Internet of Things
Pachube as Example

TU/e Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

Internet of Things

- The Internet of Things refers to uniquely identifiable objects (things) and their virtual representations in an Internet-like structure

Pachube

- **Pachube ("patch-bay") connects people to devices, applications, and the Internet of Things. As a web-based service built to manage the world's real-time data, Pachube gives people the power to share, collaborate, and make use of information generated from the world around them.**

Pachube in Processing

- **Download and install EEML library:**
  - **http://www.eeml.org/library/**

Pachube

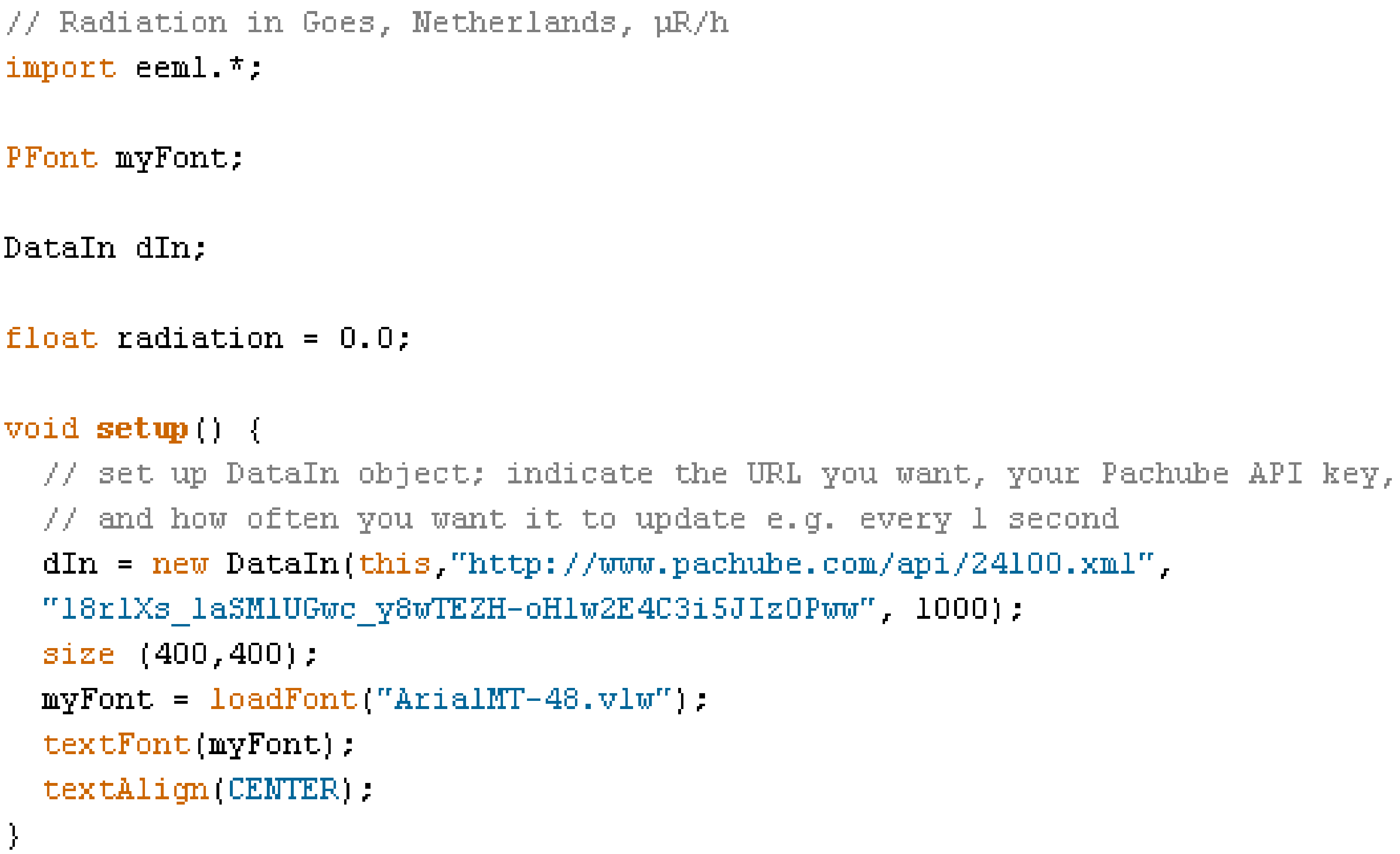
- First sign up.
- Find your own application key.

Pachube

- **Let’s first sneak into somebody else’s sensors.**
- **Search for “Goes”, find the feed number**
  - **Radiation in Goes, Netherlands, µR/h**

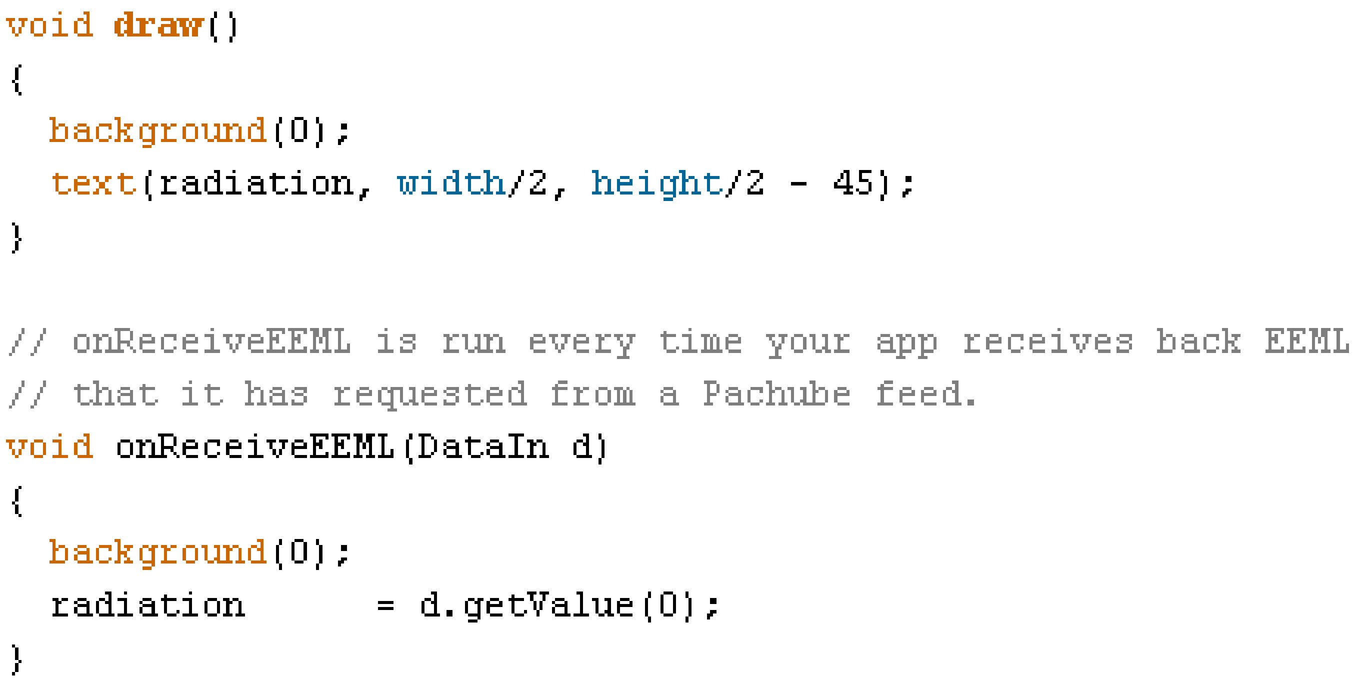
Map
Satellite
Ireland
Dublin
Leeds
Preston
Hull
Manchester
Leicester
Birmingham
Norwich
Ipswich
Northampton
Swansea
Barnet
Basildon
Cardiff
Bristol
Exeter
Portsmouth
Plymouth
Eastbourne
Amsterdam
Netherlands
Breda
Eindhoven
Bruges
Lille
Belgium
Liege
Charleroi
Amiens
Paris
Reims
Caen
Celtic Sea
Luxembourg
Metz
Cuxhaven
Bremerhaven
Hamburg
Schwerin
Groningen
Bremen
Luneburg
Osnabrück
Hanover
Berlin
Bielefeld
Magdeburg
Paderborn
Göttingen
Leipzig
Germany
Cologne
Dresden
Chemnitz
Frankfurt
Würzburg
Nuremberg
Mannheim
Czech Republic
Szczecin
Bydgoszcz
Torun
Poznan
Poland
Zielona Góra
Kalisz
Lodz
Wroclaw
Opole
Walbrzych
Katowice
Rybnik
Olomouc
Brno
Gdansk
Google
Map data ©2011 Europa Technologies, GIS Innovatsia, Geocentre Consulting, Google, PPWK, Tele Atlas, Transnavicom - Terms of Use

Radiation in Goes, Netherlands, µR/h



```
// Radiation in Goes, Netherlands, µR/h
import eeml.*;

PFont myFont;

DataIn dIn;

float radiation = 0.0;

void setup() {
  // set up DataIn object; indicate the URL you want, your Pachube API key,
  // and how often you want it to update e.g. every 1 second
  dIn = new DataIn(this,"http://www.pachube.com/api/24100.xml",
  "18r1Xs_laSM1UGwc_y8wTEZH-oHlw2E4C3i5JIz0Pww", 1000);
  size (400,400);
  myFont = loadFont("ArialMT-48.vlw");
  textFont(myFont);
  textAlign(CENTER);
}
```

Radiation in Goes, Netherlands, µR/h



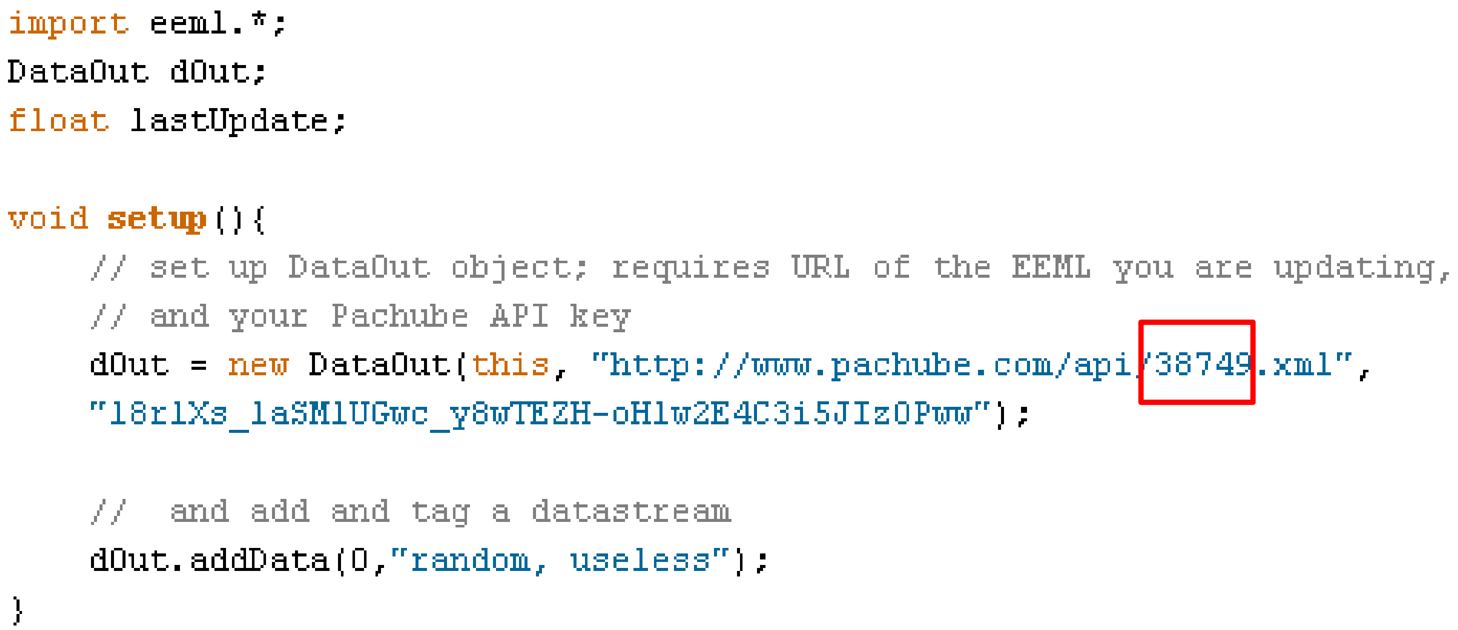
```
void draw()
{
  background(0);
  text(radiation, width/2, height/2 - 45);
}

// onReceiveEEML is run every time your app receives back EEML
// that it has requested from a Pachube feed.
void onReceiveEEML(DataIn d)
{
  background(0);
  radiation      = d.getValue(0);
}
```

My Feed

- Now try to create your own feed.
- Remember your feed number.

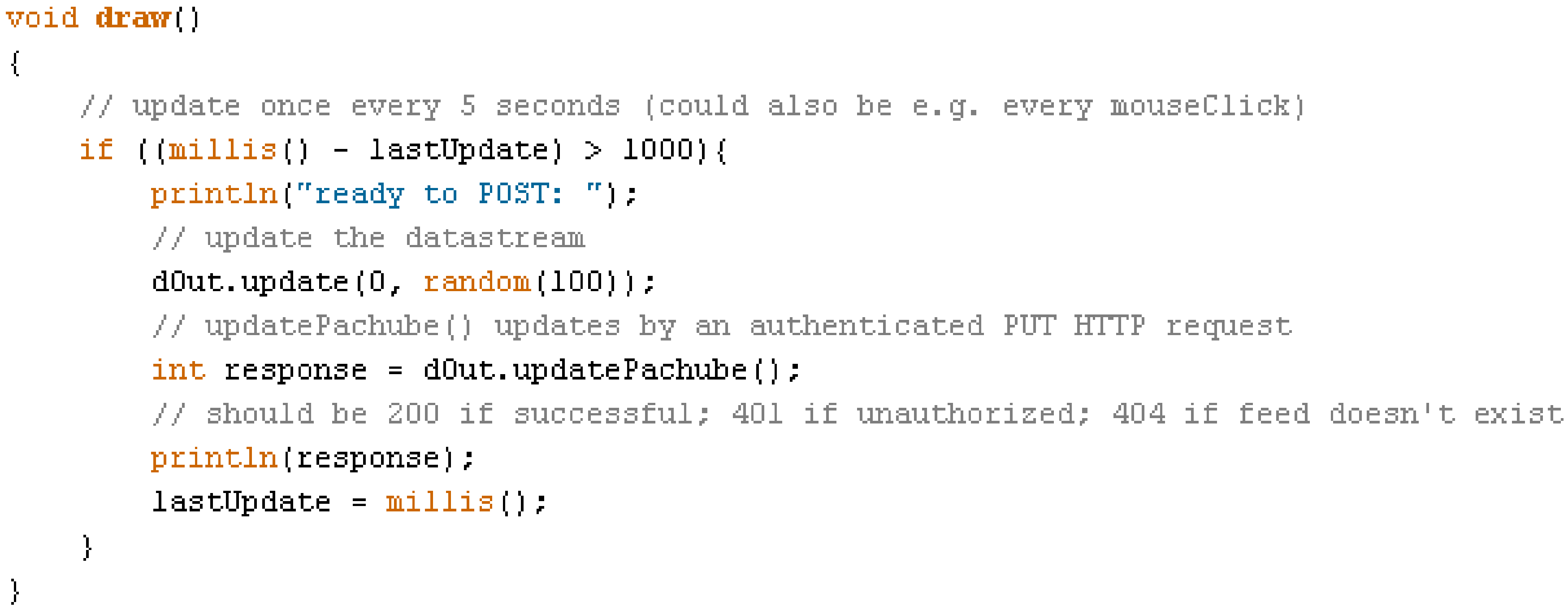
Technische Universiteit
Eindhoven
University of Technology

My Feed – Random Update



```
import eeml.*;
DataOut dOut;
float lastUpdate;

void setup(){
    // set up DataOut object; requires URL of the EEML you are updating,
    // and your Pachube API key
    dOut = new DataOut(this, "http://www.pachube.com/api/38749.xml",
    "l8rlXs_laSMlUGwc_y8wTEZH-oHlw2E4C3i5JIzOPww");

    //  and add and tag a datastream
    dOut.addData(0,"random, useless");
}
```

My Feed – Random Update



```
void draw()
{
    // update once every 5 seconds (could also be e.g. every mouseClick)
    if ((millis() - lastUpdate) > 1000){
        println("ready to POST: ");
        // update the datastream
        dOut.update(0, random(100));
        // updatePachube() updates by an authenticated PUT HTTP request
        int response = dOut.updatePachube();
        // should be 200 if successful; 401 if unauthorized; 404 if feed doesn't exist
        println(response);
        lastUpdate = millis();
    }
}
```

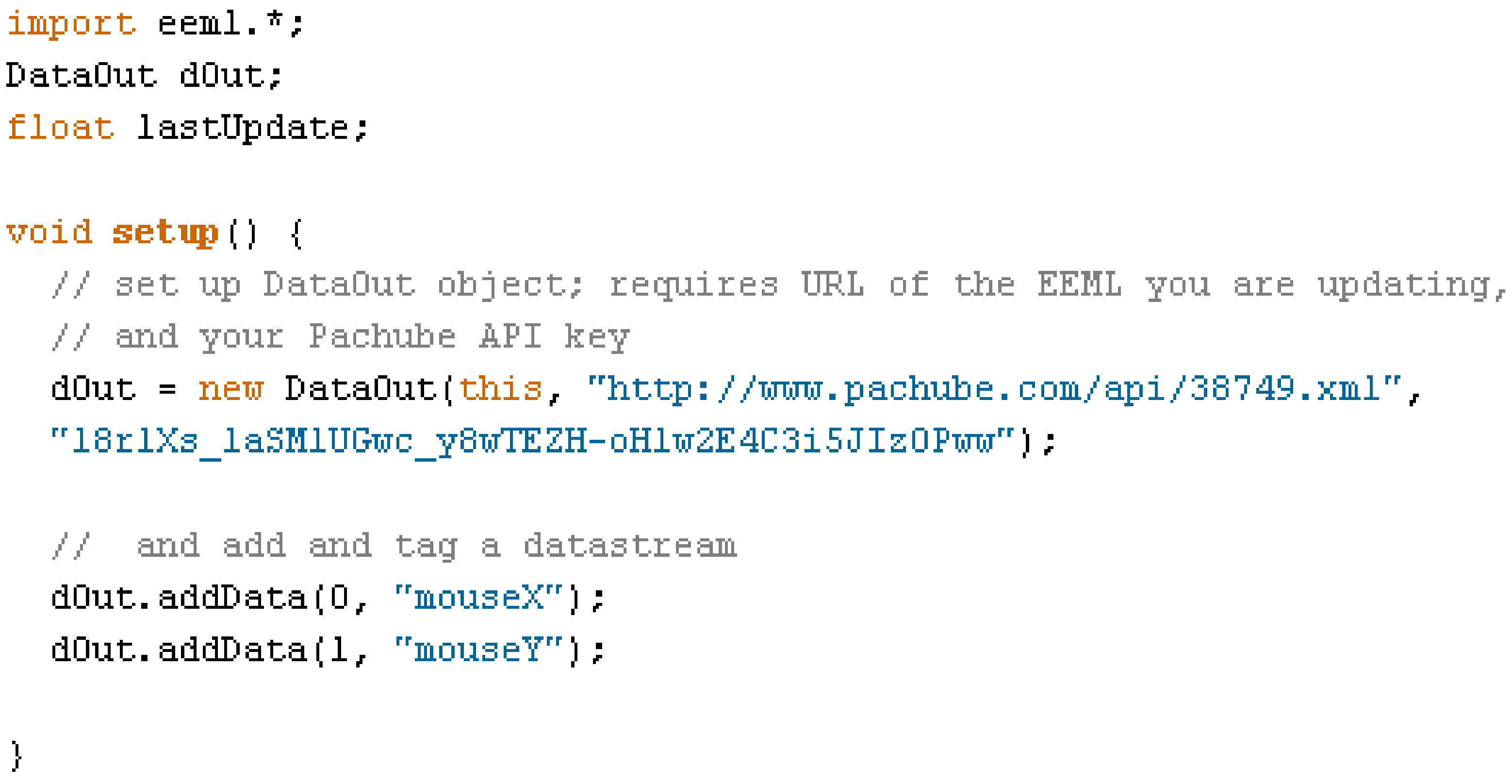
My Feed – Random Update

- **Now try to pull data from Pachube.**

My Feed – Mouse Input

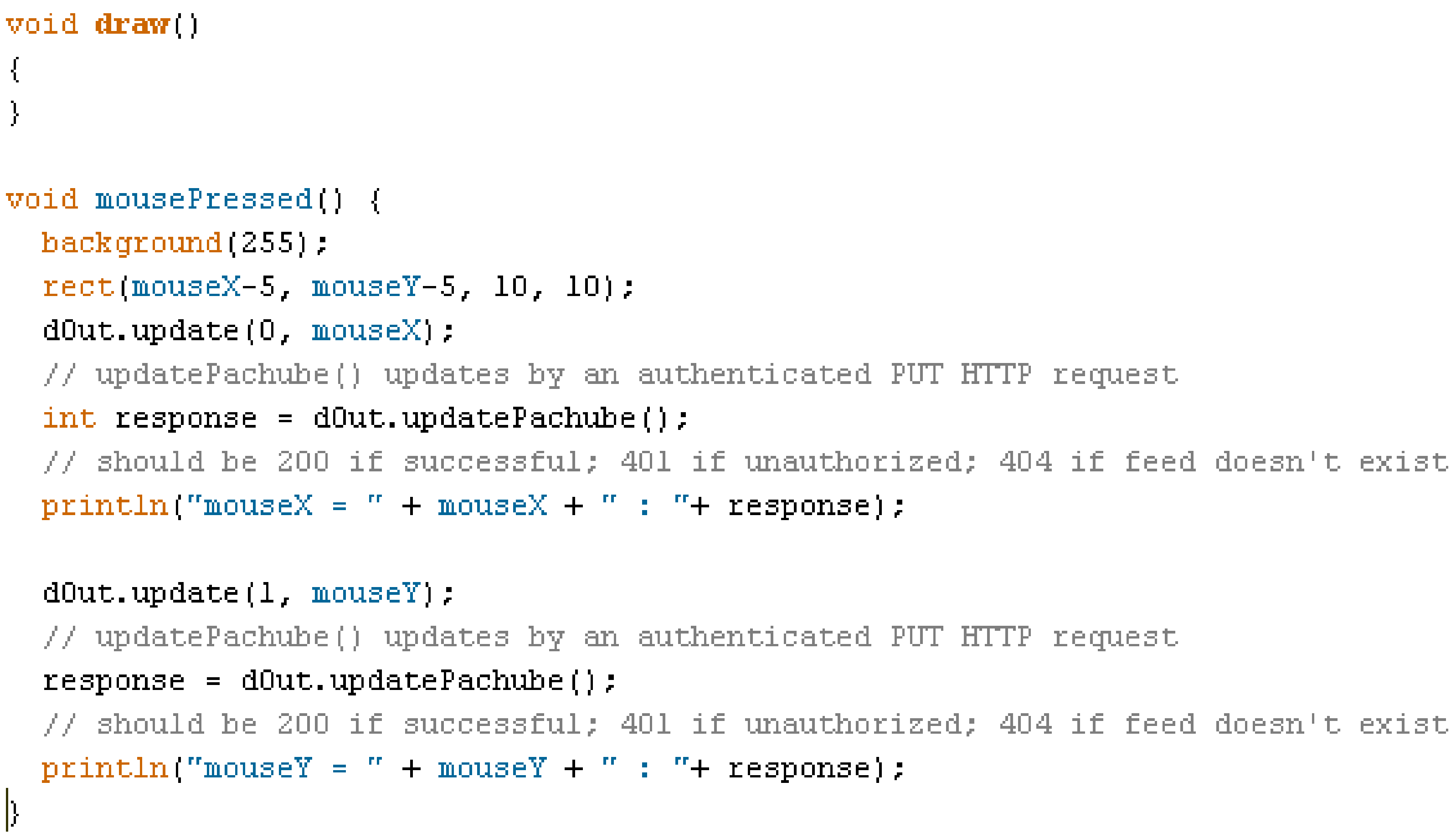
- **Let's try to feed from mouse positions**

My Feed – Mouse Input



```
import eeml.*;
DataOut dOut;
float lastUpdate;

void setup() {
  // set up DataOut object; requires URL of the EEML you are updating,
  // and your Pachube API key
  dOut = new DataOut(this, "http://www.pachube.com/api/38749.xml",
  "l8rlXs_laSMlUGwc_y8wTEZH-oHlw2E4C3i5JIzOPww");

  //  and add and tag a datastream
  dOut.addData(0, "mouseX");
  dOut.addData(1, "mouseY");

}
```

My Feed – Mouse Input

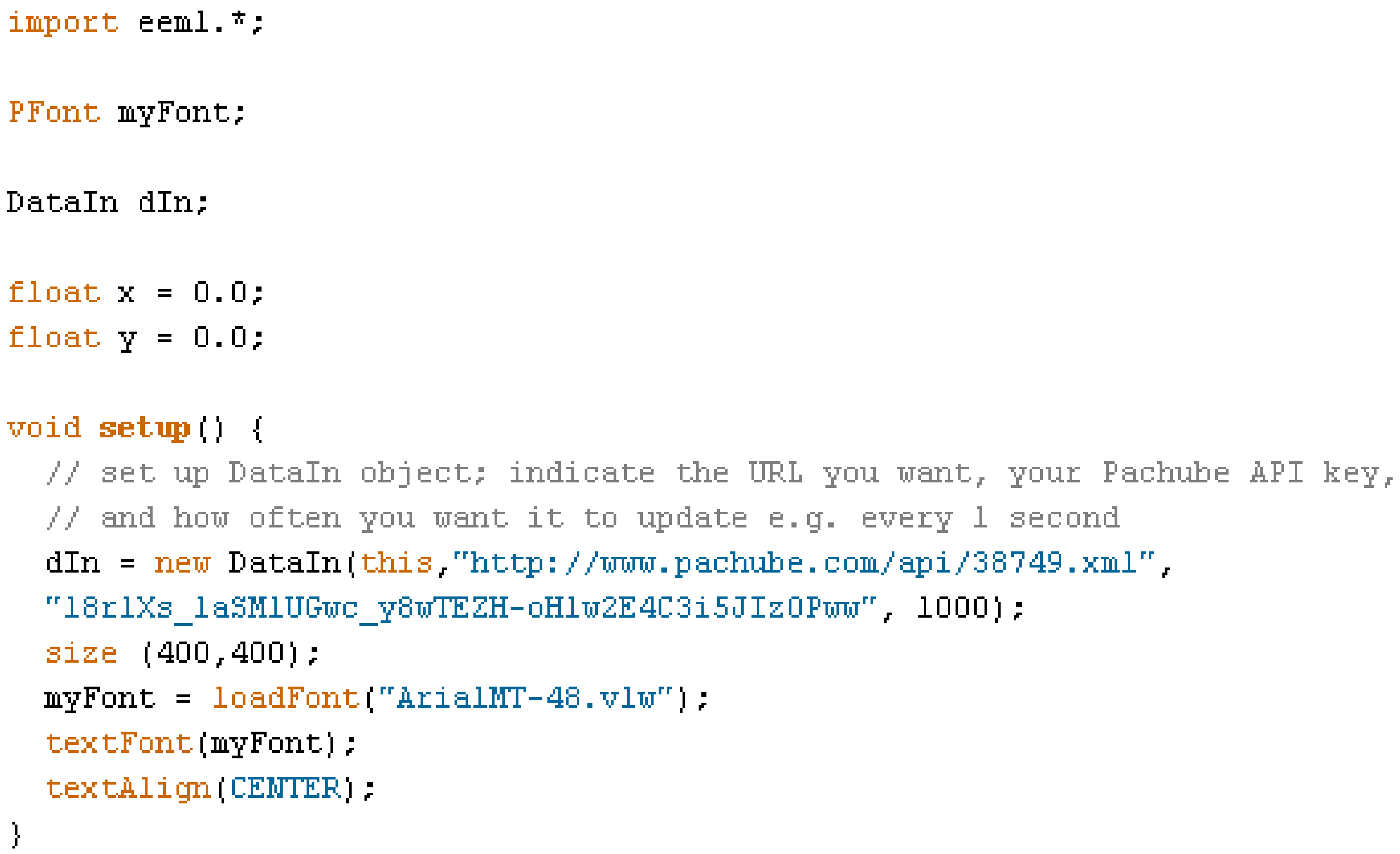


```
void draw()
{
}

void mousePressed() {
  background(255);
  rect(mouseX-5, mouseY-5, 10, 10);
  dOut.update(0, mouseX);
  // updatePachube() updates by an authenticated PUT HTTP request
  int response = dOut.updatePachube();
  // should be 200 if successful; 401 if unauthorized; 404 if feed doesn't exist
  println("mouseX = " + mouseX + " : "+ response);

  dOut.update(1, mouseY);
  // updatePachube() updates by an authenticated PUT HTTP request
  response = dOut.updatePachube();
  // should be 200 if successful; 401 if unauthorized; 404 if feed doesn't exist
  println("mouseY = " + mouseY + " : "+ response);
}
```

My Feed – Random Update

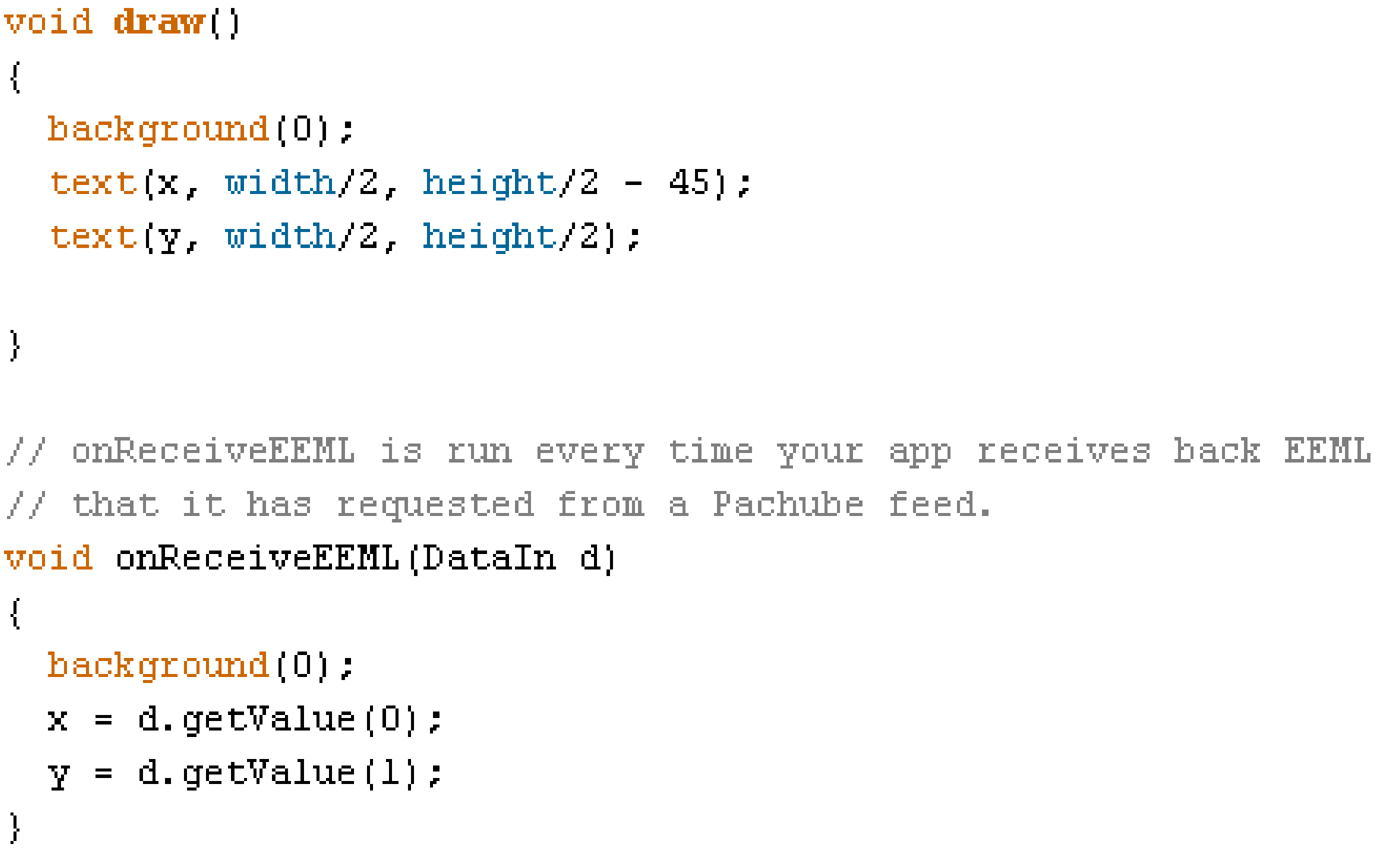
- **Now try to pull data from Pachube.**

My Feed – Mouse Input
Pull from Pachupe



```
import eeml.*;

PFont myFont;

DataIn dIn;

float x = 0.0;
float y = 0.0;

void setup() {
  // set up DataIn object; indicate the URL you want, your Pachube API key,
  // and how often you want it to update e.g. every 1 second
  dIn = new DataIn(this,"http://www.pachube.com/api/38749.xml",
  "l8rlXs_laSMlUGwc_y8wTEZH-oHlw2E4C3i5JIzOPww", 1000);
  size (400,400);
  myFont = loadFont("ArialMT-48.vlw");
  textFont(myFont);
  textAlign(CENTER);
}
```

My Feed – Mouse Input
Pull from Pachupe



```
void draw()
{
  background(0);
  text(x, width/2, height/2 - 45);
  text(y, width/2, height/2);

}

// onReceiveEEML is run every time your app receives back EEML
// that it has requested from a Pachube feed.
void onReceiveEEML(DataIn d)
{
  background(0);
  x = d.getValue(0);
  y = d.getValue(1);
}
```

Now it is your turn

- To integrate
  - Processing
  - Arduino
  - Network
  - Pachupe
  - ……