

### CS 113 – Computer Science I

## Lecture 18 – Relationships & Class Actions

Tuesday 11/14/2024

#### Announcements

HW 09 – Due Monday 11/18 Autograder is up

HW 10 – Due Monday 11/25 Object Oriented Programming: Inheritance & Interface Releasing it tomorrow

Mid-semester feedback form: <u>https://forms.gle/Ed7G9oe74QQBT5sy5</u>

Midterm 2: Thursday December 5<sup>th</sup>

# Inheritance: feature for organizing classes into hierarchies



#### Class inheritance

Classes can be arranged hierarchically where, a child class "inherits" from a parent class

# Inheritance: feature for organizing classes into hierarchies



#### Inheritance: subclasses refine behavior/state

Subclasses can override methods from parent class

### Polymorphism

Program can treat all objects that extend a base class the same

Java automatically calls the specific methods for each subclass

### Polymorphism: Demo

public class Zoo {

```
public static void main(String[] args) {
    Animal animal1 = new Animal();
    animal1.locomote();
```

```
Animal animal2 = new Reptile();
animal2.locomote();
```

```
public class Animal {
    public Animal() {
    }
    public void locomote() {
        System.out.println("I am moving!");
    }
}
```

public class Reptile extends Animal {
 public Reptile() {

```
public void locomote() {
   System.out.println("I am walking!");
```

### Exercise: What is the output of this program?

public class Zoo {

public static void main(String[] args) {
 Animal animal1 = new Animal();
 animal1.locomote();

```
Animal animal2 = new Fish();
animal2.locomote();
```

public class Animal {
 public Animal() {
 }
 public void locomote() {
 System.out.println("I am moving!");
}

public class Fish extends Animal {
 public Fish() {

public void locomote() {
 System.out.println("I am swimming!");

#### Question: How would we implement Minion?





#### Inheritance: subclasses refine behavior/state

Subclasses can override methods from parent class

```
class Animal {
    public Animal(String name, boolean hasHair,
        int numberLegs, boolean swimable) {
        this.hasHair = hasHair;
        this.numberLegs = numberLegs;
        this.name = name;
        this.swimable = swimable;
    }
```

```
public class Fish extends Animal {
```

#### Inheritance: constructors - super();

super();

reference variable that is used to refer parent class constructor

#### Inheritance: subclasses refine behavior/state

Subclasses can override methods from parent class

```
class Animal {
    public Animal(String name, boolean hasHair,
        int numberLegs, boolean swimable) {
        this.hasHair = hasHair;
        this.numberLegs = numberLegs;
        this.name = name;
        this.swimable = swimable;
    }
}
```

```
public class Fish extends Animal {
```

#### Inheritance: constructors - super();

class Animal {

```
public class Fish extends Animal {
```

public class Fish extends Animal {

#### Inheritance: constructors - super();

super();

reference variable that is used to refer parent class constructors

Note:

super:

reference variable that is used to refer parent class object

# Inheritance: feature for organizing classes into hierarchies



#### interfaces

A common set of methods that each implementing class must include (like a blueprint)

*Contract* for a class to implement a certain set of methods

Implementing class *inherits* a list of functions from the interface

methods in an interface are abstract

- declared method without an implementation
- contains just method signature

Define an interface using the interface keyword

#### Implementing an interface

- 1. Use implements keyword instead of extends (demo)
- 2. Implement the functions

## Inheritance vs Extends

Interfaces (subtyping)

- implements
- Guarantees same types have same functions
  - Though the same functions are implemented differently

Inheritance (subclassing)

• extends

- Reuses implementations
- Consequences:
  - Dependent on base class
  - Changes in superclass affects all subclasses
  - Can re-use code inside classes

- A class can implement multiple interfaces
- An interface can extend another interface

 A class can extend just one parent class