Polymorphism!

- Review these help slides!
- You **will** need to use polymorphism!
- It **is** your friend!
cs015.prj.TASafeHouseSupport.GraphicalRoom

- Models an empty room
  - Has methods to add walls, a ceiling and a floor
  - Furniture, LightSources, and GraphicalLightSwitch subclasses need to be added via their respective add... methods

- Constructor:
  - GraphicalRoom(javafx.geometry.Point2D location); Creates a GraphicalRoom at location

- How will you subclass this GraphicalRoom?
  - Hint: a Room class could model a generic room by extending the cs015.prj.TASafeHouseSupport.GraphicalRoom class
  - But even your Room class doesn’t know what type of room it is, so you should subclass Room further!
Room Inheritance

```
cs015.support.prj.TASafeHouseSupport.GraphicalRoom

Room

Bathroom  Bedroom  Kitchen
```
An abstract class that models the basic capabilities of a light switch

Methods:

```java
GraphicalLightSwitch(cs015.prj.TASafeHouseSupport.GraphicalRoom room)
abstract void flickedOn();
abstract void flickedOff();
```

Should you subclass this `GraphicalLightSwitch`?
- Hint: if you look at the handout, `flickedOn()` and `flickedOff()` are abstract and don’t do anything at all. How can you override them to make them do something useful?
- Can you make it communicate with some other class to do something useful?
LightSwitch Inheritance

\[ \text{cs015.prj.TASafeHouseSupport.GraphicalLightSwitch} \]

\[ \text{LightSwitch} \]
LightSwitch vs. Room

Things to think about...

**LightSwitch:**
- Do you want to make a subclass of `GraphicalLightSwitch` for each type of room, or can you make a generic `LightSwitch` that doesn’t care what type of room it is in?
- How can you code a generic `LightSwitch` *without* using conditionals (if-statements)?

**Room:**
- How are you going to differentiate between the Bedroom, Bathroom and Kitchen?
- What do they have in common that can be generalized? What do they need to do differently?

Hint: *Use Polymorphism!!*
Stuff to ponder over breakfast

● When does the `LightSwitch` need to know about the `Room`?
  o `flickedOn();`
  o `flickedOff();`
● Does it need to know what type of room it is?
  o What concept allows us to ignore that question?
● Should you have three `Room` subclasses AND three `LightSwitch` subclasses?

POLYMORPHISM!
How do containers and contained objects communicate?

- **LightSwitch** is graphically contained by your **Room** superclass
- Containment Diagram

  ![Containment Diagram](image)

  - **Room** contains a **LightSwitch**
  - This allows **Room** to send messages (i.e. call methods on) the **LightSwitch**
  - **LightSwitch** has a reference to **Room**, so it can send messages to anything that is a **Room** (Think Polymorphism!!)
Using the **LocationGenerator**

- When we call `getTopLeft()`, it will return an instance of `javafx.geometry.Point2D` which we can then use.
- The **LocationGenerator** is written for you. You don’t need to add *any* functionality to this class. So do you need a subclass of this or just an instance?
- Does every **Room** need to know about **LocationGenerator** or does each room only need a `javafx.geometry.Point2D`?
Containment vs. Inheritance
Inheritance Diagram
- Shows an inheritance hierarchy
- What class a given class is a **subclass** of (and inherits from)
- What classes a given class is a **superclass** of

Containment Diagram
- Shows containment and associations
- **Containment**: what classes contain instances of other classes
- **Associations**: what classes a given class knows about (has a reference to)
Inheritance and containment diagrams must be SEPARATE!

DO NOT make a diagram that attempts to show both. It will be confusing, wrong, and ugly. 😞
Room Inheritance

cs015.support.prj.TASafeHouseSupport.GraphicalRoom

Room

- Bathroom
- Bedroom
- Kitchen
LightSwitch Inheritance

cs015.prj.TASafeHouseSupport.GraphicalLightSwitch

LightSwitch
Note: the Room superclass should contain a LightSwitch, as that is common to all types of Rooms. But will you ever instantiate your Room superclass?
TASafeHouse Containment
GOOD LUCK

On TASafeHouse!!!