



THE UNIVERSITY OF
MELBOURNE

SWEN20003 Workshop

Week 03

Demonstrator: Xulin Yang





Workshop tasks 🙄

1. Doing some concepts review (as much as I can in 20 mins, if not you can do it in your own time)
2. Demonstrate Q1 (10 mins)
3. Divide student to groups
4. Do workshop Q2-Q4 (60 mins) + take attendance
5. Go through Q2-Q4 (30 mins)



What is the purpose of a constructor, and how do we “invoke” them?



What is the purpose of a constructor, and how do we “invoke” them?

Constructor: A method used to create and initialize an object.

Constructors literally “construct” an object by initializing its values. We invoke a constructor using the **new** keyword

```
Circle aCircle = new Circle();
```



Explain the difference between a class and an object and an instances



Explain the difference between a class and an object and an instances

Class = a “blueprint” setting out the data associated with a type of object (attributes), and the actions the object can perform (methods).

-> something with **common** property (attribute) and behavior (method)

= Fundamental unit of abstraction in Object Oriented Programming.

= Represents an “entity” that is part of a problem

Object = an instance of a class, containing its own data (separate from other objects of that class).

Instance = An object that is created during a program's execution (exist in your code)

What’s the relationship between them?

an **object** is an **instance** of a **class**



In general, what do instance variables and methods “represent” about a class?



In general, what do instance variables and methods “represent” about a class?

Instance variables represent the properties or attributes of a class

Instance methods represent the actions that can be performed by a class

What if they are not “instance”? See next ...



Why static?

static: Indicates a constant, variable, or method exists **without an object**.

i.e., you do not need to create a variable to use something defined as **static**

You can access static variable/method without instantiate an instance of a class (i.e. no **new**).

Static attributes are shared between all objects of a given class.

When using static variable?

Static methods are not associated with any particular object.

When using static method?



Why static?

static: Indicates a constant, variable, or method exists **without an object**.

i.e., you do not need to create a variable to use something defined as **static**

You can access static variable/method without instantiate an instance of a class (i.e. no **new**).

Static attributes are shared between all objects of a given

class.

When using static variable?

If (one copy per object) {

 instance variable

} else if (one copy per class) {

 static variable

}

Static methods are not associated with any particular object.

When using static method?

If (access instance variable) {

 instance method

} else {

 static method

}



What are some of the “standard” methods we write for a class, and what purpose do they serve?



What are some of the “standard” methods we write for a class, and what purpose do they serve?

- `equals` is used to determine whether two objects are the same
- `toString` is used to create a String representation of an object;
 - useful for displaying information to humans, or recording the contents of an object
- Copy constructor is used to create a copy of an object



THE UNIVERSITY OF
MELBOURNE

Thank you
