Logitech

HW 09

1. Mediator, make sure you allow communication between mediator and colleague.  
  
 int main()  
 {

FileSelectionDialog fileDialog;  
fileDialog.add(new ButtonComponent); // colleague 0  
fileDialog.add(new ScrollComponent); // colleague 1  
fileDialog.add(new TextboxComponent); // colleague 2  
  
fileDialog.handleEvent( 0, "Hello" ); // send "Hello" to colleague 0  
fileDialog.handleEvent( 2, "Hi" ); // send "Hi" to colleague 2   
EXPECT\_STREQ(fileDialog.get\_response(0), "Button received Hello");  
EXPECT\_STREQ(fileDialog.get\_response(2), "Textbox received Hi");

}  
  
2. Memonto, implement undo.  
  
 TEST  
 {

Caretaker caretaker = new Caretaker; //keep the memonto  
Originator originator = new Originator (0); //state =0  
caretaker.addMemento( originator.save() );  
originator.setState(1);  
caretaker.addMemento( originator.save() );  
originator.setState(2);  
EXPECT\_EQ(originator.get\_state(), 2);  
originator.undo(caretaker.get\_last\_memento());  
EXPECT\_EQ(originator.get\_state(), 1)  
  
// You can move on implementing redo.

}  
  
  
3. Flyweight, make sure you only create and delete two instances of TILE. You can prepare some prompts on the screen when TILE is created or deleted.  
  
  
 TEST  
 {

std::vector< std::vector<TILE\*> > mytiles (100, std::vector<TILE\*>(100));  
 for (auto& x : mytiles)  
 {  
 for (auto& y : x)  
 {  
 // Sprinkle some hills.  
 if (random(10) == 0) // 1/10 chance  
 {  
 y = Terrain\_Factory::get\_instance()->create\_hill();  
 }  
 else  
 {  
 y = Terrain\_Factory::get\_instance()->create\_grass();  
 }  
 }

}  
  
 for (int x = 0; x< 100; x++ )  
 {  
 for (int y=0; y<100; y++)  
 {  
 mytiles[x][y]->print\_basic\_info();  
 mytiles[x][y]->print\_position\_info(x,y);  
 }  
 }  
 }