

---

**ITEC 120**

Dr. Ray  
Lecture I  
Introduction to Principles of CS I

---

**Objectives**

- Introduce each other
- Introduce you to the course
- Content
- Expectations
- Semester agenda
- Simple problem

Introduction

Fill out forms

**Questions**

- Why are you taking this course?
- What do you want to learn?
- I thing software does that improves life
- Introduce each other to the class
  - Answers to questions / list of activities

Introduction

**Expectations**



Introduction


What you may want to do...

## Definition

**Principle:**  
The laws or facts of nature underlying the working of an artificial device


**Computer Science:**  
A branch of science that deals with the theory of computation or the design of computers

Currently you are used to using computer software



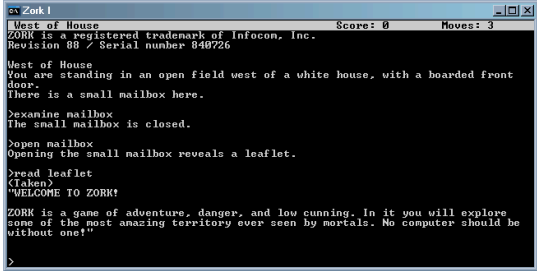
This course teaches you to write software

```
public void Fun()
{ loadGraphics();
  startLevel();
}
```



Introduction m-w.com


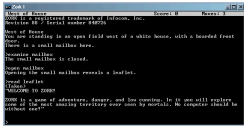
## Reality



Introduction What you are going to do.

## Why...

<p><b>Darkest of days:</b> 24 person team 3.5 years 1.7 million dollars 200,000 lines of code</p>	<p><b>This class</b> 1 person team 14 weeks \$0 1,000 lines of code</p>
---	---


Introduction

## Course introduction

**Information:**  
[www.radford.edu/~aaray](http://www.radford.edu/~aaray)  
Click ITEC 120

**Submitting HW:**  
[learn.radford.edu](http://learn.radford.edu)

- Writing software can be fun
- What you do in ITEC 120 is what professionals do every day
- We are going to use these in ITEC 120



Introduction

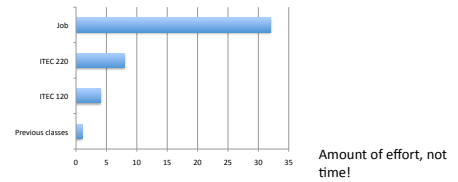
## Structure

- How class time will be used
  - Lecture MWF
    - M => Introduction to topic, simple examples
    - W => Answer Questions, Best practices
    - F => Review, high level lectures, Homework
  - Lab TR
    - Hands on practice
- Outside of class
  - Homework assignments, finishing up labs

Introduction



## Expectations

- 4 times the work of previous classes
  - 5-15 hours per week outside of class



Introduction

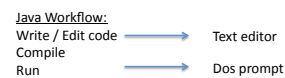
## Language types

- Interpreted 
  - Master program reads lines of codes and executes them
- Compiled 
  - Lines of code are turned into 0s and 1s
  - Executed on the machine

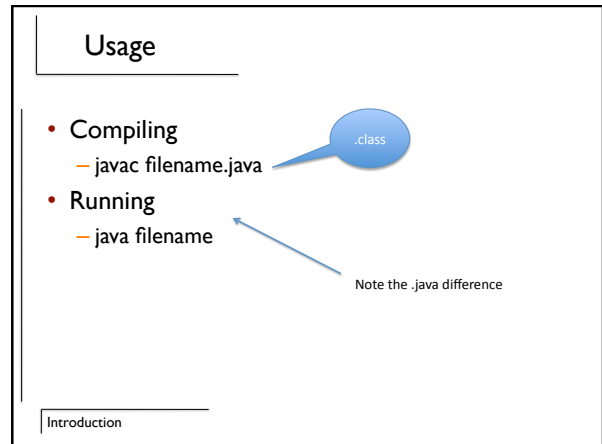
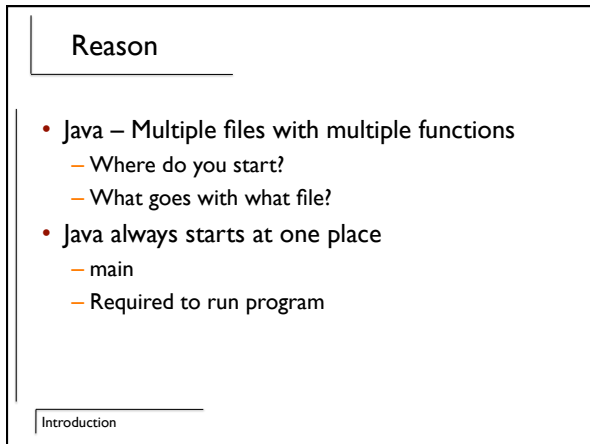
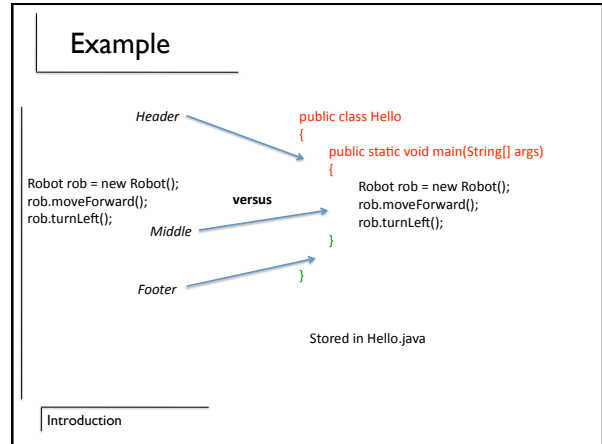
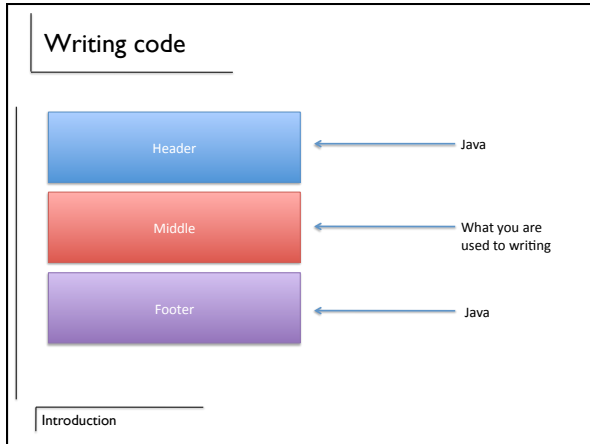
Introduction

## 2 parts

- Interface for writing code
  - jEdit / text editor
- Interface for running code
  - Command prompt



Introduction



## Variables

- `<type> name;`

Must begin with A-Z  
Cannot contain spaces

```
public class Example
{
    public static void main(String[] args)
    {
        int x;
        double y;
        float z;
        String a;
        char b;
    }
}
```

Introduction

## Output

- Java has builtin variables

– System

- Nested variables

– System.out.println();

```
public class Example
{
    public static void main(String[] args)
    {
        int x;
        x=3;
        System.out.println(x);
    }
}
```

Introduction

## Strings

```
public class Example
{
    public static void main(String[] args)
    {
        String a;
        a="Hello World";
        System.out.println(a);
    }
}
```

Need the "" to tell javac  
what to expect

Introduction

## Example

- Hello World in jEdit
- Compile it
- Modify it to handle Fahrenheit to Celsius

Introduction

## Summary

- Class structure
- Expectations
- Basics of Java

Introduction