Problem statement:

Two cats and three mice are in a house with 10 rooms.

- The cats and mice move from room to room randomly, as long as they are alive.
- When a cat enters a room:
 - o If a mouse is in that room, the cat eats the mouse.
 - Then the cat sleeps in that room for a while.
- When a mouse enters a room:
 - The mouse sometimes leaves immediately,
 - o but otherwise it sleeps in that room for a while.

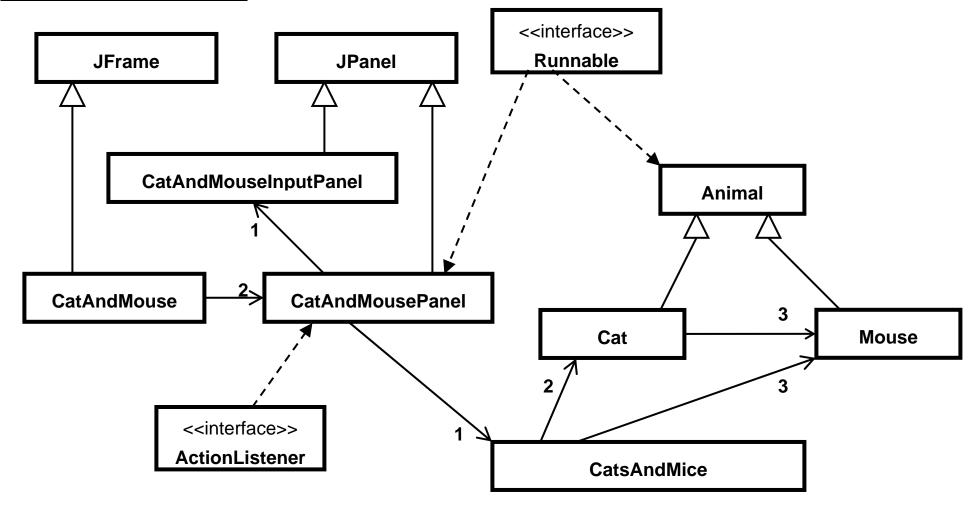
Notes:

- 1. The rooms are numbered from 1 to 10.
- 2. If there is more than one mouse in the room, the cat eats all of them.
- 3. Each mouse has a "skittishness" that is a number between 0 and 1. When a mouse enters a room, it leaves the room immediately with probability per its skittishness. For example, if a mouse's skittishness is 0.2, then with probability 0.2 the mouse immediately leaves the room that it enters and with probability 0.8 it sleeps in that room for a while.
- 4. Each animal has a "timeToSleep" that is the number of milliseconds that the animal sleeps in a room before moving to another randomly chosen room.
- 5. When an animal moves to a new (randomly chosen) room, the new room should really be new (i.e., not the same room in which the animal currently is in).

Problem statement (repeated for your convenience):

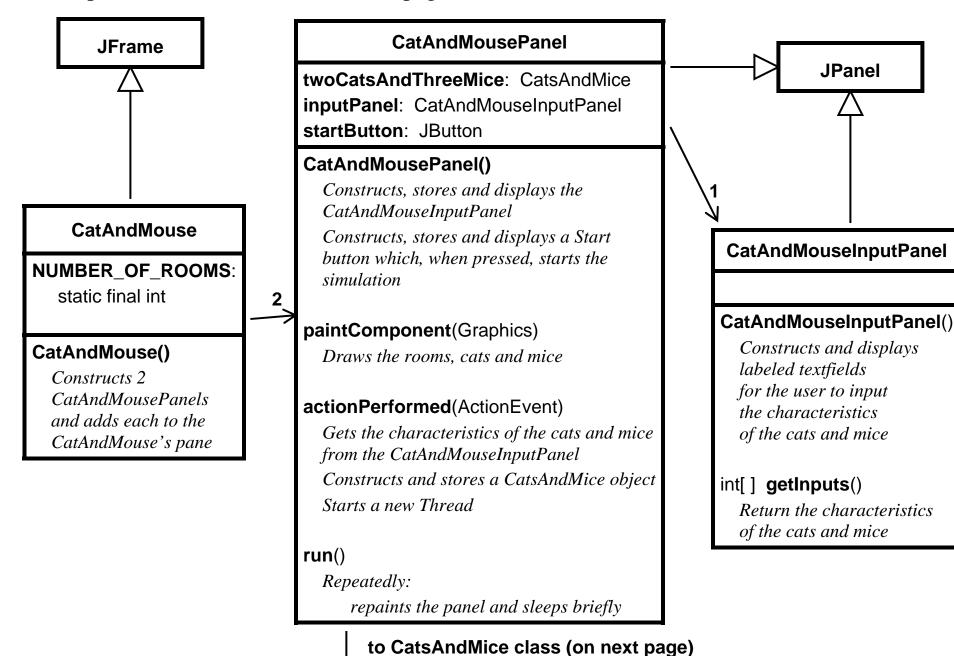
Two cats and three mice are in a house with 10 rooms. The cats and mice move from room to room randomly, as long as they are alive. When a cat enters a room, if a mouse is in that room, the cat eats the mouse. Then the cat sleeps in that room for a while. When a mouse enters a room, the mouse sometimes leaves immediately, but otherwise it sleeps in that room for a while.

Classes and associations



Classes:

We have implemented all the classes on this page



Classes (continued)

You implement only the constructor of this class

1 from CatAndMousePanel class (on previous page)

CatsAndMice

cat1: Cat cat2: Cat

mouse1: Mouse
mouse2: Mouse
mouse3: Mouse

CatsAndMice(

int, int,
int, int,
double, double, double)

Constructs 2 Cats and 3 Mice, where:

- The first 2 arguments specify the timeInRoom values for the 2 Cats
- The next 3 arguments specify the timeInRoom values for the 3 Mice

The last 3 arguments specify the skittishness values for the 3 Mice

Animal **getAnimal**(int k)

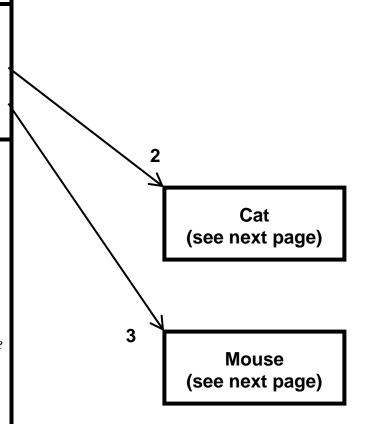
If k is 1, returns the 1^{st} Cat

If k is 2, returns the 2^{nd} Cat

If k is 3, returns the 3^{rd} Cat

If k is 4, returns the 1^{st} Mouse

If k is 5, returns the 2^{nd} Mouse



Classes (continued) You implement all of these classes

Animal

isAlive: boolean

True if the animal is alive, else false

room: int

The room that the animal is currently in

timeInRoom: int

Number of milliseconds that the animal sleeps in a room

Animal(int)

Set the timeInRoom for this Animal to the given argument Start a new Thread

enterRoom(int k)

Simulate the Animal entering the given room by: Sleep in the room for this Animal's timeInRoom

int getRoom()

Return the room that the animal is currently in

boolean isAlive()

Return true if the Animal is alive, else return false

run()

Repeat:

- 1. Choose a random room (i.e., a random number between 1 and CatAndMouse.NUMBER_OF_ROOMS)
- 2. Enter that room

until the Animal is no longer alive



mouse1: Mousemouse2: Mousemouse3: Mouse

Cat(Mouse, Mouse, Mouse, int)

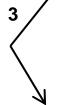
Set the Cat's Mice to the 1st three arguments Set the Cat's timeInRoom to the 4th argument

enterRoom(int k)

Simulate the Cat entering the given room by:

- 1. For each of the Cat's 3 Mice:

 If the Mouse is in the same room that the cat entered, the Cat eats the Mouse (i.e., the Mouse is eaten)
- 2. Sleep in the room for this Cat's timeInRoom



Mouse

skittishness: double

Mouse(int, double)

Set the Mouse's timeInRoom to the 1st argument Set the Mouse's skittishness to the 2nd argument

beEaten()

Make the Mouse be dead (i.e., not alive)

enterRoom(intk)

Simulate the Mouse entering the given room by:

- With probability given by skittishness, the Mouse does nothing (thereby leaving the room immediately)
- Otherwise, the Mouse sleeps in the room for this Mouse's timeInRoom