



Darüşşafaka
Lisesi. 2014



Boğaziçi Ün. 2015



Koç Ün. 2016



Koç Ün. 2017



CTU, Czech Republic +
Koç Ün. 2018



Asena



Bryce



Julia



Nick



Chris

Our awesome 2019
Koç teaching team!



Lisa



Barış



Kaan



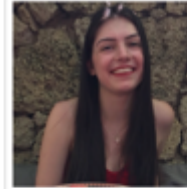
Ahmet



Beyzanur



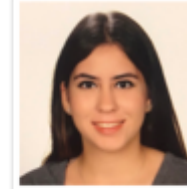
Ceren



Ece



Eren



Ezgi



Gül Sena



Haluk



Hasan



İpek



Levent



Necla



Oğuzhan



Ozan D



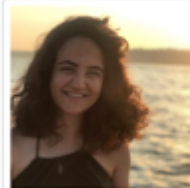
Ozan N



Quincy



Sabri



Seher



Serhat



Chris



Nick



Asena



Pelin



Barış

Logistics

9:30-10:15	AM Lecture
10:20-11:15	Lab 1
11:20-12:15	Section
12:15-13:15	Lunch
13:15-14:15	PM Lecture
14:15-15:30	Lab 2
15:30-16:00	Break
16:00-17:15	Lab 3

LOL: Lots of (computer) Labs!



Labrador Retrievers (Labs)

Logistics

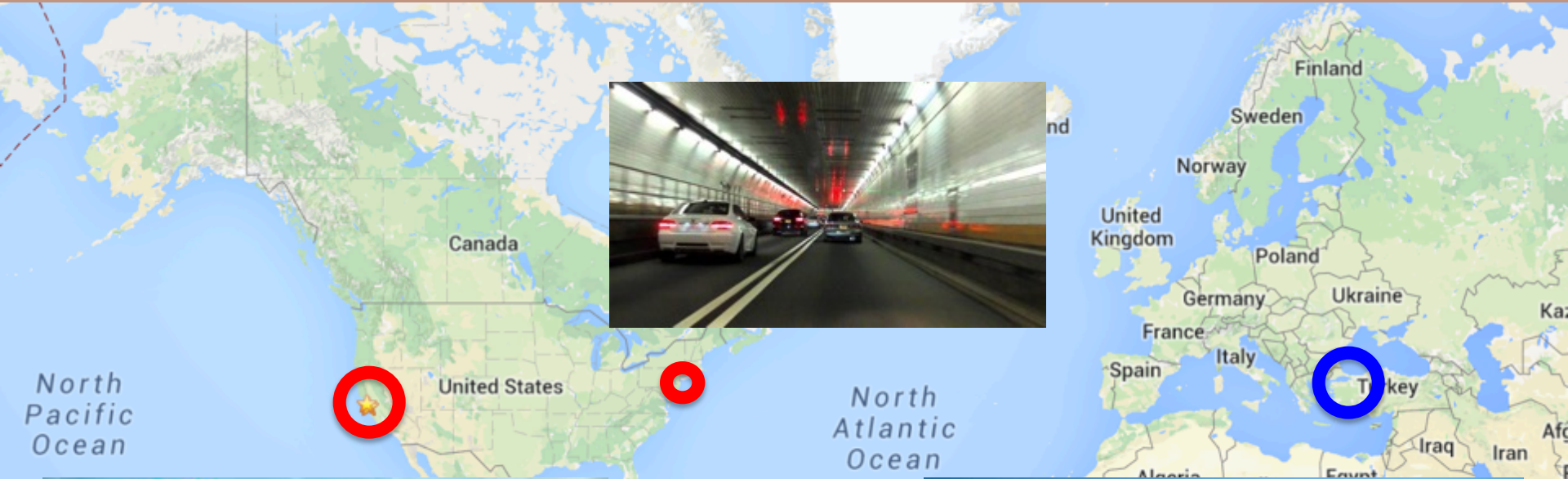
9:30-10:15	AM Lecture
10:20-11:15	Lab 1
11:20-12:15	Section
12:15-13:15	Lunch
13:15-14:15	PM Lecture
14:15-15:30	Lab 2
15:30-16:00	Break
16:00-17:15	Lab 3



Discussion Section

Meet your friends
and talk through problems
with your Section Leader!

Stanford?



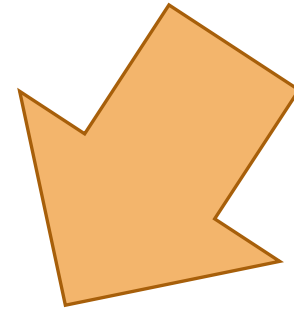
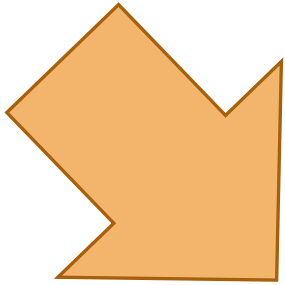
Stanford



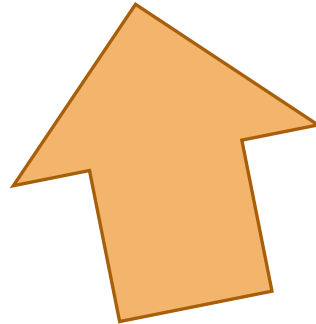
Prerequisites



Course Website

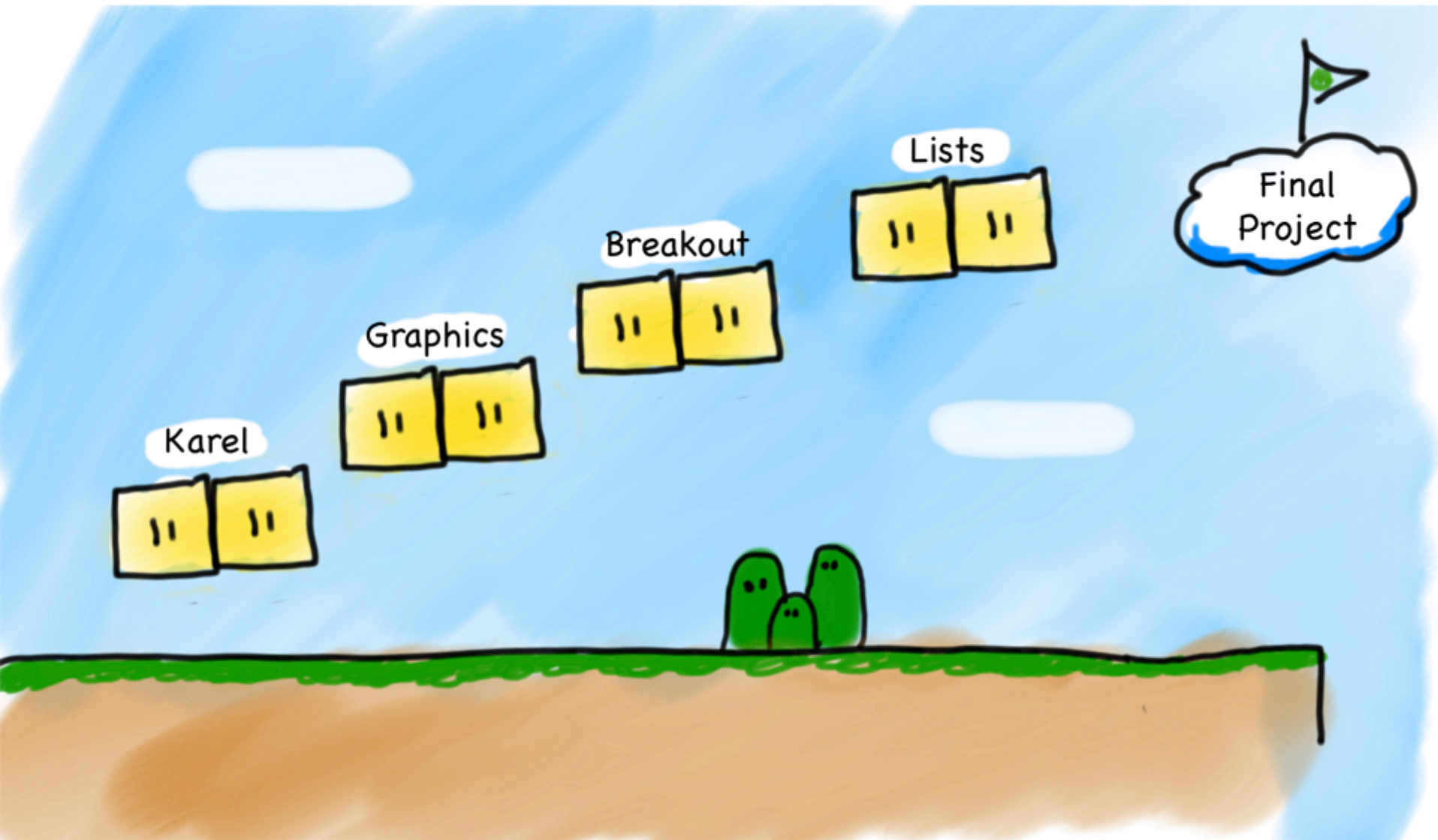


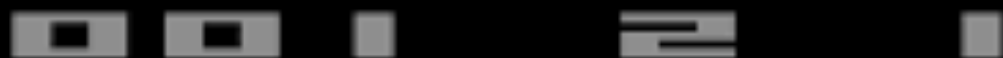
<http://koc.csbridge.org>



*note that its **org**, not **com**

Very High Level Journey





Breakout

What if I fall behind?



CS
Bridge

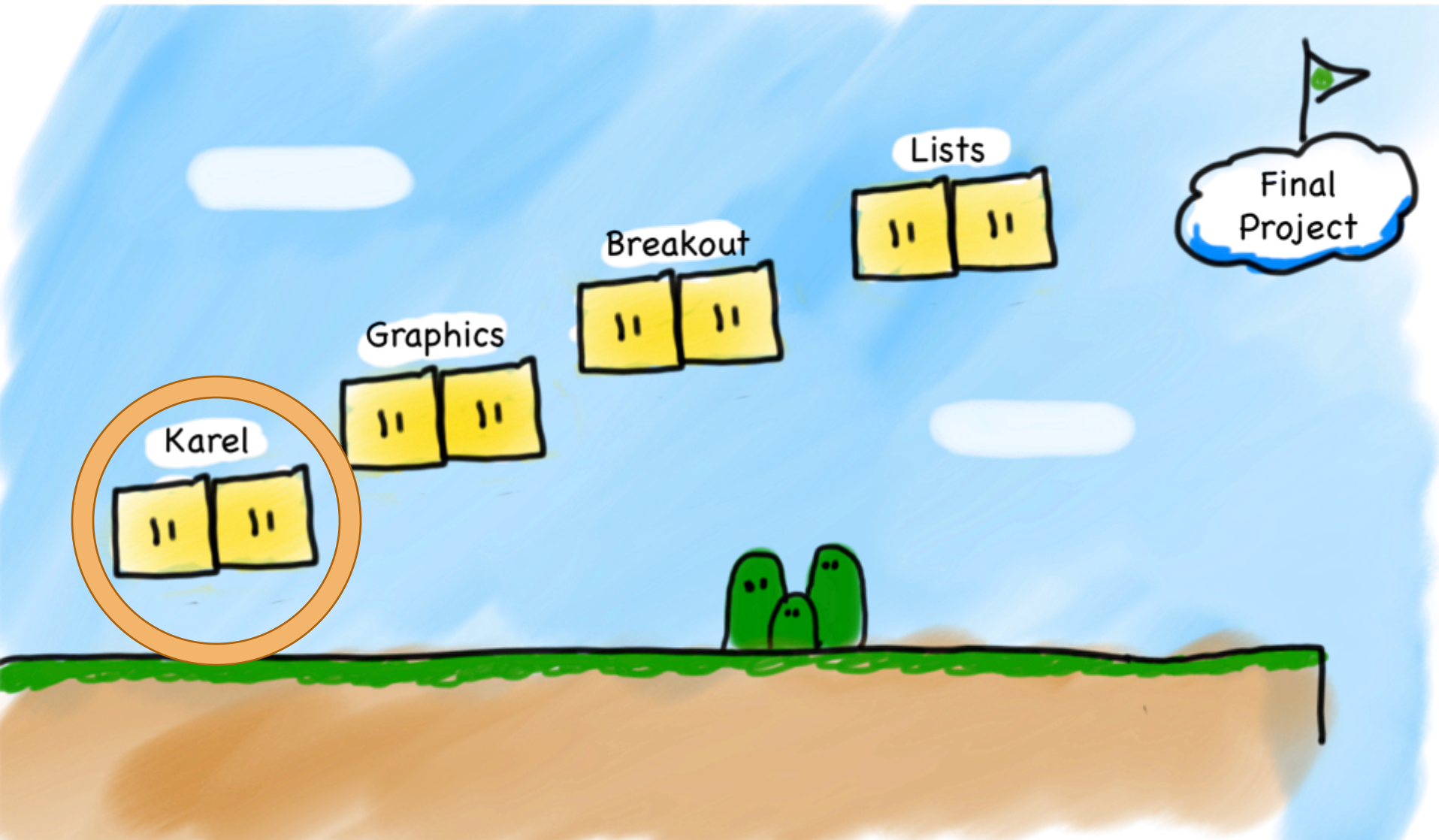


Share Ideas Not Code



Questions?

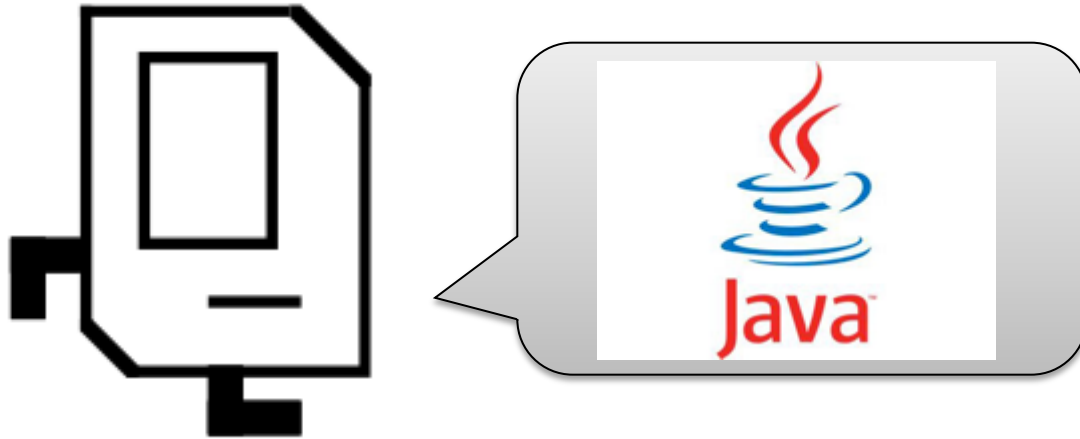
Our First Step



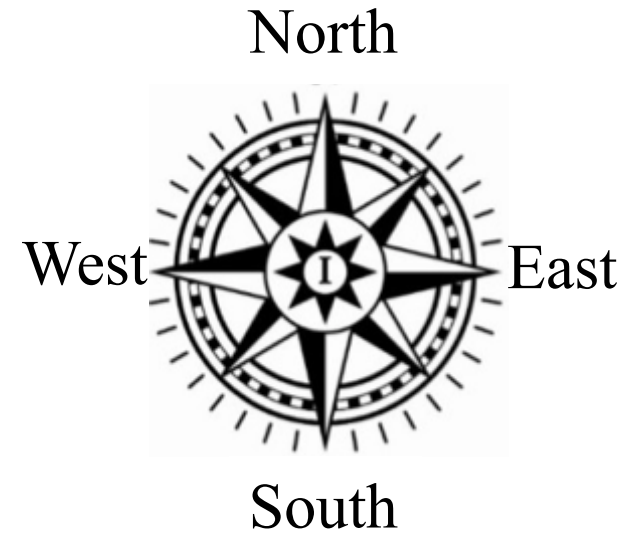
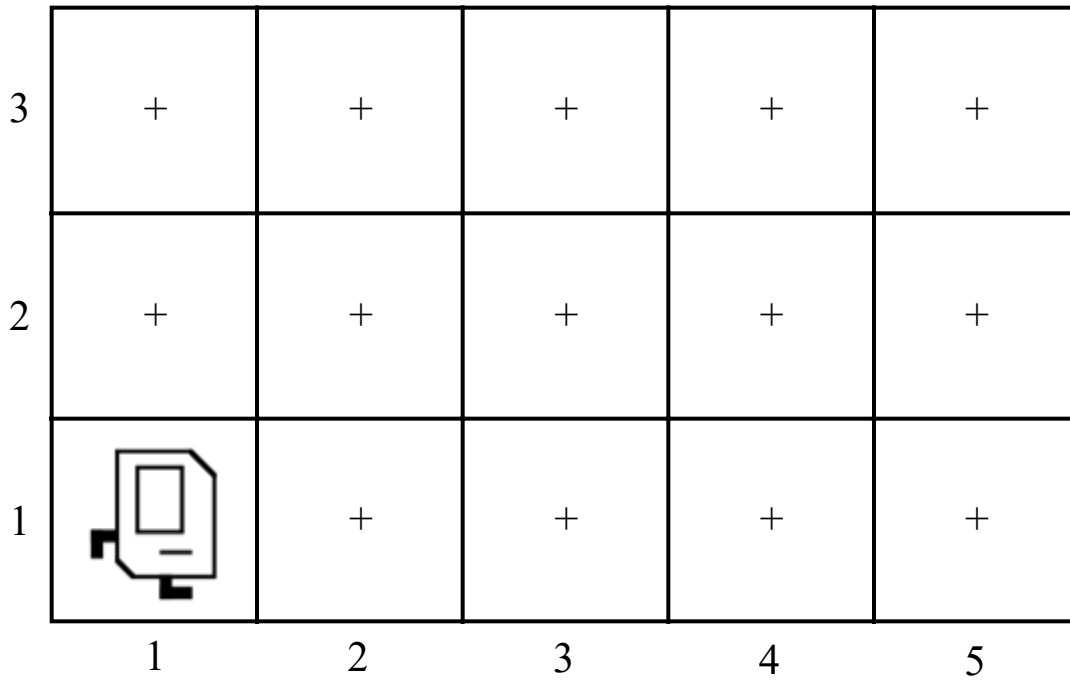
Karel



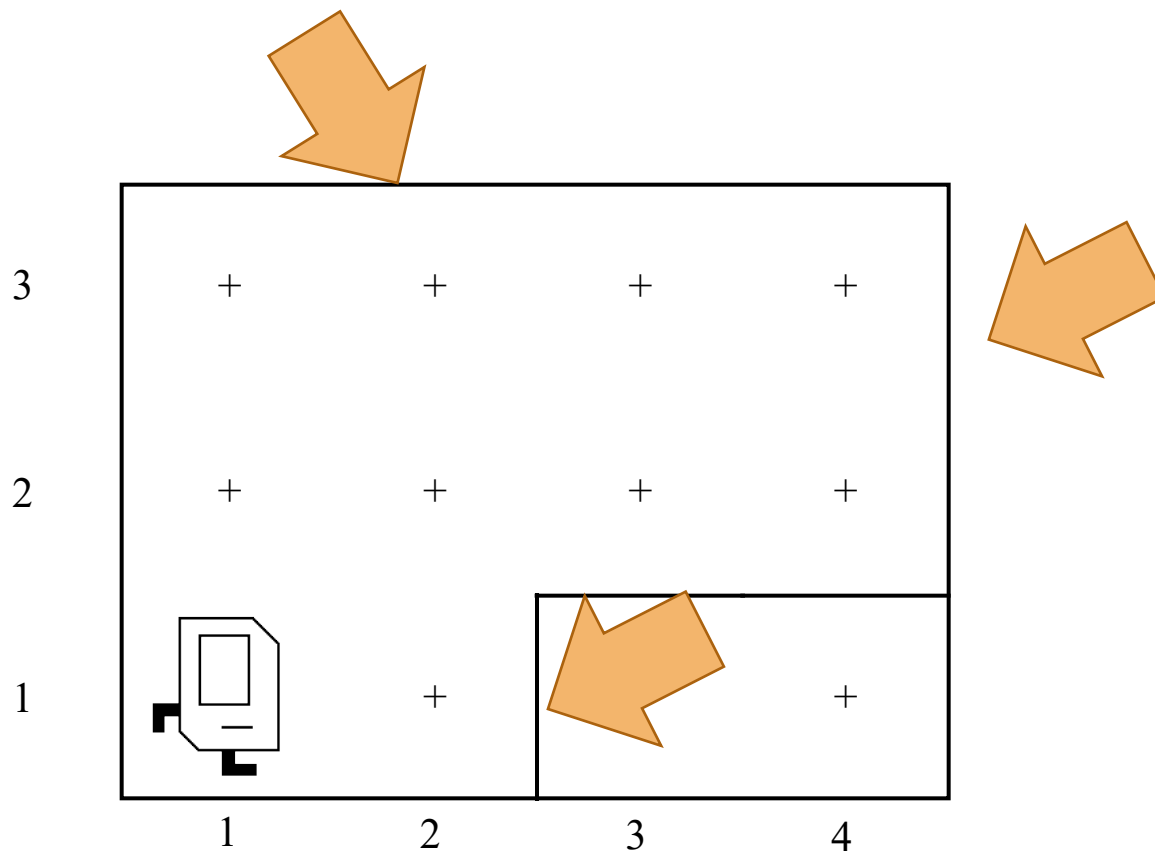
Karel Speaks Java



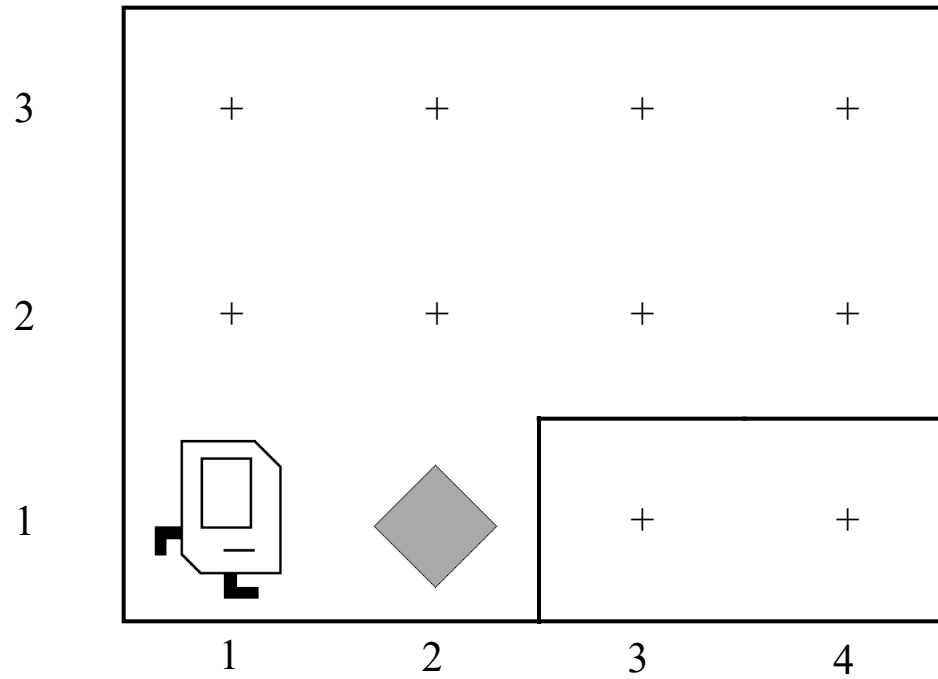
Karel's World



Walls



Beeepers



Knows Four Commands



```
move();
```

```
turnLeft();
```

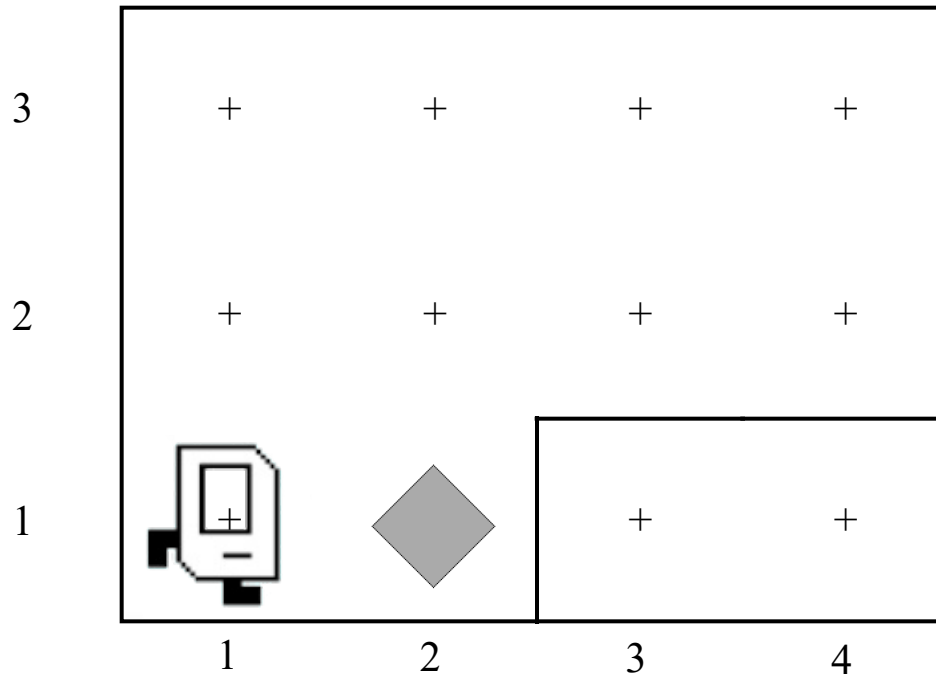
```
pickBeeper();
```

```
putBeeper();
```

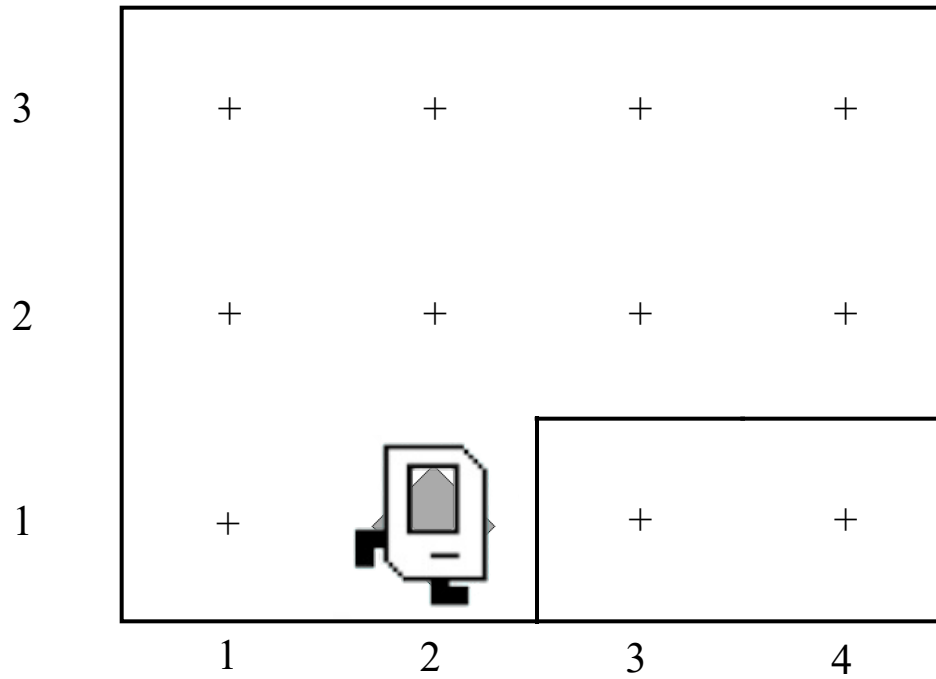


```
move ( ) ;
```

move () ;

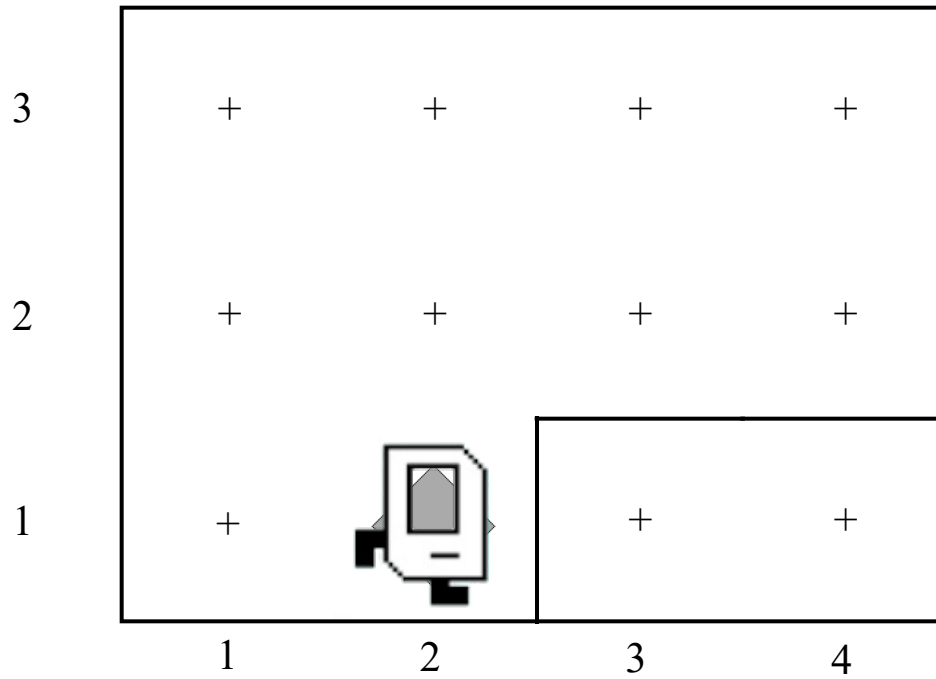


move () ;

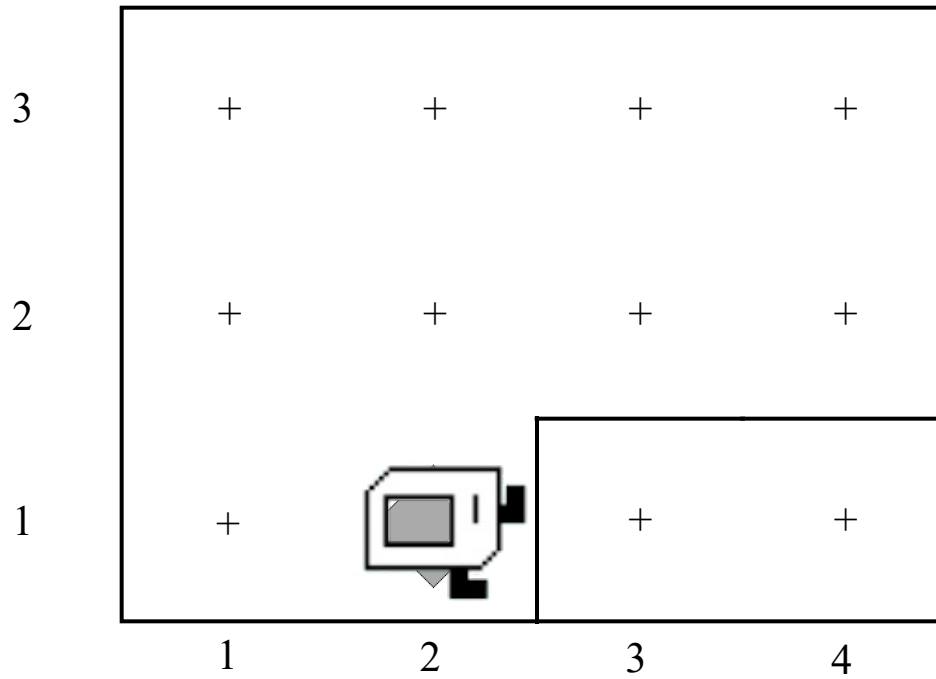


```
turnLeft();
```

turnLeft();

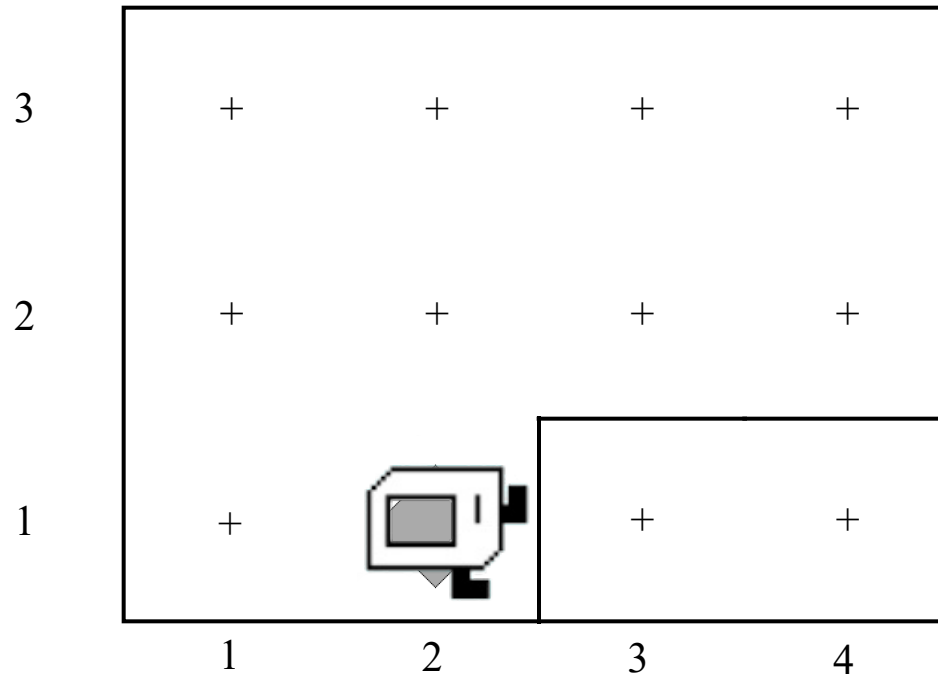


turnLeft();

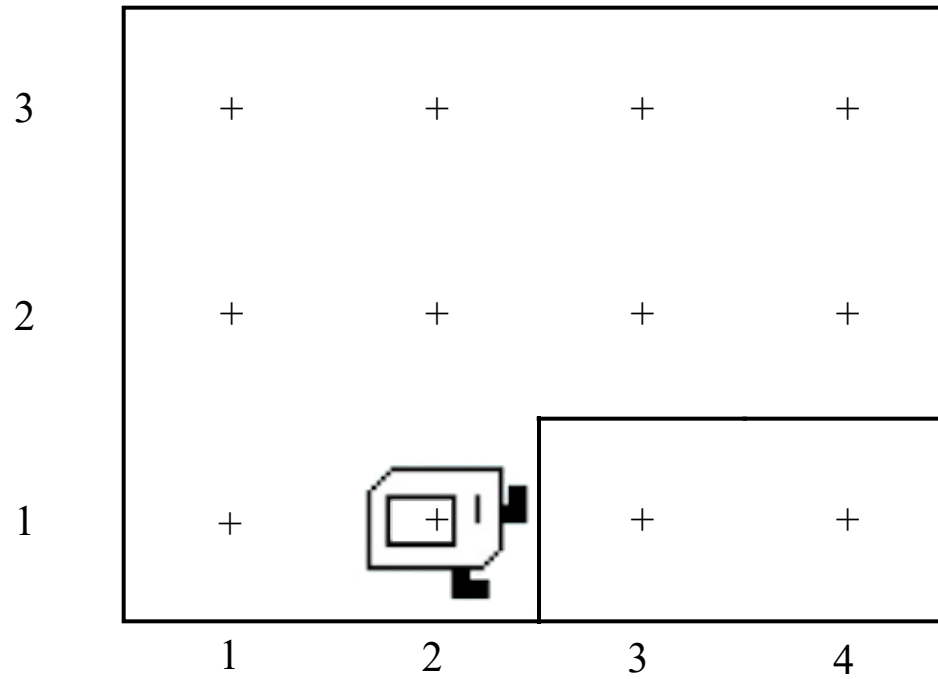


```
pickBeeper ( ) ;
```

pickBeeper () ;

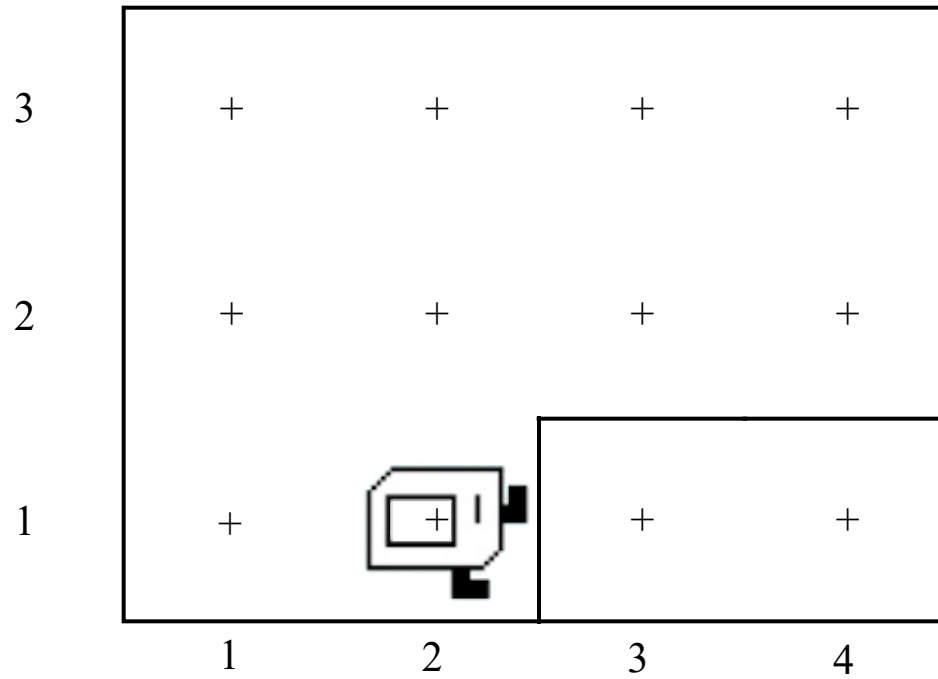


`pickBeeper () ;`

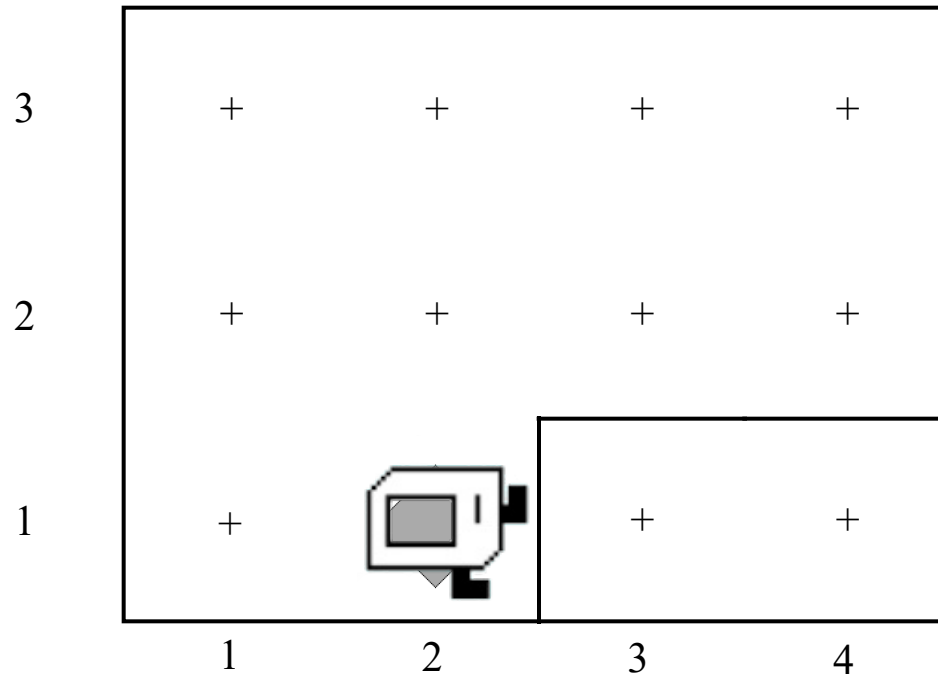


```
putBeeper ( ) ;
```

putBeeper () ;

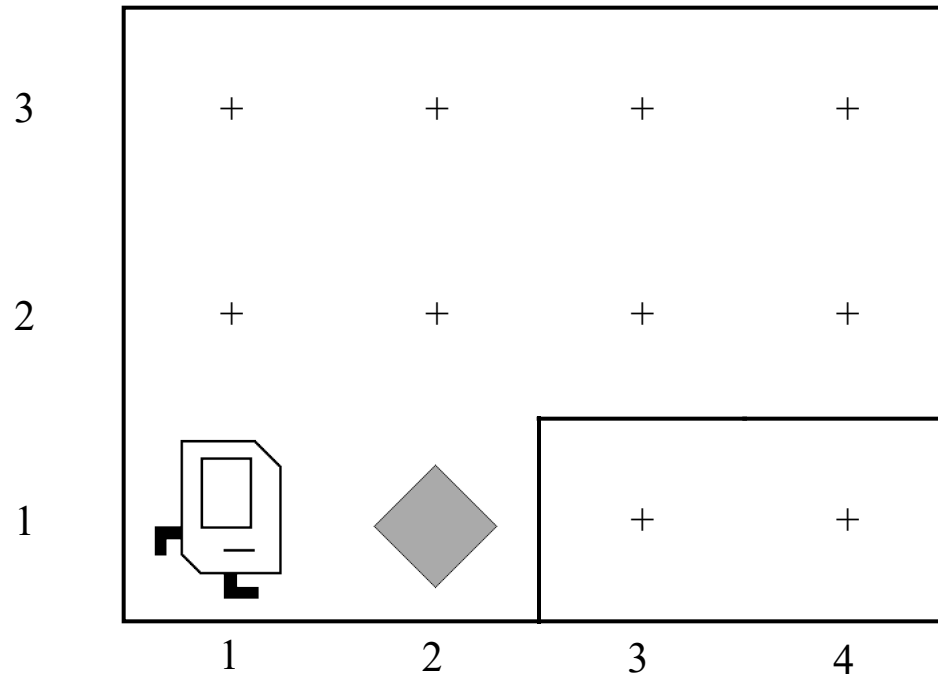


putBeeper () ;

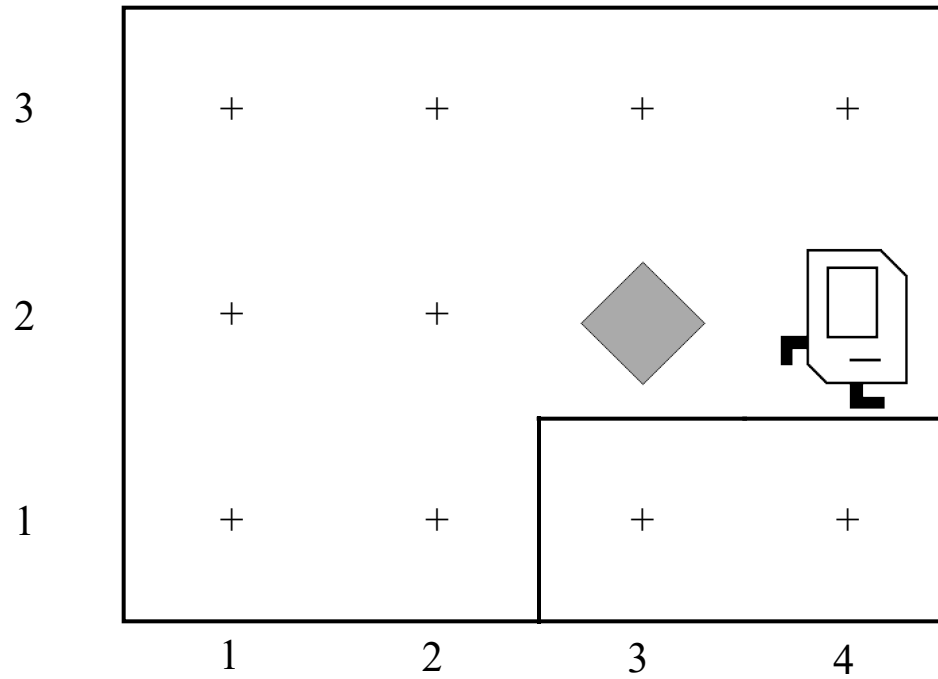


Questions?

First Challenge

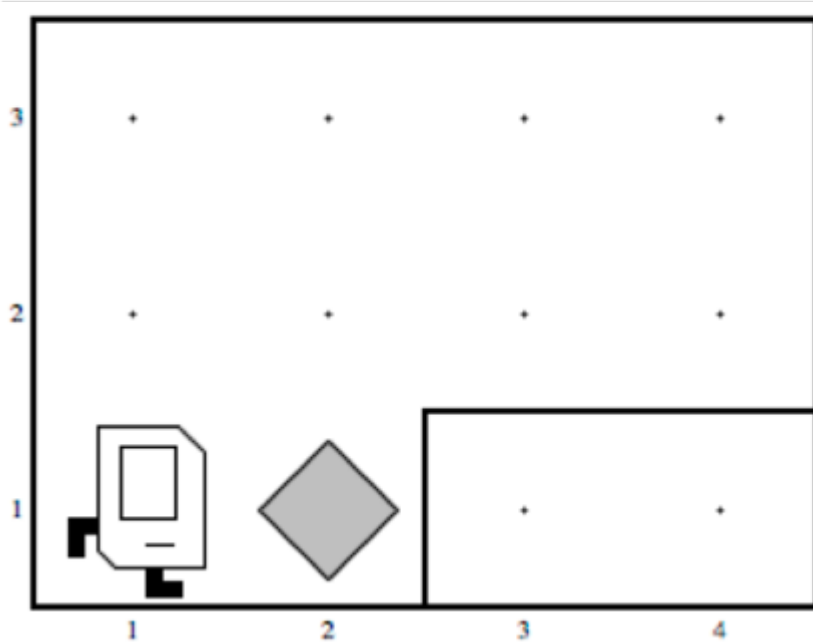


First Challenge

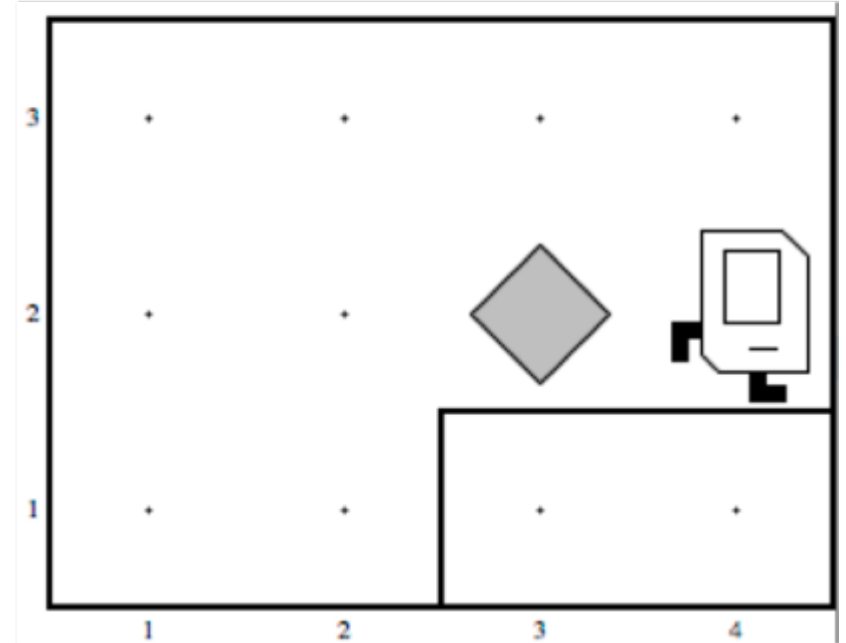


First Challenge

Before



After



```
move();
```

```
pickBeeper();
```

```
turnLeft();
```

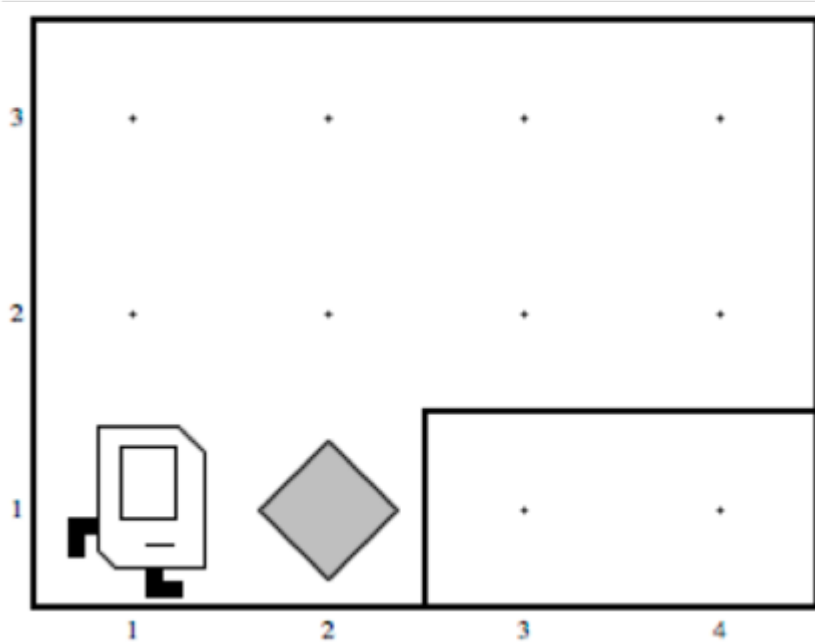
```
putBeeper();
```


Need a Volunteer

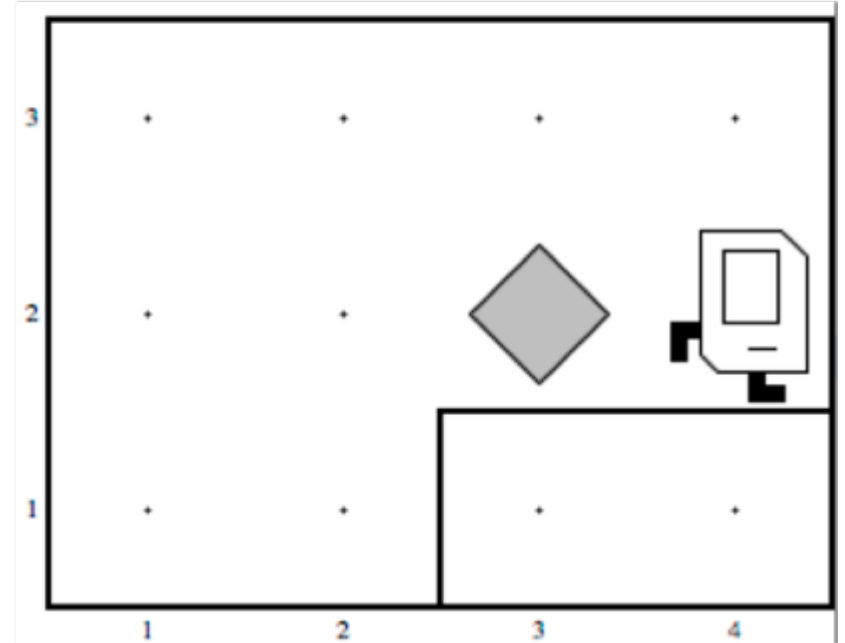


First Challenge

Before



After



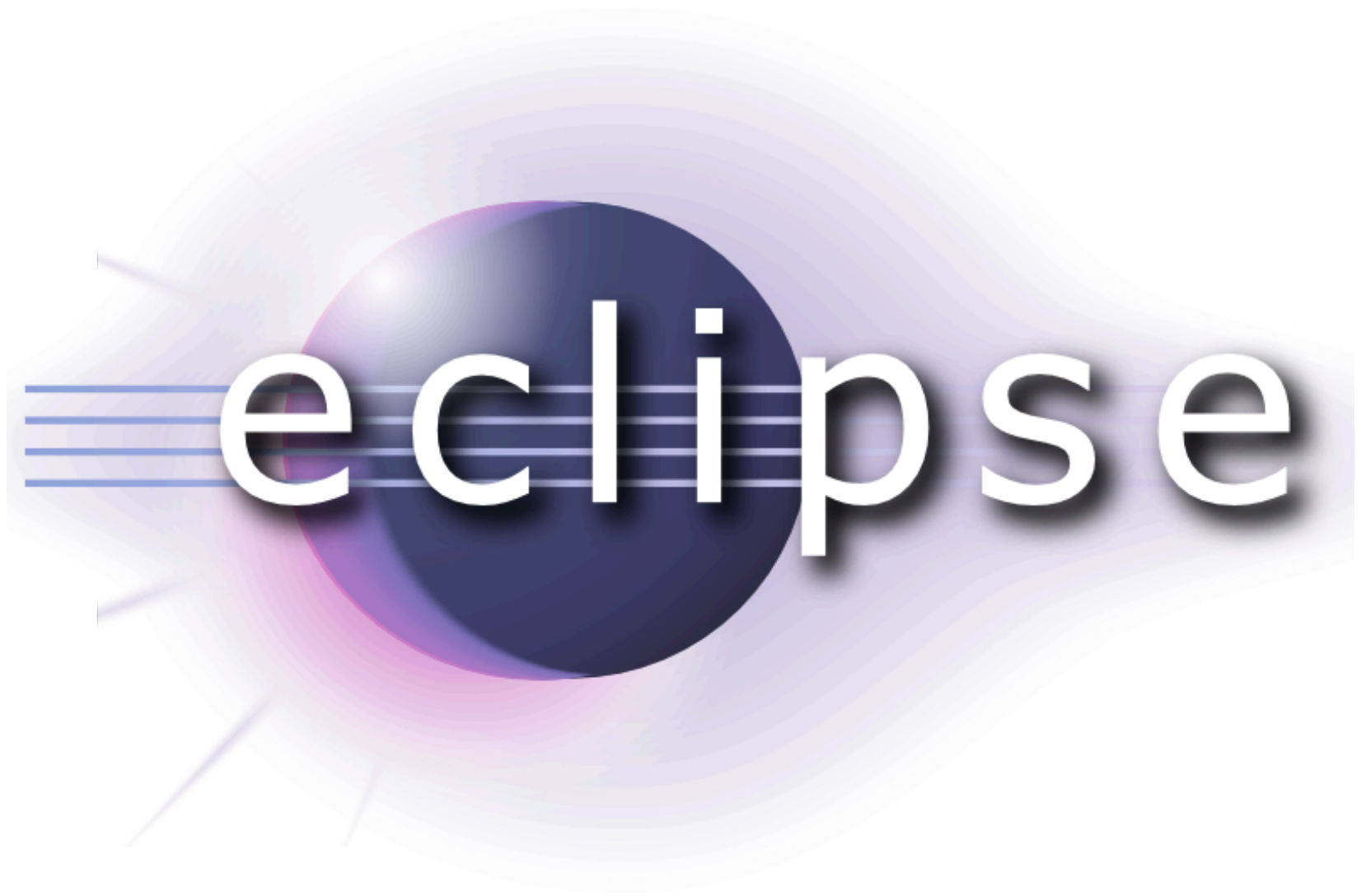
```
move();
```

```
pickBeeper();
```

```
turnLeft();
```

```
putBeeper();
```

Let's Try It



`StepUp.java`

Questions?

Improving our Program

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();
        pickBeeper();
        turnLeft();
        move();
        turnLeft();
        turnLeft();
        turnLeft();
        move();
        putBeeper();
        move();
    }
}
```

It's "Turn Right", but...

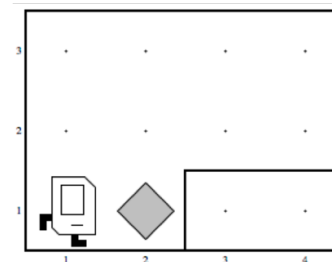


not very clear

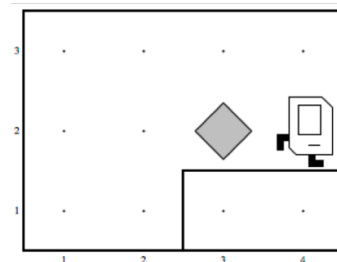


tedious

Before



After



Improving our Program

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();
        pickBeeper();
        turnLeft();
        move();

        // turns Karel right
        turnLeft();
        turnLeft();
        turnLeft();

        move();
        putBeeper();
        move();
    }
}
```

It's "Turn Right", but...
descriptive comments
😞 tedious

Anatomy of a Program

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();
        pickBeeper();
        turnLeft();
        move();

        // turns Karel right
        turnLeft();
        turnLeft();
        turnLeft();

        move();
        putBeeper();
        move();
    }
}
```

This is the
program's
source code

Anatomy of a Program

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();
        pickBeeper();
        turnLeft();
        move();

        // turns Karel right
        turnLeft();
        turnLeft();
        turnLeft();

        move();
        putBeeper();
        move();
    }
}
```

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

Anatomy of a Program

```
import stanford.karel.*;
```

```
public class StepUp extends Karel {
```

```
    public void run() {
```

```
        move();  
        pickBeeper();  
        turnLeft();  
        move();
```

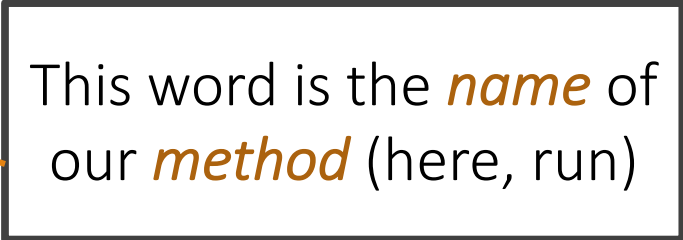
```
        // turns Karel right
```

```
        turnLeft();  
        turnLeft();  
        turnLeft();
```

```
        move();  
        putBeeper();  
        move();
```

```
    }
```

```
}
```



This word is the *name* of our *method* (here, run)

We must always have a run() method to run our program.

But we can make **any other methods we want!**

Method Definition

```
private void name() {  
    statements in the method body  
}
```



This adds a new
command to Karel's
vocabulary

Improving our Program

```
import stanford.karel.*;
```

```
public class StepUp extends Karel {
```

```
    public void run() {
```

```
        move();
```

```
        pickBeeper();
```

```
        turnLeft();
```

```
        move();
```

```
        turnRight();
```

```
        move();
```

```
        putBeeper();
```

```
        move();
```

```
    }
```

```
    private void turnRight() {
```

```
        turnLeft();
```

```
        turnLeft();
```

```
        turnLeft();
```

```
    }
```

```
}
```



Call our turnRight *method*



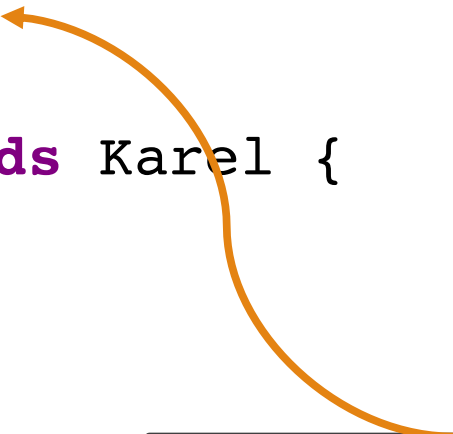
Define a turnRight *method*

Anatomy of a Program

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();
        pickBeeper();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```



This is called an **import statement**.
It tells Java what Karel is.

Anatomy of a Program

```
import stanford.karel.*;
```

```
public class StepUp extends Karel {
```

```
    public void run() {
```

```
        move();
```

```
        pickBeeper();
```

```
        turnLeft();
```

```
        move();
```

```
        turnRight();
```

```
        move();
```

```
        putBeeper();
```

```
        move();
```

```
    }
```

```
    private void turnRight() {
```

```
        turnLeft();
```

```
        turnLeft();
```

```
        turnLeft();
```

```
    }
```

```
}
```

This is called a
code block.

Program Style

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();

        pickBeeper();

        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```

These are
terrifying!




Program Style

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();

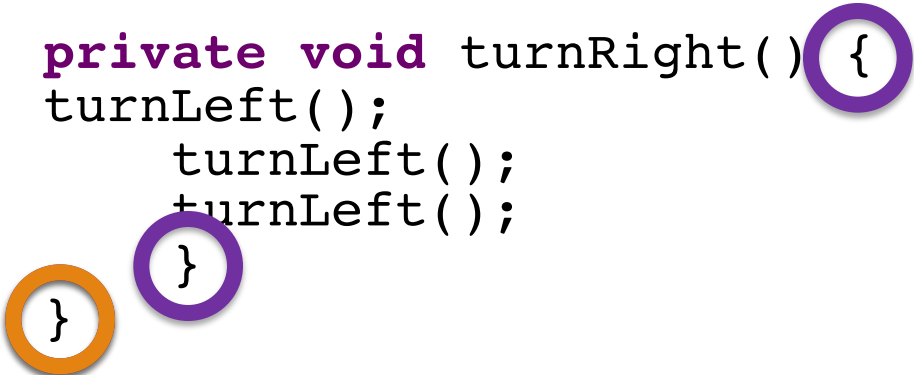
        pickBeeper();

        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }
}
```



Style Tip #1:
Align braces in
your code blocks.

```
private void turnRight() {
    turnLeft();
    turnLeft();
    turnLeft();
}
```



Program Style

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();

        pickBeeper();

        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```

Style Tip #1:
Align braces in
your code blocks.

Program Style

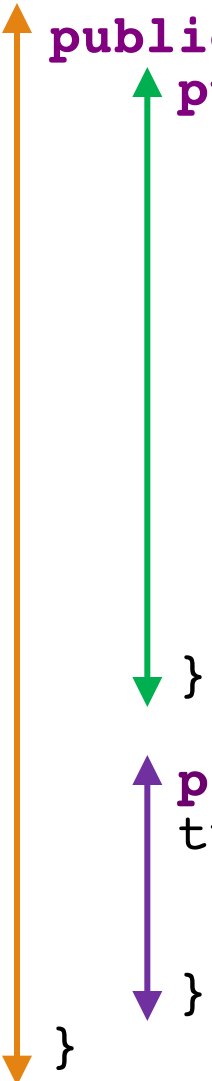
```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();

        pickBeeper();

        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```



Style Tip #1:
Align braces in
your code blocks.

Program Style

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();

        pickBeeper();

        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```

Style Tip #2:
Align indentation
in your code blocks.

Program Style

```
import stanford.karel.*;

public class StepUp extends Karel {
    public void run() {
        move();

        pickBeeper();

        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

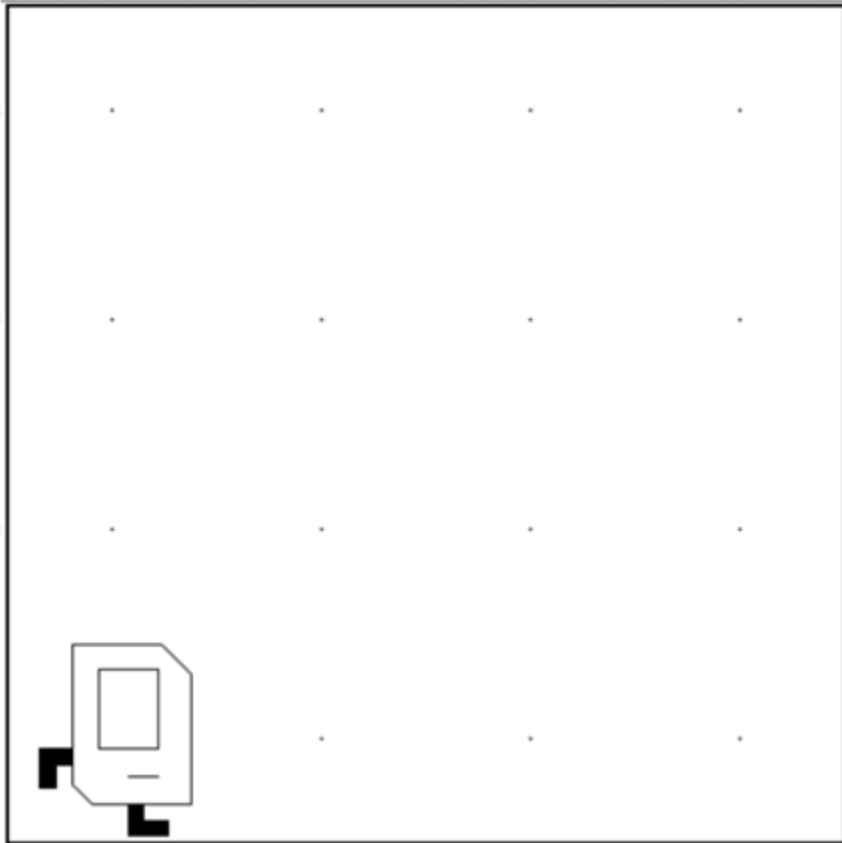
    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```

Style Tip #2:
Align indentation
in your code blocks.

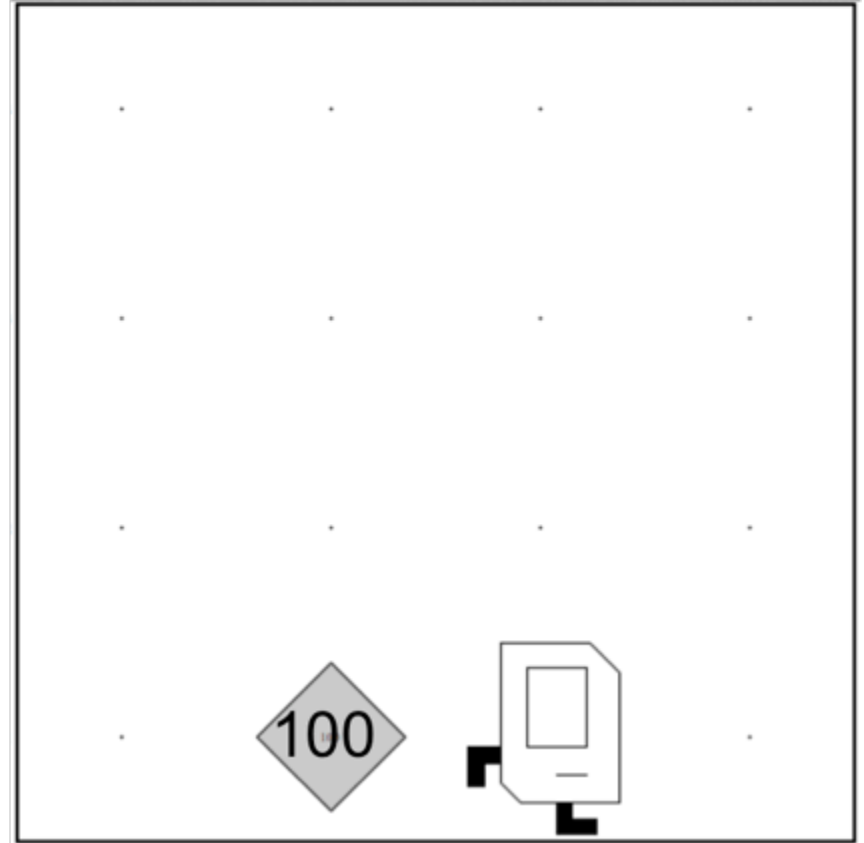
Questions?

Place 100 beeper?

Before



After

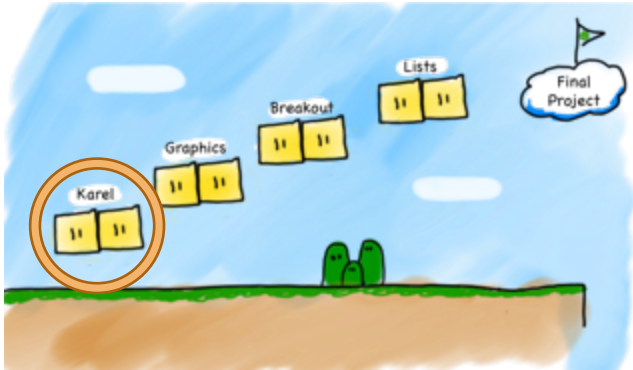


Place100.java

For Loop

```
for(int i = 0; i < N; i++) {  
    // to repeat N times  
}
```

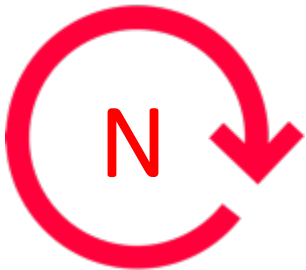
Review



```
move();  
turnLeft();  
putBeeper();  
pickBeeper();
```

Helper *methods* for defining new commands

```
private void name() {  
    statements in the method body  
}
```



for loops repeat code *N* times

```
for(int i = 0; i < N; i++) {  
    // to repeat N times  
}
```

Today's Examples on the Website

CS Bridge

Handouts ▾

Projects ▾

Examples ▾

Slides ▾

Bonus ▾

Forms ▾



Step Up

Based on a handout by Eric Roberts

Handouts: [Karel Reference](#)

Day1: [StepUp.java](#)

Step Up

Place 100

Beeper Line



Invert

UN Karel

```
/**
 * Program: Place100
 * -----
 * This program makes karel place a pile of 100 beepers. Good times.
 */
public class Place100 extends SuperKarel {

    public void run() {
        move();
        // This for-loop will repeat the code inside 100 times.
        for(int i = 0; i < 100; i++) {
            putBeeper();
        }
        move();
    }
}
```


Today's Slides on the Website

CS Bridge Handouts ▾ Projects ▾ Examples ▾ Slides ▾ Bonus ▾ Forms ▾  



Intro to Computer Science

Summer 2019

June 24th to July 4th at Koç University, Istanbul

Karel

ControlFlow

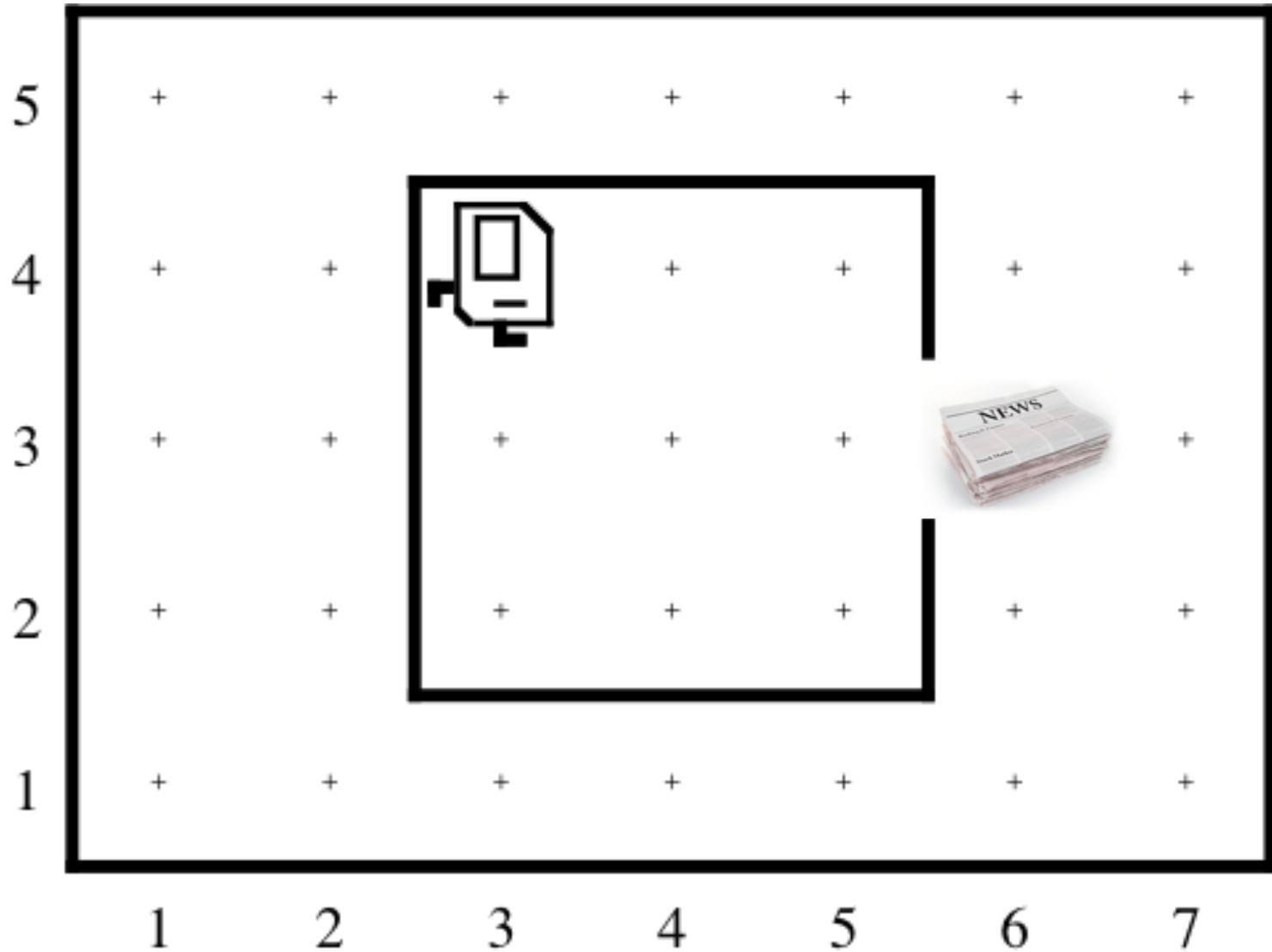
All necessary links (to sign into lab, to submit your work) can now be found under the 'Forms' tab in the top toolbar.

Your turn



x 200
(in Labs)

Newspaper Karel



Your turn



To the
Labs!!



Build Efes

