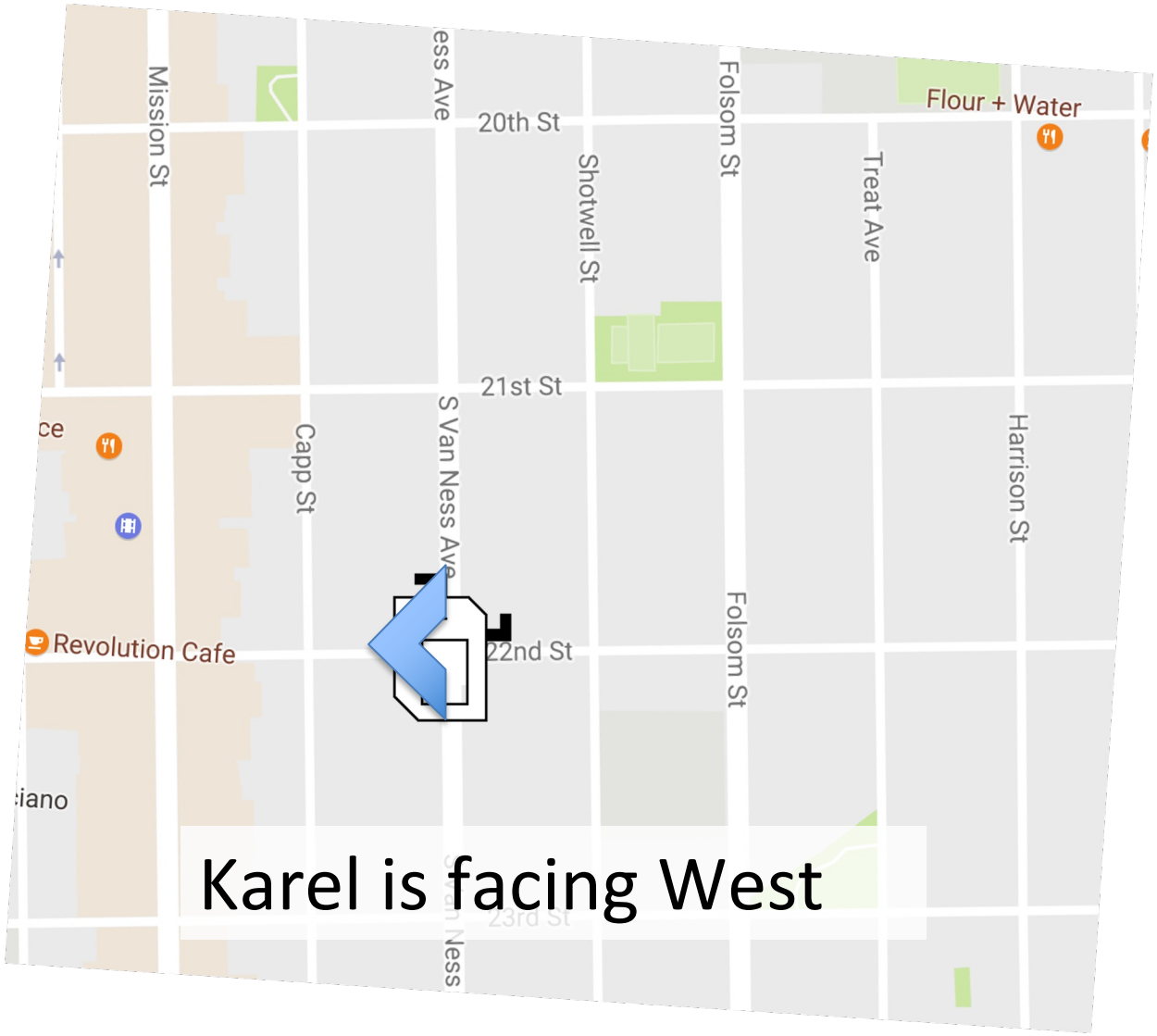
Bird's Eye View



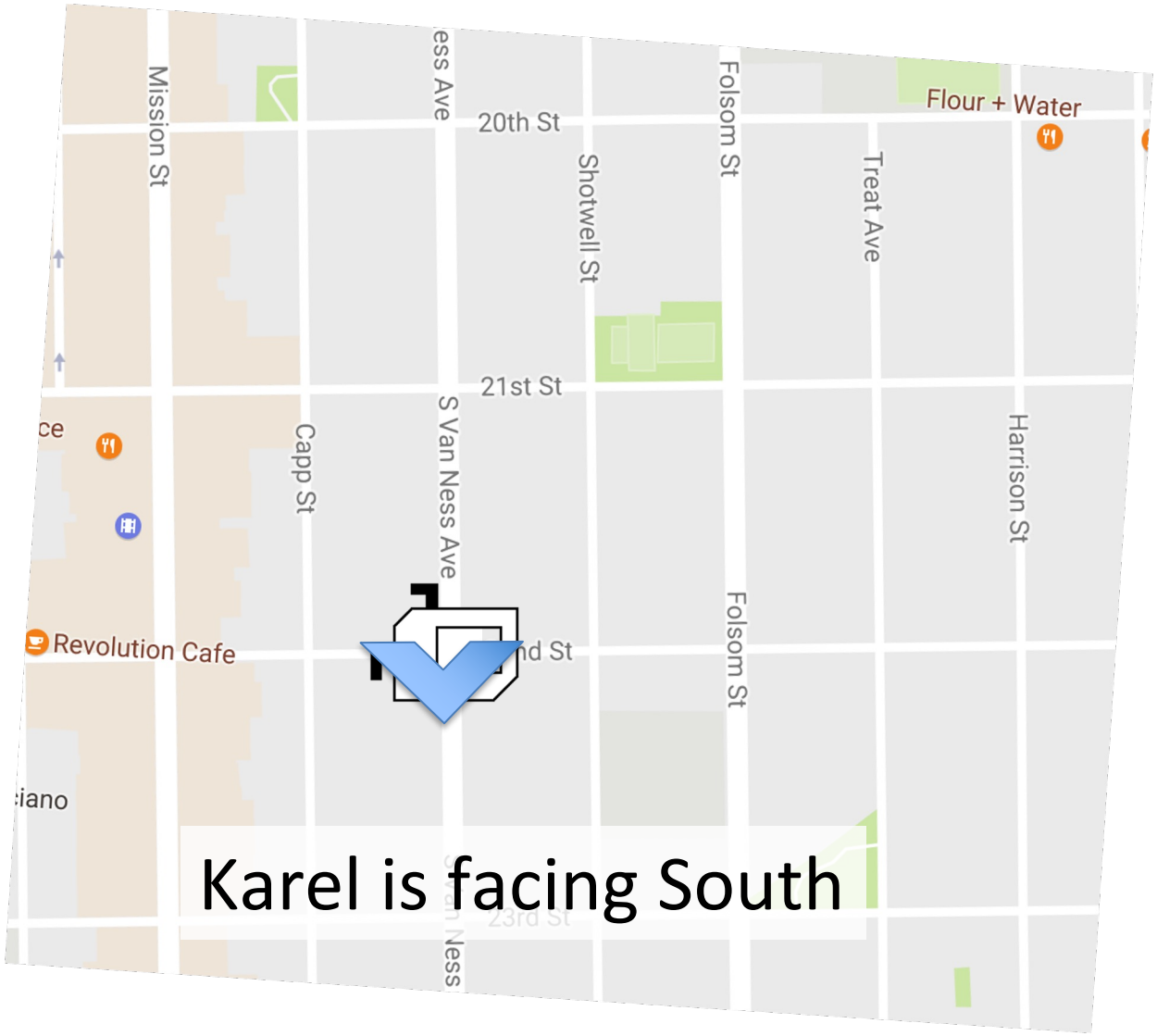
Turn Left


Mission St
ess Ave
20th St
Shotwell St
Folsom St
Treat Ave
Flour + Water
21st St
Capp St
S Van Ness Ave
Harrison St
Revolution Cafe
Folsom St
Karel is facing North

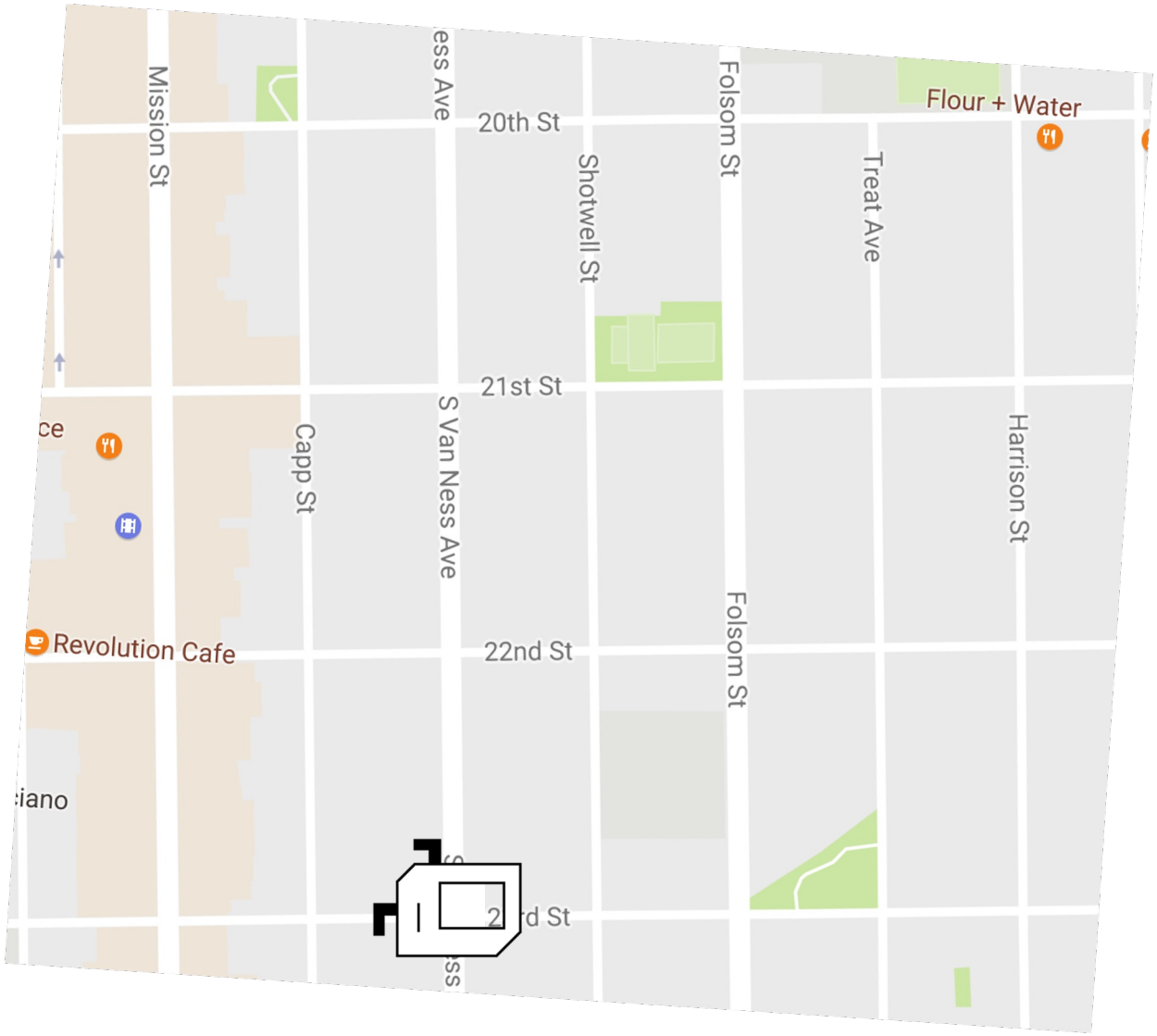
Turn Left


Mission St
ess Ave
20th St
Shotwell St
Folsom St
Treat Ave
Flour + Water
21st St
Capp St
S Van Ness Ave
Harrison St
Revolution Cafe
22nd St
Folsom St
iano
Karel is facing West
23rd St

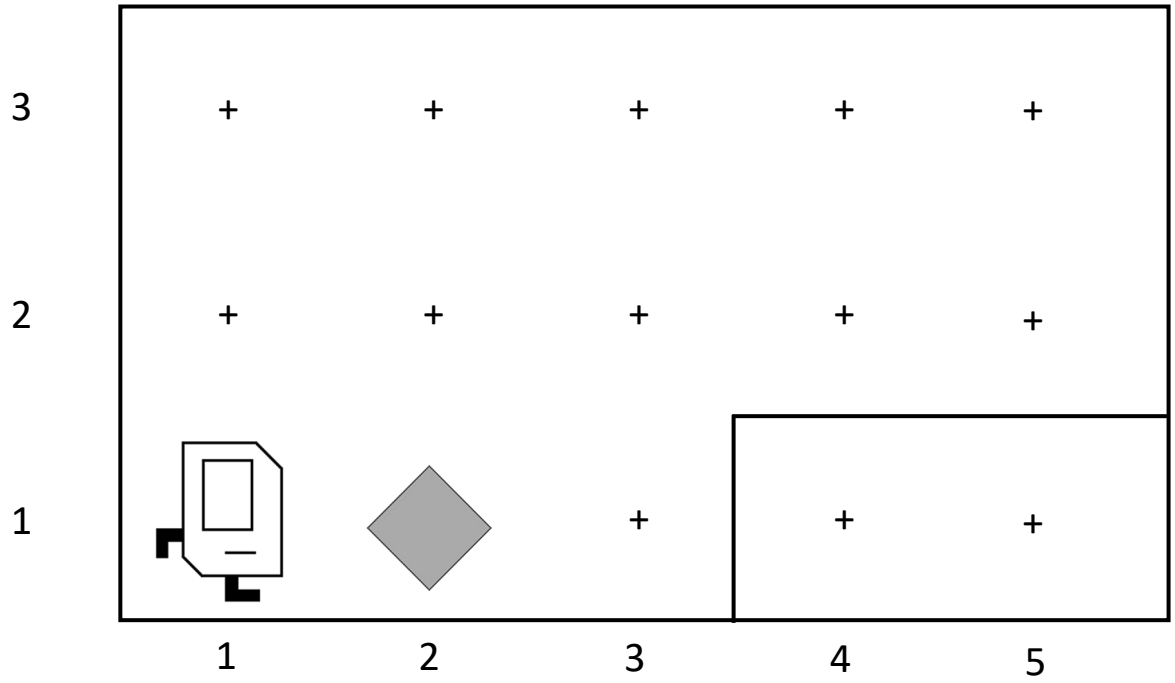
Turn Left


Mission St
ess Ave
20th St
Shotwell St
Folsom St
Treat Ave
Flour + Water
21st St
Capp St
S Van Ness Ave
Harrison St
Revolution Cafe
Folsom St
Karel is facing South

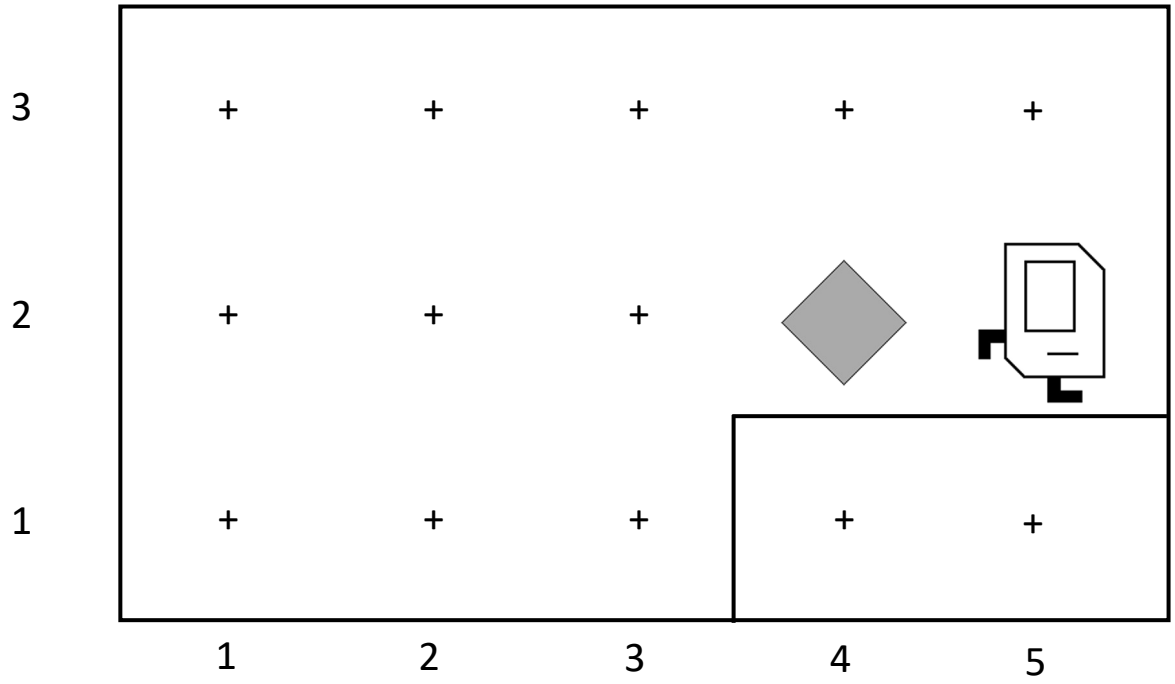
Move


Mission St
ess Ave
20th St
Shotwell St
Folsom St
Treat Ave
Flour + Water
21st St
Capp St
S Van Ness Ave
Harrison St
Revolution Cafe
22nd St
Folsom St

First Challenge



First Challenge



Midjourney AI

Learn By Doing


PC
PyCharm
The Python IDE
for Professional
Developers
DOWNLOAD
Full-fledged Professional or Free Community

Function Definition

```
def name():
    function statements
```

This adds a new command to Karel's vocabulary

Anatomy of a Program

Import Packages

Program

Anatomy of a Program

```
Import Packages
```

Anatomy of a Program

```
Import Packages
```

```
main function
```

```
helper functions
```

```
start program
```

Anatomy of a Program

```
Import Packages
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

```
helper functions
```

```
start program
```

Anatomy of a Program

Import Packages

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()
```

start program

Anatomy of a Program

```
Import Packages
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

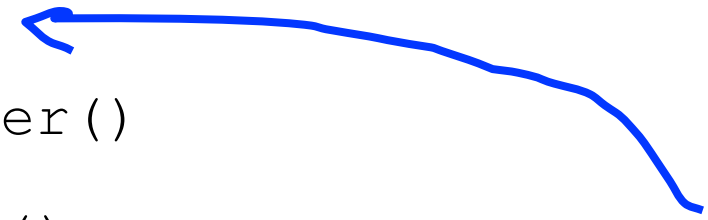
Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This piece of the program's ***source code*** is called a ***function***.

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```



This line of code gives the ***name*** of the function (here, the name is: **`main`**)

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This line of code gives the ***name*** of the function
(here, the name is: **`turn_right`**)

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This is called a ***code block***
(Note the indenting)

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This is called a ***code block***
(Note the indenting)

Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This is called a ***code block***
(Note the indenting)