CSE 11
Accelerated Intro to Programming
Discussion Section 10

Ayon Biswas, Spring 2021

This discussion is being recorded

Logistics

- PA9 due tomorrow at 11:59PM
- Final Exam
  - Release: 8am Wednesday June 9, 2021
  - Due: 11:59pm Friday June 11, 2021

Overriding methods

- providing a different implementation of an existing method
- the method's header must be identical to the method in the superclass. The body then can be different

```
// Base Class
class Parent {

    void show()

    {

        System.out.println("Parent's class()");

    }

}

  // Inherited class

class Child extends Parent {

    // This method overrides show() of Parent

    @Override

    void show()

    {

        System.out.println("Child's class()");

    }

}
```

The instanceof operator

- It is used between an object and the name of a class, and returns true if that object's type is equal to or is a subclass of that class.

```
class A {}
class B extends A {}
class C extends A {}

A a = new A();
B b = new B();
C c = new C();

boolean isAAnInstanceOfA = a instanceof A; // true
boolean isBAnInstanceOfA = b instanceof A; // true
boolean isBAnInstanceOfB = b instanceof B; // true
```

Casting

- To treat an instance as having the type of the superclass
- hiding any fields or methods that is only defined on the subclass.

```
class A {}

class B extends A {

    int x;


    B(int x) {

        this.x = x;

    }

}

A a = new B(1);

// error: cannot find symbol

int error = a.x;

// This works, and assigns 1 to x

B b = (B) a;

int x = b.x;
```

Access Modifiers

The public, protected and private access modifiers to clearly indicate the access to different classes, fields and methods.

- The public modifier allows access anywhere.
- The protected modifier allows access in any of the subclasses of the protected class.
- The private modifier only allows access within the class that contains them.

Final Exam

- Similar to Exam 1 and 2 (programming + video)
- Cumulative
- Read instructions carefully
- Follow instructions closely

Thanks!