

---

**ITEC 120**

Lecture 28  
Blending variables and functions

---

**Review**

- **Classes**
  - Getters / setters
  - Public / private

---

Class variables / functions

**Objectives**

- Learn blending of functions and variables
- Examples

---

Class variables / functions

**Goal**

- Learn how to make objects semi-sentient
- Mix functions and variables
- **Before**
  - Functions took data and returned / printed results
- **Current**
  - Functions initialize values, make a copy, or keep it in range
- **Next**
  - The ability to use fields / class variables

---

Class variables / functions

## Rationale

- Move responsibility away from main
- Delegating
- Tell what, not how to do task X

Class variables / functions

## Example

```
public class example
{
    int cost=0;
    public example()
    {
        cost=4;
    }
    public void setCost(int num)
    { if (cost > 0) { cost = num; } }
    public int getCost()
    { return cost;}
    public void printHalfOff()
    {
        System.out.println("The Half off cost is " + cost/2);
    }
}
Example test = new example();
test.printHalfOff();
```

Class variables / functions

## Mixing

```
public class Pokemon
{
    public String name;
    public String type;
    public int HP;
    public int damage;
}

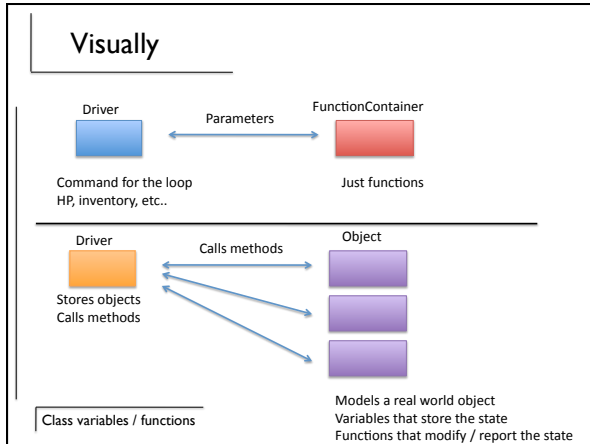
public class MixedPokemon
{
    public String name;
    public String type;
    public int HP;
    public int damage;
    public void printInfo()
    {
        System.out.println("Hello, my name is " + this.name);
        if (type.equals("fire"))
            System.out.println("Don't touch, I'm hot stuff");
    }
}
```

Class variables / functions

## Power

- Class variables can be used by any function in the class
  - Changes where information lives
- Another reason for multiple function containers

Class variables / functions



### Example

- Item in a virtual store

```

public void Item
{
    public String name;
    public String description;
    public double price;
    public double getWholesalePrice()
    {return price*.70; }
    public double getRegularSalePrice()
    {return price*.9; }
    public double getClearancePrice()
    {return getWholesalePrice() / 2.0; }
}
    
```

Three arrows point to the code: '30% markup on price' points to the getWholesalePrice() method, '10% off' points to the getRegularSalePrice() method, and 'Half of the wholesale price' points to the getClearancePrice() method. A legend at the bottom left indicates that blue boxes represent 'Class variables / functions'.

How do you create an item and set the values?

### Workers

```

public class Employee
{
    public Employee(String ename)
    {name = ename;}

    public String name;
    public String title;
    public double salary;
    private Employee() { }
}
    
```

Employee a = new Employee();  
 Employee b = new Employee("Kyle");  
 b.salary = 290299;  
 b.title = "President";  
 System.out.println(b.title + " " + b.name  
 + " makes " + b.salary + " per year");

Won't work, private

Class variables / functions

### Expanding

```

public class Employee
{
    //Include code from the previous slide here
    public String name;
    public String title;
    public double salary;
    public double monthlyGross()
    {
        return salary/12.0;
    }
    public double payCheck()
    {
        return monthlyGross()**.70;
    }
    public double bonus()
    {
        return payCheck() * .10;
    }
}
    
```

30% flat tax

10% of their paycheck

Class variables / functions

### NPC example

- Would you like your enemies regular or extra crispy?
- Mind if I turn up the heat?
- Who turned on the air conditioning, I hate air conditioning?



Class variables / functions

### Code

```

public class FireNPC
{
    String[] msgs;
    int num;
    public class FireNPC()
    {
        num=0;
        msgs = new String[3];
        msgs[0] = "Would you like your enemies regular or extra crispy?";
        msgs[1] = "Mind if I turn up the heat?";
        msgs[2] = "Who turned on the air conditioning, I hate air conditioning";
    }
    public String talk()
    {
        num++;
        if (num ==3)
            num=0;
        return msgs[num];
    }
}
    
```

```

FireNPC burn = new FireNPC();
System.out.println(burn.talk());
System.out.println(burn.talk());
System.out.println(burn.talk());
System.out.println(burn.talk());
    
```

Class variables / functions

### Arrays

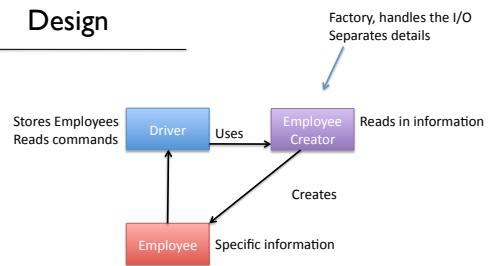
- Can have arrays of objects (1D/ 2D)

```

class Apple
{
    public int number;
    public String type;
    public String review;
    public int calories;
}
Apple[] lineOfApples = new Apple[1000];
Apple[][] gridOfApples = new Apple[100][100];
    
```

Class variables / functions

### Design



Class variables / functions

## Creator

```
public class ECreator
{
    public Scanner scan;
    public Employee createEmployee()
    {
        Employee alpha = new Employee();
        alpha.name = scan.next();
        alpha.title = scan.next();
        alpha.salary = scan.nextDouble();
        return alpha;
    }
}
```

Class variables / functions

## Usage

```
public static void main(String[] args)
{
    ECreator factory = new ECreator();
    Scanner scan = new Scanner(System.in);
    factory.scan = scan;
    String command=scan.nextLine();
    Employee[] employees = new Employee[100];
    int end=0;
    while (!command.equals("quit"))
    {
        if (command.equals("addEmployee"))
        {
            employees[end] = factory.createEmployee();
            end++;
        }
        command = scan.nextLine();
    }
}
```

How would we figure out how much giving a 3% raise to all our employees costs?

Class variables / functions

## Summary

- Functions can use class variables
- Functions can be the middleman between variables

Class variables / functions